Journal of Computer-Aided Molecular Design **17:** 231–240, 2003.
© 2003 *Kluwer Academic Publishers. Printed in the Netherlands.*

New developments in PEST shape/property hybrid descriptors

Curt M. Breneman[1], C. Matthew Sundling[1], N. Sukumar[1], Lingling Shen[1], William P. Katt[1] & Mark J. Embrechts[2]

[1]*Department of Chemistry;* [2]*Department of Decision Sciences and Engineering Systems, Rensselaer Polytechnic Institute, Troy, New York, 12180, USA*

MS received 1 November 2002; accepted for publication 15 November 2002

Summary

Recent investigations have shown that the inclusion of hybrid shape/property descriptors together with 2D topological descriptors increases the predictive capability of QSAR and QSPR models. Property-Encoded Surface Translator (PEST) descriptors may be computed using ab initio or semi-empirical electron density surfaces and/or electronic properties, as well as atomic fragment-based TAE/RECON property-encoded surface reconstructions. The RECON and PEST algorithms also include rapid fragment-based wavelet coefficient descriptor (WCD) computation. These descriptors enable a compact encoding of chemical information. We also briefly discuss the use of the RECON/PEST methodology in a virtual high-throughput mode, as well as the use of TAE properties for molecular surface autocorrelation analysis.

Electron-density-derived (TAE) descriptors

While traditional QSAR methods have been successfully employed within homologous sets of molecules, they have been much less effective when applied to datasets containing a great deal of structural variation. Much of this difficulty can be attributed to the type of molecular property descriptors employed. While descriptors representing simple molecular properties provide intuitive insight into the physicochemical nature of the activity/property under consideration, other descriptors that correlate with less clearly defined intermolecular interactions can often lead to models with excellent predictive power [1].

Quantum chemically derived descriptors have a number of advantages over empirically derived indices: they are related to the actual molecular electron density, they are applicable to a wide variety of molecules and are readily accessible through semi-empirical or small *ab initio* calculations. Within the confines of requiring experimental data for a training set, QSAR/QSPR models employing theoretical descriptors now provide flexibility for computing physical, chemical and biological properties. The disadvantage of such descriptors is the intensive computational effort required to generate them, through quantum chemical calculations, precluding their routine use for large molecules or large datasets. This drawback is circumvented in fragment-based approaches. Transferable Atom Equivalent (TAE) descriptors encode the distributions of electron density based molecular properties, such as kinetic energy densities, local average ionization potentials, electrostatic potentials, Fukui functions, electron density gradients and second derivatives, in addition to the density itself. Table 1 shows a complete list of TAE descriptors. The TAE descriptors are capable of generating high quality models; however, since these descriptors are non-orthogonal, traditional regression analysis (such as multiple regression analysis) is not appropriate, as the system can become overdetermined. Modeling techniques such as principal component analysis, artificial neural networks [2–4], kernel partial least squares regression or Support Vector Machine (SVM) regression [5], can be fruitfully employed on such data, with feature selection accomplished using genetic algorithms [6] or sensitivity analysis [7]. These routines are incorporated in the StripMiner™ package developed at RPI.

Table 1. Electron-density-derived TAE descriptors. ρ(r) represents the electron density distribution.

Integrated Electronic Properties		
Energy		
Electron population		
Volume		
Surface area		
Surface electronic properties (extrema, surface integral averages and histogram bins are available for each property)		
SIEP	Surface integral of electrostatic potential	
EP	Electrostatic potential	$EP(r) = \sum_\alpha \frac{Z_\alpha}{\lvert r - R_\alpha \rvert} - \int \frac{\rho(r')dr'}{\lvert r - r' \rvert}$
DRN	Electron density gradient normal to 0.002 e/au^3 electron density isosurface	$\nabla\rho.\mathbf{n}$
G	Electronic kinetic energy density	$G(r) = -(1/2)(\nabla\psi^*.\nabla\psi)$
K	Electronic kinetic energy density	$K(r) = -(1/2)(\psi^*\nabla^2\psi + \psi\nabla^2\psi^*)$
DGN	Gradient of the K electronic kinetic energy density normal to surface	$\nabla K.\mathbf{n}$
DGN	Gradient of the G electronic kinetic energy density normal to surface	$\nabla G.\mathbf{n}$
F	Fukui F$^+$ function scalar value	$\mathrm{F}^+(\mathbf{r}) = \rho_{\mathrm{HOMO}}(\mathbf{r})$
L	Laplacian of the electron density	$L(r) = -\nabla^2\rho(r) = K(r) - G(r)$
BNP	Bare nuclear potential	$BNP(r) = \sum_\alpha \frac{Z_\alpha}{\lvert r - R_\alpha \rvert}$
PIP	Local average ionization potential	$PIP(r) = \sum_i \frac{\rho_i(r)\,\lvert\varepsilon_i\rvert}{\rho(r)}$

Wavelet coefficient descriptors (WCD)

In recent years, wavelet encoding has gained popularity in diverse applications as an efficient means of data compression and pattern recognition. The wavelet basis has advantages over the Fourier basis in that, while the trigonometric functions used in Fourier expansion are monochromatic in frequency but entirely delocalized in position, the wavelet basis is well localized in both frequency and position. Wavelet encoding and decoding are accomplished by a simple scaling and dilation algorithm.

