DNurbs: DNA modeled with NURBS

Mike Carson and Zi Yang

Center for Macromolecular Crystallography, The University of Alabama at Birmingham, Alabama USA

DNurbs (dee-in-urbs) are a small number of bicubic patches wrapping each base pair in complementary DNA. NURBS, the nonuniform rational B-spline surfaces now popular in computer graphics, are employed for the patches. Control points for the surface patches are generated from the molecular surface based on canonical base pairs.

Keywords: DNA, NURBS, surfaces, splines, ribbons

INTRODUCTION

Illustrations of the famous double helix of Watson and Crick have become an integral part of our culture. This journal features this image (generated by Jane Burridge) on its cover page.

The method given here attempts to model the solid surface of complementary DNA strands using parametric surface patches rather than individual atoms. The work is, in part, motivated by a desire to illustrate simulations of supercoiled DNA, where the computational model does not employ individual atoms.[1,2] Color and texture mappings may be applied to these patches to present chemical information.

The author is indebted to the drawings of Irving Geis (e.g., in standard texts[3]) and the crystallographic work of Dickerson's group.[4] This method is similar in spirit, though not in style, to that of Lesk and Lesk,[5] who employed iconic line drawings.

METHODS

Spline curves and surfaces

Earlier work succinctly represented protein ribbon models as cubic B-spline curves fit to peptide planes,[6] with the peptide bond forming a natural underlying basis. Raster versions and an extension of the method to nucleic acids were later developed.[7] Here it was suggested that bicubic surface patches might be appropriate for this modeling. Bicubic ribbons, as well as shrink-wrapped surfaces based on patches, were developed on and shown with the AT&T Pixel Machine, one of the first devices with such patches provided as built-in primitives.[8]

A key feature of these programs was the use of the spline curves and surfaces supported by the hardware (e.g., Evans and Sutherland, Silicon Graphics) and software libraries (e.g., PEX, OpenGL) of the current generation of graphics workstations commonly used in molecular modeling. These techniques have become standard in the field of computer graphics, and are described in general texts[9] and specialized monographs.[10] A very brief explanation follows.

A NURB curve can be formulated, in general, as follows:

$$C(t) = (x(t)/w(t) \quad y(t)/w(t) \quad z(t)/w(t))$$

where the polynomial curve, $C(t)$, is given in terms of the ratio of polynomials in a dimensionless parameter, t. These parametric curves usually have $0.0 \leq t \leq 1.0$. These curves are specified in modern graphics libraries by a sequence of "knot" values and a set of "control points" coordinates.[9,10] Simple examples are shown in Figure 1.

A cubic polynomial

$$C(t) = a_0 + a_1t^1 + a_2t^2 + a_3t^3$$

is the most simple type of curve that can be joined piecewise in segments, ensuring continuity in position, tangent and curvature. These curves are specified by giving four Cartesian control point values and a nondecreasing set of 8-knot values that determine exactly how the control points influence the shape of the curve. Figure 1 shows the two types of cubic curves used in this paper: the Bezier and the B-spline forms.

A NURBS surface is defined by two such curves, $C_a(u)$ and $C_b(v)$, which form the basis functions of a surface, $S(u,v)$. Topologically, this is a square with a two-dimensional array of 4×4 (or 16) control points required to specify the surface in terms of the parameters u and v. The curves C_a and C_b need not be of the same type. Figure 2 illustrates such a surface constructed from the cubic curves shown in Figure 1. This surface is known as a bicubic patch. The user only needs to establish the positions of the control points.

Control points for DNA

The classic Watson-Crick structure, B-DNA, provides the canonical model. There are four standard base pairs encountered traversing a single strand starting at the phosphate and proceeding in the 5′–3′ direction: GC, CG, AT and TA. The standard orientation used herein (Figure 3) is defined by looking down the pseudodyad of the strands perpendicular to a single base pair with the phosphate at 12 o'clock (positive Y). Circling clockwise from this phosphate in the plane

Color Plates for this article are on page 195.

Address reprint requests to Dr. Carson, Center for Macromolecular Crystallography, The University of Alabama at Birmingham, Birmingham, Alabama 35294 USA.

Received 8 July 1993; accepted 1 December 1993.