The Discrete Wavelet Transform (DWT) is a fast linear operation on a data vector with length 2^n (where n is an integer) that transforms the original data vector into a wavelet coefficient vector of the same length. The resulting vector consists of 2^{n-1} scaling coefficients and 2^{n-1} detail coefficients. The former represent a smoothed envelope of the data, while the latter give the detailed deviations from this smoothed function. The scaling coefficient vector can, in turn, be subjected to another round of DWT, resulting in 2^{n-2} scaling coefficients and 2^{n-2} detail coefficients, encoding a coarser level of detail. For a data vector of length 2^n the DWT can be performed $n - 1$ times, resulting in a single scaling coefficient and $2^n - 1$ detail coefficients. This entire procedure can be reversed in the same iterative manner to decode the wavelet coefficient vector, reconstructing the original signal. Since molecular surface property distributions are smoothly varying functions in property space, it is reasonable to expect that the important physicochemical information relevant to intermolecular interactions will be contained in the scaling and first few levels of detail coefficients, rather than in the finer levels of detail. Discarding the finer levels of detail coefficients therefore results in significant data compression with little loss of signal. In PEST, each of the ten surface electronic properties in Table 1 is represented by a 1024-point distribution and encoded in the symmlet-8 wavelet basis, retaining only 32 wavelet coefficients. Property distributions reconstructed from these 32 wavelet coefficients reproduce the original distributions to greater than 95% accuracy.

TAE WCDs generated from *ab initio* quantum computations have been employed with success,

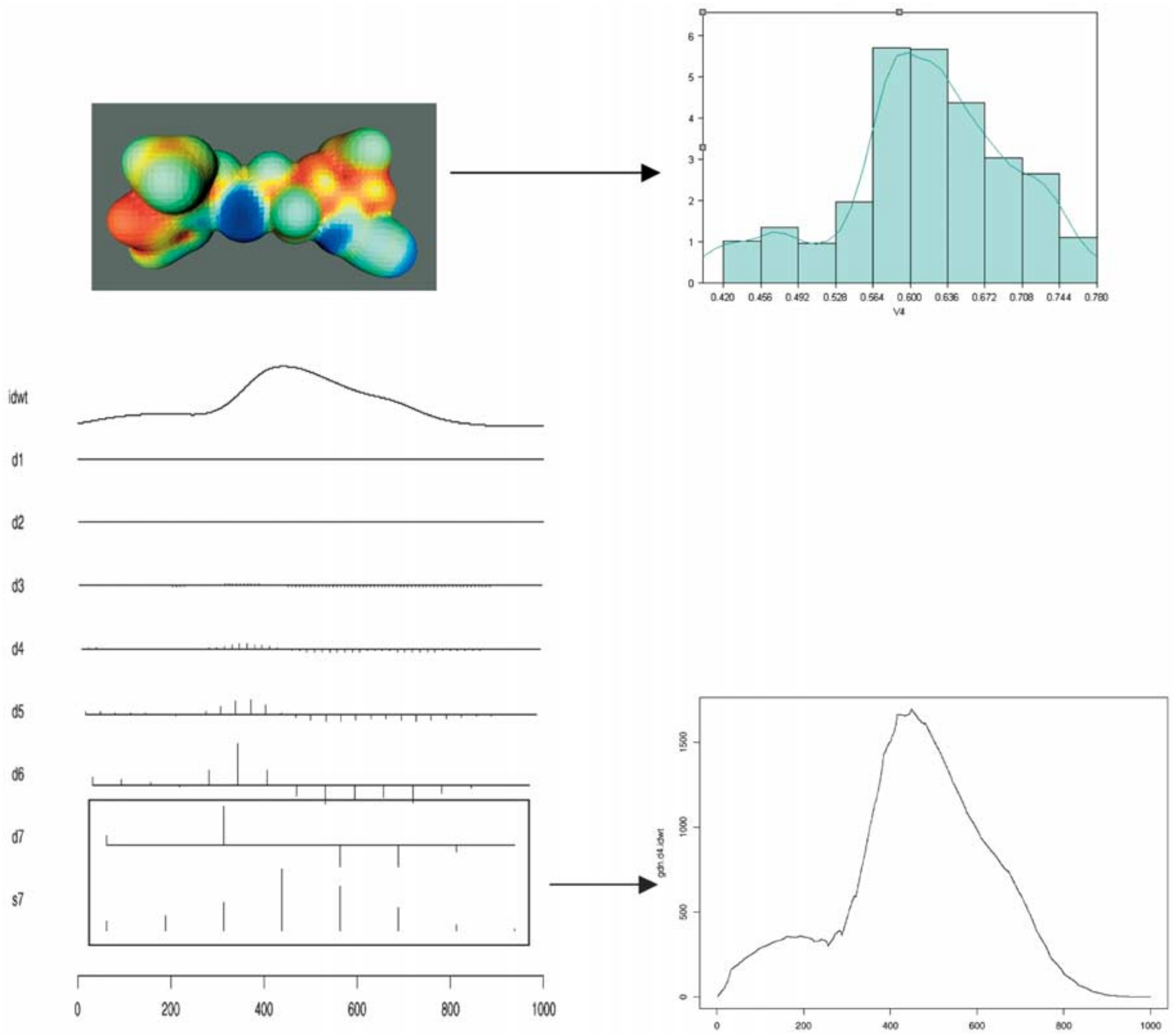

Figure 1. (a) Surface histogram distribution (upper panel); (b) Discrete wavelet transform (lower left) and inverse transform (lower right) for PIP using only 16 low-order coefficients.

in conjunction with other TAE and traditional descriptors, to model a variety of chemical and biochemical phenomena. Since *ab initio* quantum chemical descriptors are laborious to compute and impracticable to implement in high-throughput mode, it is of considerable value to obtain these WCD descriptors through the RECON method. Just as for other TAE descriptors, wavelet coefficients of atomic property distributions (WCD) can be simply summed (weighted by the atomic surface area) to give molecular wavelet representations, from which approximate distributions in property space can be reconstructed, if desired. This has been implemented in Beta version 6.3 of RECON and the atomic wavelet library is presently being constructed.

Hybrid shape-property descriptors (PEST)

TAE descriptors can be supplemented, with some increase in computational time, by hybrid shape-property descriptors, that encode information about the molecular shape, without requiring an alignment procedure for their computation. The supplemental information available from these descriptors is useful where the shape of the molecule plays a determining role in binding.

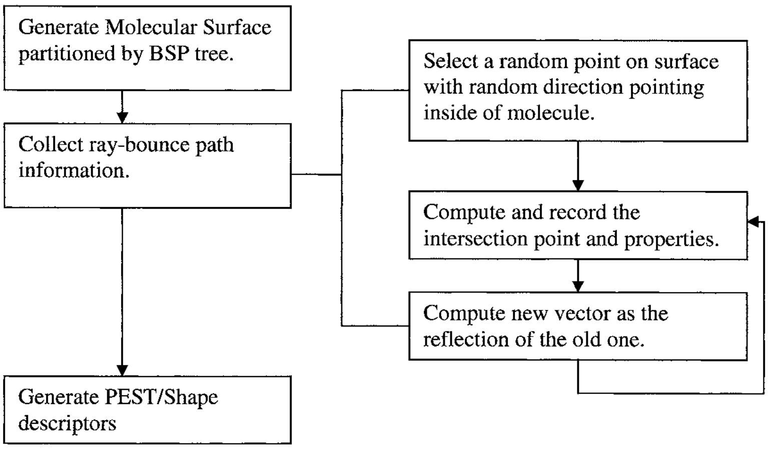
Generate Molecular Surface partitioned by BSP tree.
Collect ray-bounce path information.
Generate PEST/Shape descriptors
Select a random point on surface with random direction pointing inside of molecule.
Compute and record the intersection point and properties.
Compute new vector as the reflection of the old one.

Figure 2. PEST Flowchart.

Based upon an idea originally proposed by Zauhar [8], PEST (Property Encoded Surface Translator) descriptors are generated using TAE molecular surface representations to define property-encoded boundaries for implementing the Zauhar Shape Signature ray-tracing approach to shape/property convolution. The Shape Signature approach seeks to encode the shape of a molecular volume using the distribution of ray lengths obtained by performing a ray-tracing procedure within the molecule from an arbitrary starting position. The converged ray-length distribution then represents a shape signature of the original molecular envelope. PEST hybrid shape-property descriptors have proven to be very useful and out-performed other descriptor sets in a number of applications, as shown below.

Algorithm and methodology

Earlier TAE descriptors entirely ignore the 3D spatial information of the isosurface from which they are computed. The goal of the TAE PEST hybrid shape-property descriptors is to retain some of the shape information of the surface, and to use it to resolve the molecule's property values. This approach utilizes a ray/surface intersection detection method derived from the computational geometric/graphic technique of ray tracing and regular light reflection mathematics.

The general algorithm for computing the TAE shape descriptors follows these basic steps:
I. Compute molecular surface property distributions;
II. Collect ray-bounce path information using light reflection algorithm.
III. Generate TAE/Shape descriptors

Each of these steps is now discussed in detail.

I. Computation of molecular surface property distributions

Molecular surface property distributions can be computed from either *ab initio* molecular wave functions or reconstructed from fragment-based distributions using the RECON algorithm. We define the molecular surface as the 0.002 electrons/Bohr^3 electron density isosurface. This corresponds approximately to the Van der Waals surface.

II. Collecting ray-bounce path information

The steps involved in this procedure are summarized below:

1. Select any random starting point on the molecular isosurface and a random direction vector (starting from the current point) with the constraint that it must be directed towards the inside of the molecule;

As, there is no guarantee that the triangulated isosurfaces are closed or well-behaved in all cases; specific heuristics are employed for determining an appropriate starting point and initial direction vector for an isosurface: (1) select a random point suspected of being inside the surface, (2) propagate M random rays from that point, (3) determine the number of surface intersections for each ray, and (4) if the majority (i.e. greater then 95%) of the rays intersect an odd number of times, then the suspected point is an interior point; else repeat these steps until an interior point is