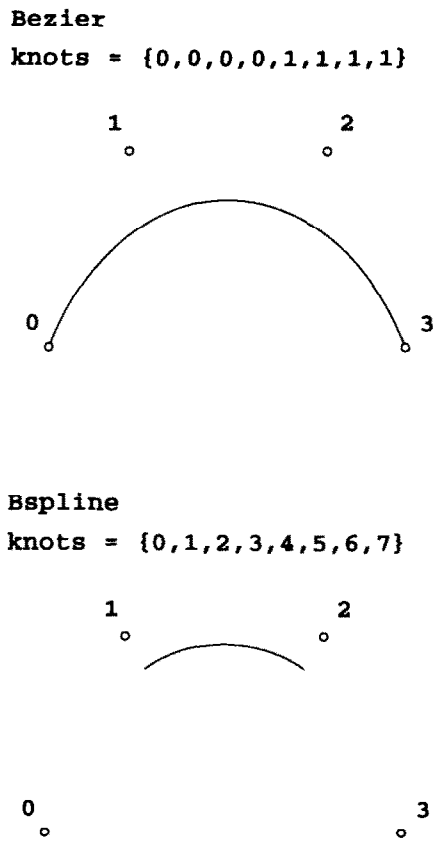

Figure 1. Simple cubic curves. The Bezier and B-spline forms, C(t), are labeled and their 8-knot values given. The four control points labeled 0..3 are identical for both curves. The multiple weighting of the Bezier knot sequence forces the curve to begin and end on the first and last control points. The middle control points control the slope of the curve at the endpoints. The B-spline curve does not, in general, lie on any control point. Adding an additional control point 4 and using the control points 1..4 would create a curve segment that smoothly joins the segment in the figure.

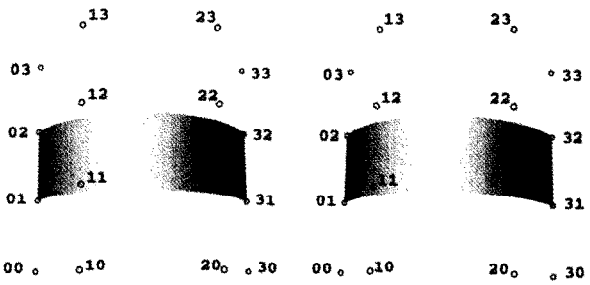

Figure 2. Simple NURBS surface. The basis curve $C_a(u)$ is a Bezier curve, while the basis curve $C_b(v)$ is a B-spline (see Figure 1). The 16 control points are labeled as uv indices 00..33. The resultant surface, S(u,v), is shown rendered in stereo.

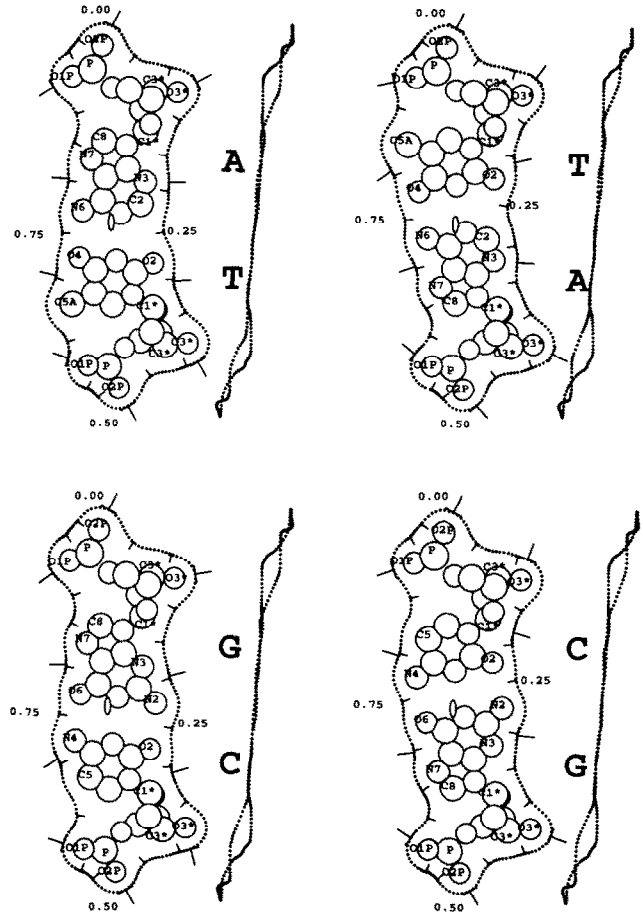

Figure 3. Surface paths around base pairs. The curve B(u), u = 0.00–1.00, traces a path around each of the four labeled base pairs. The paths are composed of about 250 discrete points shown as dots. The surface normals and tangents are also shown in increments of 0.10. The labels at 0.0, 0.25, 0.50 and 0.75 denote the first phosphate, the minor groove, the second phosphate and the major groove. The atomic models are shown in ball-and-stick form, along with the helix pseudodyad. Only atoms that contribute to the surface are labeled. The paths rotated by 90° are also shown.