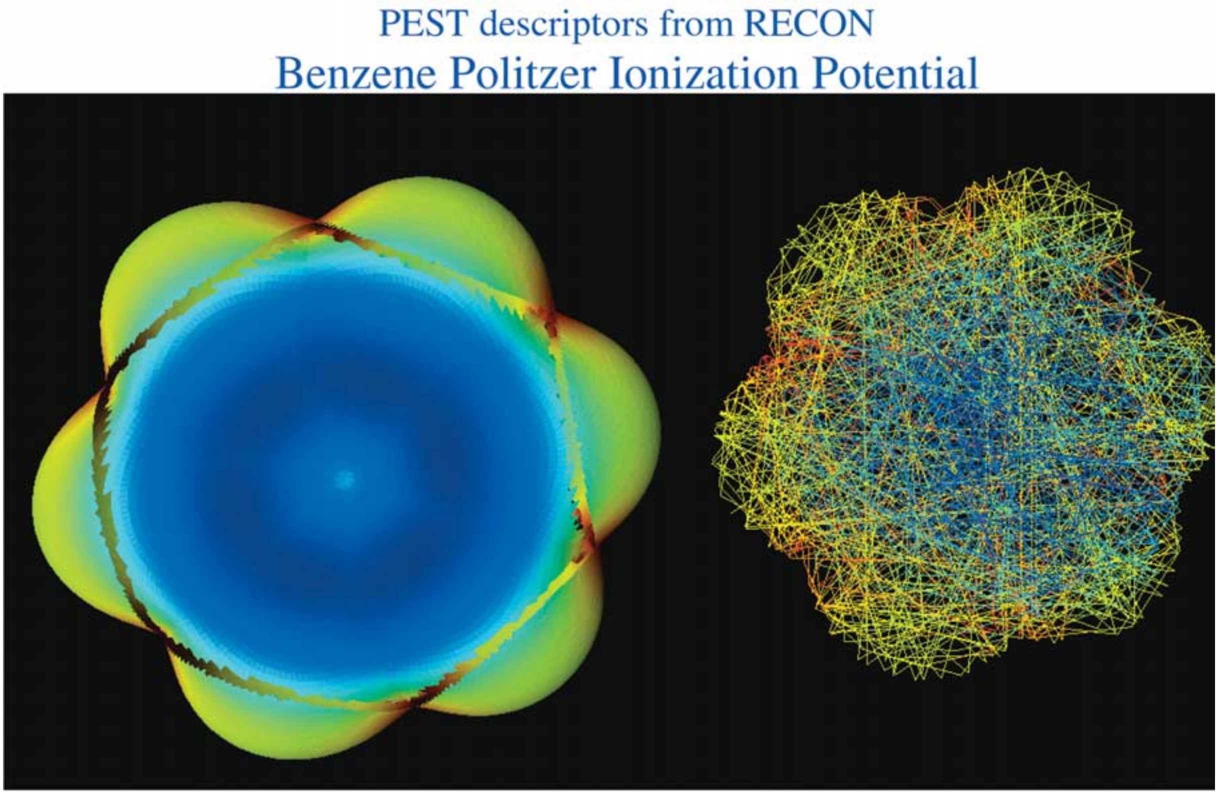
PEST descriptors from RECON
Benzene Politzer Ionization Potential

Figure 3. PEST descriptors from RECON. Benzene Politzer Ionization Potential.

determined. For determining a random point on the surface and its associated vector pointing towards the interior of the surface, the following simple steps are used: (1) select a random direction vector starting from an interior point, (2) determine the first intersection of the ray's path with the surface, and (3) determine the reflection of the vector off the interior surface. The intersection point and reflection vector are used as the starting position and direction for the ray-bouncing path.

2. Follow the path of this vector until it strikes the inside surface of the molecule;

Propagating a vector to determine its intersections with a surface can be restated as searching for the triangles of the surface that the line of the vector intersects. This is a classic problem in computational geometry sometimes termed ray/triangle-intersection search or ray tracing. The complexity of the problem stems from the fact that the brute force method of testing every triangle in the surface for a positive intersection is extremely computationally intensive. Classical algorithmic solutions designed to minimize searching time can be found in the computational geometry and computational graphics literature; they reduce the number of intersection computations performed. Algorithms, usually based on partitioning 3D space into successively smaller regions that encase different portions of the surface triangles themselves, allow for quick, intelligent triangle lookups, which drastically reduces the total number of ray/triangle tests performed. Using a hierarchical tree structure to capture the spatial relationships of the different regions allows quick ray-intersection testing of an entire region of space. If a ray does not pass through a large portion of space, then it cannot pass through the smaller portions of space (or their culled triangles) contained therein. This notion of testing successively finer-grain portions of space allow the triangle search to be significantly accelerated, without loss of accuracy in detection of intersections.

Each ray/triangle intersection test is a trivial linear algebra problem, that of determining if (1) a solution exists that satisfies the equations for the plane of the triangle and for the line of the vector, and (2) if that solution exists in the interior of the triangle. PEST uses the classic Binary Space Partitioning (BSP) tree (Fig-