around the molecular surface are the sugar ring, the minor groove of the purine/pyrimidine pair, the sugar/phosphate of the second strand and the major groove, then back to the phosphate of the first strand.

Figure 3 depicts the data structure employed. Each of the four base pairs has an ordered path of about 250 surface points with their surface normal vectors. The closest atom to each surface point is also maintained. The path, $B(u)$, around the base pair is accessed by the parameter u = 0.0–1.00. Each path is subdivided into four sections: u = 0.00 and u = 0.50 are set to the center of the surface points arising from the outermost oxygen atoms of the first and second phosphate groups, and u = 0.25 and u = 0.75 are the points at which the bases "touch" in the minor and major grooves, respectively. The four quarter sections in the parameter u are of roughly equal length (Table 1).

Table 1. Path lengths around base pairs. The curve B(u), u = 0.00–1.00, circles the surface in the plane of the given base pair as shown in Figure 3. The path lengths (Å) tracing around the surface are given for the four subdivisions of the path as described in the text.

Pair	0.00–0.25	0.25–0.50	0.50–0.75	0.75–1.00	Total
AT	16.33	14.58	14.35	15.38	60.64
TA	14.61	16.05	15.71	14.28	60.65
GC	16.89	14.04	14.23	15.48	60.65
CG	13.68	17.26	15.37	14.37	60.67

The data encapsulated in Figure 3 were created in an automatic fashion, from a series of programs and filtering scripts. An ideal B-DNA structure with the helix axis aligned along Z served as the starting point. Each representative base pair was extracted, with the molecular surface[11] computed at a high dot density (d = 50/Å^2). Only dots with surface normals nearly perpendicular to Z were kept ($|n_z| <$ 0.1). The dots were then culled, keeping only those approximately 0.2 Å apart. Finally, the paths were smoothed by replacing dot_i with the average of dot_{i-1} and dot_{i+1}.

Figure 4 gives examples of approximating the base pair surface path, $B(u)$, with a small number (N) of Bezier curves. First, N points are chosen on $B(u)$. Adjacent points provide the first and last (fourth) control points for the Bezier curve segment. The second and third control points are taken along the tangent (perpendicular to the normal) at the endpoints on $B(u)$. This ensures the next curve segment will join smoothly with the same tangent (i.e., C^1 continuity). The scale of the tangential line may be adjusted to control the curvature of the segments.

Figure 5 gives an example of approximating $B(u)$ with N cubic B-spline curve segments. Here, a nonuniform scale factor (S_x,S_y,S_z = 1.25,1.5,1) is applied about the center of B(u) to enlarge the loop in the x,y plane. (The control points must lie outside the actual base pair surface, as B-spline curves do not, in general, pass through their control points. A similar hack is used with a 1.5 scale factor for the helices in B-spline protein ribbons.[6]) Next, N u-values are chosen. Every four consecutive points define a B-spline curve. (If six control points, $u_0 \ldots u_5$, are chosen, the six curves are formed from $u_0u_1u_2u_3$, $u_1u_2u_3u_4$, $u_2u_3u_4u_5$, $u_3u_4u_5u_0$, $u_4u_5u_0u_1$ and $u_5u_0u_1u_2$.) Each segment joins smoothly not only in tanget, but also in curvature (C^2 continuity).