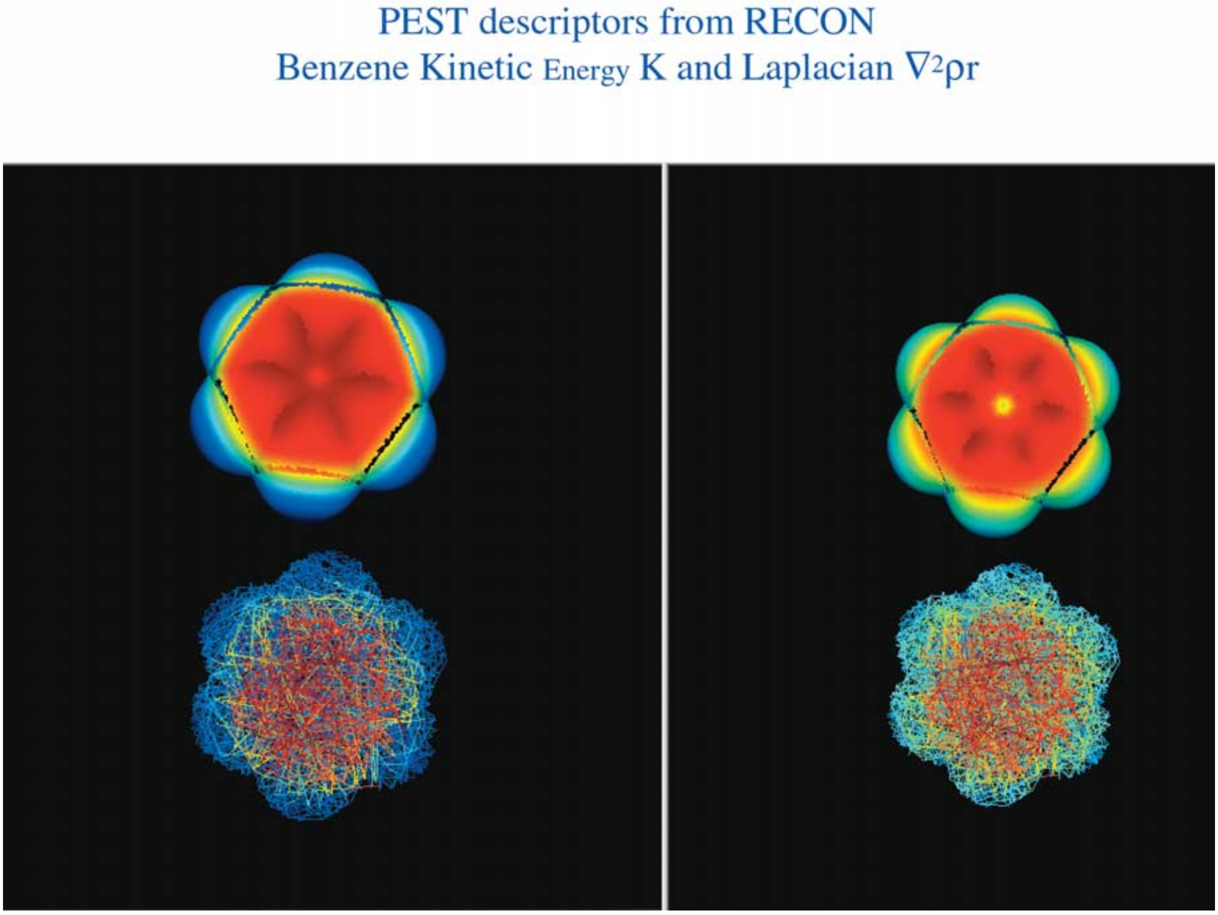
PEST descriptors from RECON
Benzene Kinetic Energy K and Laplacian ∇2ρr

Figure 4. PEST descriptors from RECON. Benzene Kinetic Energy K and Laplacian $\nabla^2\rho$r.

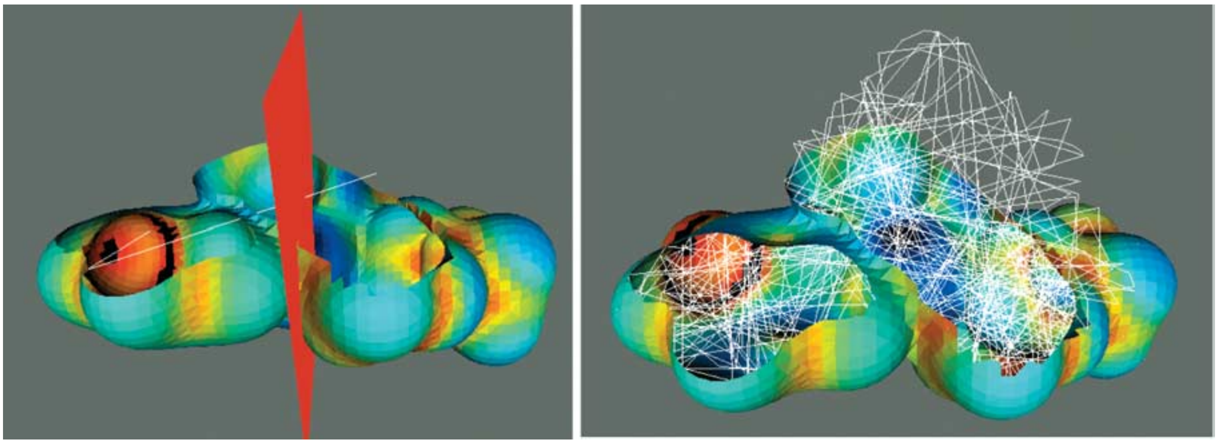

Figure 5. Property-encoded ray tracing. Binary space partitioning plane (in red), and a sparse (750 segment PEST run).

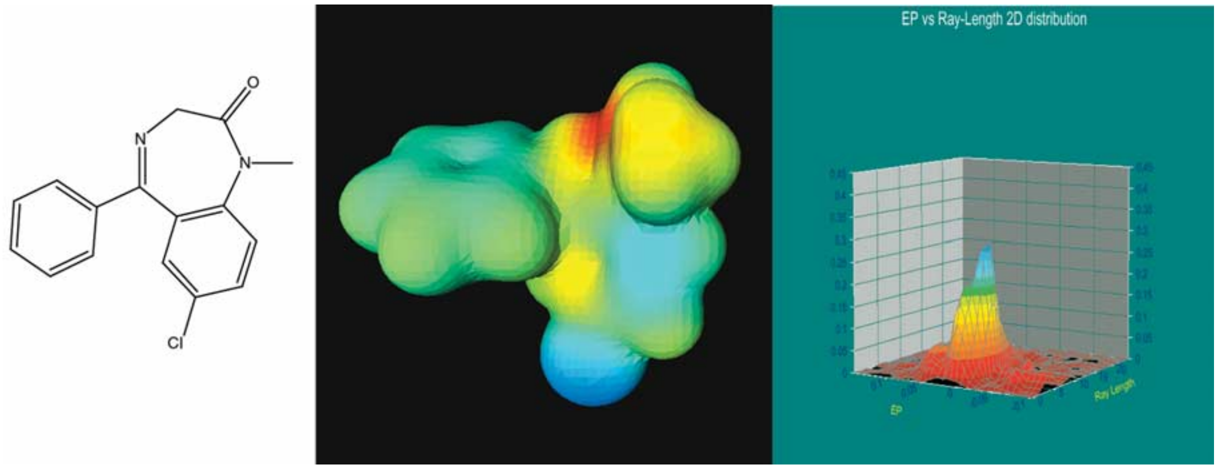

Figure 6. Benzodiazapine structure, TAE surface reconstruction and PEST shape/property signatures.

ure 5), one of the simplest space partitioning structures using simple axis-parallel planes to create box-like regions of space.

The difficulty with computing a point of intersection (assuming it exists) stems from the imprecision of the computer data types used in the computation. When the point of intersection is computed using standard imprecise data types, it contains round off error – in essence, this point is most likely outside the plane of the triangle. More specifically, the point may have been rounded out of the isosurface, and the ray-bounce path has escaped the isosurface. The heuristic used in PEST, is to reel the point back into the interior of the surface by taking minute steps in the opposite direction to the incident ray. Roughly speaking, this amounts to marching the point back into the interior of the isosurface along the path of the original vector. Obviously, the true distance between points of intersection has been lost, but the error is negligible.

The openings in a surface, due to a ‘poor’ triangulation or otherwise, would allow a ray-bouncing path to eventually bounce to the outside of the molecule. PEST deals with this situation by detecting an escaped ray (by culling the theoretical molecular space by an ‘infinite’ boundary, and detecting ray/intersections with it), ending the ray/bounce path, and beginning a new path from within the interior according to the original algorithm specified above. The break in the path does not affect the overall computation of these shape descriptors or their performance in QSAR or related learning tasks.

Another situation occurs when the incident ray intersects a triangle precisely on one of its edges. This situation will cause the ray to leak out the cracks in the triangulated isosurface and escape from being bounced as expected. The leaked ray is detected and corrected by the addition of a random, slight perturbation of the original ray direction, moving its path away from the edge of a triangle. This does not greatly affect the ray-bouncing path and offers a simple solution to avoid the pitfalls of more complex correction heuristics.

To avoid situations where the ray-bouncing path is ‘stuck’ in some sort of loop or repeated sequence, periodic random perturbations to the direction vector are added. In addition, to avoid the ray-bouncing path getting stuck in one portion of the molecule’s volume, periodic termination and random restarts of the bouncing path are inserted into the algorithm’s loop.

3. Compute and record this intersection point and all associated surface electronic (TAE) properties;

Once the point of intersection has been determined, all the properties associated with the isosurface at that location are computed and stored. In addition, the distance between the previous intersection point and the current intersection point is computed and stored. Property values derived from the electronic surface are only known at the vertices of the triangles that comprise the surface. Once the point of intersection is known, its associated property values are computed assuming the properties are linearly dependent on position in the plane.

4. Compute a new direction vector as the reflection of the old direction vector off the inside of the surface;

The reflection of the incident ray-path with the surface triangle is computed using normal ‘light re-

Table 2. Results (R^2 and RMSE) of modeling on HIV dataset.

Methods	PAD (Train)		PAD (Test)	
	R^2	RMSE	R^2	RMSE
PLS (LOO)	0.902	0.436	0.601	0.686
PLS (Bootstrap)[a]	0.866	0.520	0.631	0.608
ANN[b]	0.872	0.091	0.899	0.057

[a] Total of 64 molecules, 52 in the training set and 12 in the testing set. 500 bootstraps performed leaving 10 molecules out of 52 for every generation.

[b] ANN performed with Metaneural™ using 52 molecules in the training set and 12 in the testing set. 58 nodes in the input layer. Two hidden layers with 30 and 6 nodes respectively. One node in the output layer. The sigmoid learning parameter is 0.02. The stopping error is 0.1.

flection' computations, giving the new direction of the ray-bounce path.

5. Repeat steps 1–4 for a determined satisfactory number of 'ray bounces'.

It is necessary to repeat the above sequence of steps to obtain the number of bounces desired. This information is summarized in the shape descriptors (see next step); so a thorough exploration of the molecule is necessary. Our investigations show that the average surface-sampling density is a good metric to determine an algorithmic ending point. Typically, it has been sufficient to sample around 10 points/Å^2. It is important to sample all molecules similarly, to allow direct comparison of their resulting shape descriptors. The dependence of the descriptors on sample density is a direct one: as the number of ray-bounces increase, the shape descriptors converge, given a particular isosurface, irrespective of the randomness in the selection of the starting positions and path directions.

III. Generation of TAE shape/property descriptors

The path information (i.e. distance between successive intersection points, and the property values at the intersection points) can be summarized into 2-D histograms to obtain a surface shape profile (Figure 6). For a single electronic property, a 2-D histogram having the distribution of distances (x-axis) versus their associated property value (y-axis) gives a characteristic distribution (z-axis), based on the overall shape and property value distribution of the molecule. Such a 2-D histogram is created for every surface property for every isosurface processed. The bins of the 2-D histograms are used directly as the molecule's descriptors in typical computational learning models.

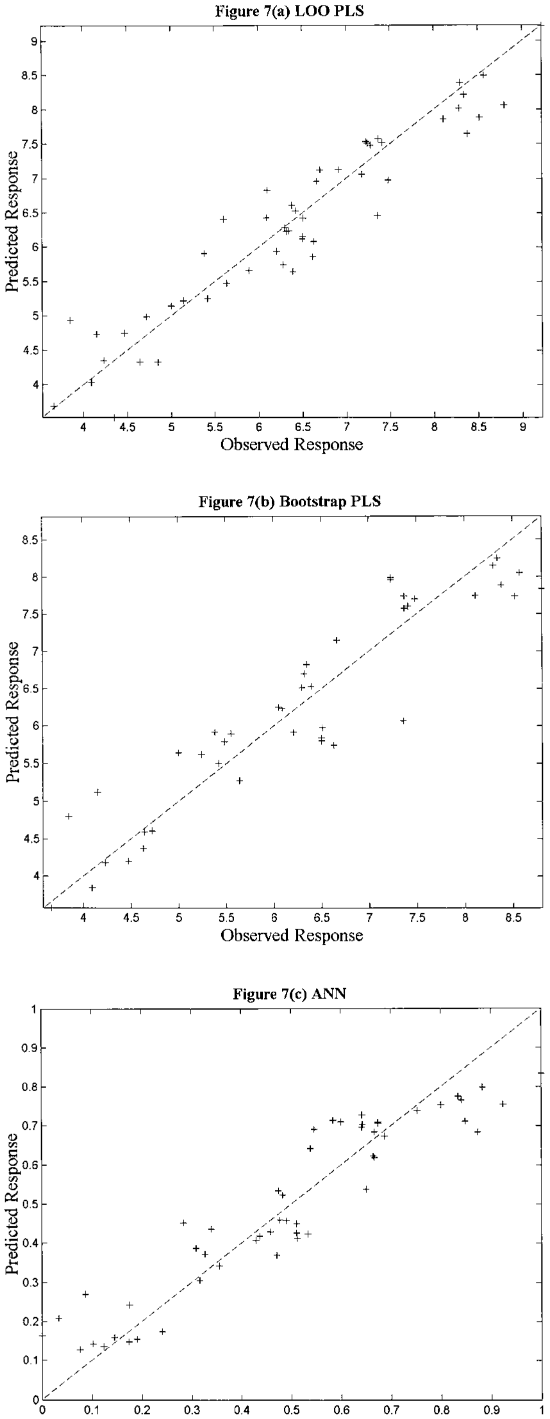
Figure 7(a) LOO PLS

Figure 7(b) Bootstrap PLS

Figure 7(c) ANN

Figure 7. PAD results on training set. (a) LOO PLS; (b) Bootstrap PLS; (c) ANN.

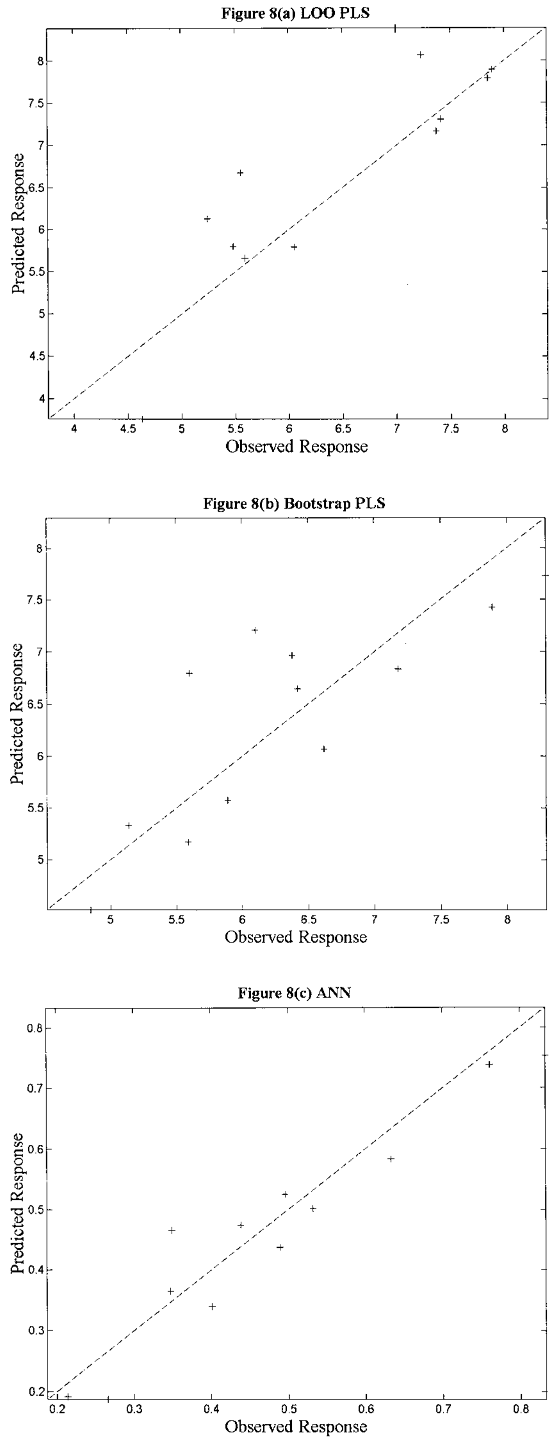

Figure 8. PAD results on training set. (a) LOO PLS; Bootstrap PLS; (c) ANN.

Autocorrelation descriptors

The autocorrelation method has been employed in the field of structure-activity relationships since 1980. The general equation for autocorrelation is defined below:

$$A(d) = a \sum_{x} P_x^b \times P_{x+d}^c$$

where $A(d)$ is the autocorrelation component corresponding to topological distance d in the molecular surface, P is the property associated with surface points, x and $x + d$ are surface points, the distance between them being less than or equal to d and a, b, c are coefficients.

The advantages of using autocorrelation methods are that they are canonical and hence independent of the coordinates. Autocorrelation methods can represent the molecular geometry characteristics, while substantially reducing the input information.

Our implementation of autocorrelation descriptors is derived from Gasteiger's definition [9]:

$$A(d) = 1/L \sum_{x} P_x \times P_{x+d}$$

where $A(d)$ is the autocorrelation component of distance d, P is the property associated with surface points x and $x + d$, the distance between them being less or equal to d and L is the total number of point-pairs of d. Our autocorrelation descriptors are created based on TAE and PEST descriptors.

Autocorrelation descriptors generation algorithm

For each ray in PEST, the length of the ray and the product of the property values at starting and ending points were computed. The distribution was binned into 20 bins along the ray length and the autocorrelation values for each bin calculated. For 10 TAE properties this yields a total of 200 PEST autocorrelation descriptors (PAD).

Results and discussion

The HIV dataset has 64 molecules, of which 52 molecules were included in the training set, with 12 molecules being used in the test set. The modeling techniques used were Leave-one-out (LOO) PLS, Bootstrap PLS and ANN from Analyze™. The modeling results are summarized in Table 2 and Figures 7–8. In Table 2, the first column lists the modeling methods. The second and third columns show modeling and prediction results using PAD. Figure 7 shows modeling results for PAD with LOO-PLS, Bootstrap-PLS and ANN, while Figure 8 shows the corresponding prediction results.

The best model was obtained using ANN. In PAD, R^2 for the model was 0.872 and R^2 for prediction was 0.899. For the same dataset, the best model was obtained using ANN and a combination of MOE and RECON descriptors with an R^2 of 0.73 [10, 11]. The Autocorrelation descriptors are better than the other descriptors for this dataset. This indicates that this dataset is strongly conformation-dependent. PEST autocorrelation descriptors represent a lot of surface shape information.

References

1. Breneman, C.M. and Rhem, M., J. Comput. Chem., 18 (1997) 182–197.
2. Embrechts, M.J., Robert Kewley, J. and Breneman, C., *Computationally Intelligent Data Mining for the Automated Design and Discovery of Novel Pharmaceuticals*. in *Smart Engineering Systems: Neural Networks, Fuzzy Logic, Evolutionary Programming, Data Mining and Rough Sets.* St. Louis, MO: ASME Press, 1998.
3. Embrechts, M.J., *et al.* Scientific Data Mining with Strip-MinerTM. In *Proceedings, 2001 SMCia Mountain Workshop on Soft Computing in Industrial Applications*. Blacksburg – Virginia: IEEE Press.
4. Kewley, R., Embrechts, M.J. and Breneman, C.M., Neural Network Analysis for Data Strip Mining Problems, In *Intelligent Engineering Systems through Artificial Neural Networks*, Dagli, C. Editor. St Louis, MO; ASME Press. 1998. 391–396.
5. Bennett, K. and Campbell, C., SIGKDD Explorations, 2 (2000) 1–13.
6. Bennett, K., Demiriz, A. and Embrechts, M. Semi-Supervised Clustering Using Genetic Algorithms. In *Artificial Neural Networks in Engineering (ANNIE'99)*. 1999.
7. Embrechts, M.J., *et al.* Bagging Neural Network Sensitivity Analysis for Feature Reduction in QSAR Problems. In *2001 INNS – IEEE International Joint Conference on Neural Networks*. 2001. Washington D.C: IEEE Press.
8. Zauhar, R.J. and Welsh, W.J. In *American Chemical Society National Meeting.* Washington, D.C.: American Chemical Society, 2000.
9. Wagener, M., Sadowski, J. and Gasteiger, J., J. Am. Chem. Soc., 117 (1995) 7769–7775.
10. Lockwood, L. In *Chemistry.* Rensselaer Polytechnic Institute: Troy, 2000.
11. Aboufadel, E. and Schlicker, S., *Discovering Wavelets*. 1999, New York: John Wiley & Sons.
12. Hubbard, B.B., *The World According to Wavelets*. 1996, Wellesley, MA: A.K. Peters, Inc.
13. Daubechies, I., *Ten Lectures on Wavelets*. 1992, Philadelphia: SIAM.