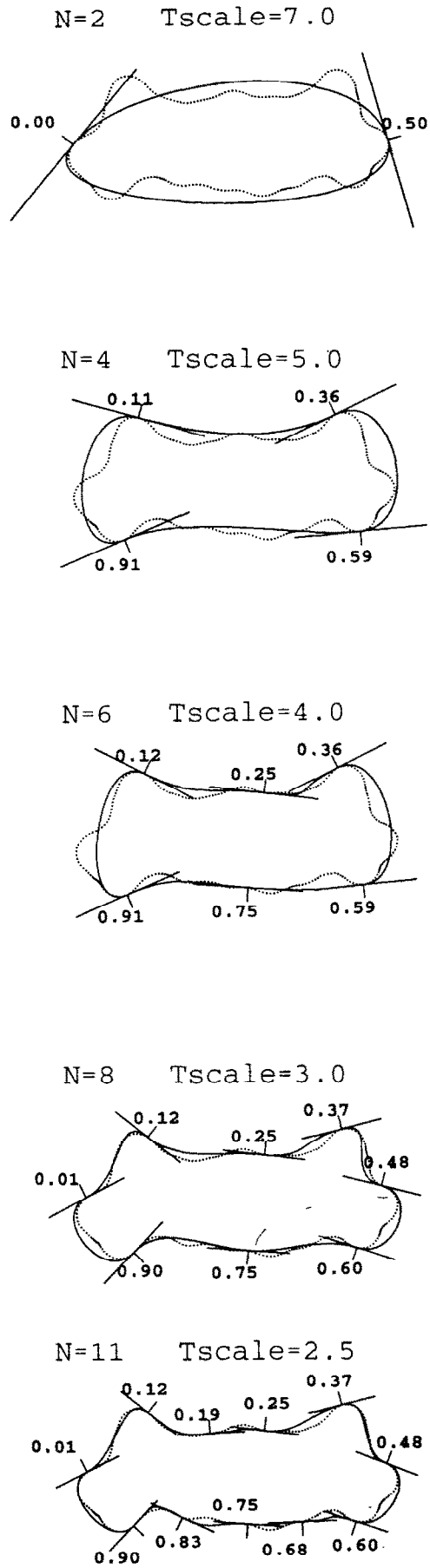

Figure 4. Approximation of B(u) with Bezier curves. Each illustration gives the number of curve segments (N) and the tangent scale (Tscale) used. The GC base pair is used for all examples. The curve B(u) is shown as a series of dots. The control points for the ends of a Bezier segment are taken on B(u) and are marked with the u-values by the normals. The central control points are at the ends of the scaled tangent lines. The resultant Bezier curves are shown as solid lines.

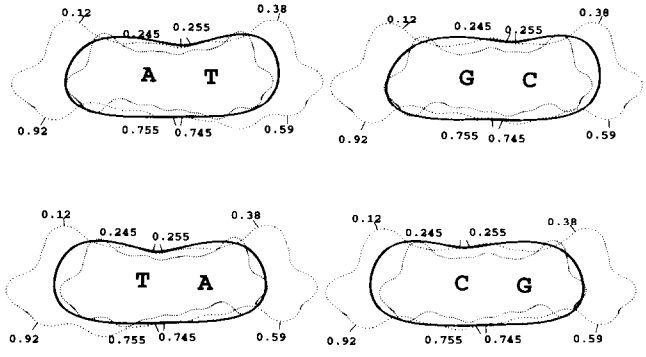

Figure 5. Approximation of B(u) with B-spline curves. Each illustration is a different base pair type with 8 curve segments. The original curve B(u) and the scaled curve B(u) are shown as a series of dots. The B(u) scale is 1.25, 1.5-fold about the short-long directions. All use the same u-values, marked on the scaled B(u). The resultant B-spline curve segments are shown as alternating, thick, black and gray lines.

The approximated base path spline curves specified by the same N u-values and scale factor for each of the four base pair types are similar. Figure 5 illustrates this for the B-spline case and the same is true for the Bezier case. Thus, only N values and a scale factor will be required to define the DNurbs geometry. The program currently sets a limit of $N_{max} = 20$, since 20 atoms, at most, contribute to the surface.

Control points for patches

The desired result is a surface patch roughly centered on the atomic surface for each base pair. Thus, control points are required between the planes of the neighboring stacked base pairs. (Recall that an array of 4 × 4 control points are needed to specify a bicubic patch.)

For B-spline DNurbs, each of the N corresponding u-values from adjacent base pairs are averaged, forming a "plane" of values between the base pairs that is roughly perpendicular to the helix axis. Each four successive planes provide the control points in the v direction, running along the length of the helix. For a particular base pair, i, control points are derived from the five adjacent pairs, $i \pm 2$. Therefore, the ends of the molecule require special attention. Two extra copies of the first and last base pair u-values are added.

The B-spline curves in the v direction are themselves helices. An additional scale factor to expand the control points is needed (see previous discussion concerning Figure 5.) For the u,v = B-spline,B-spline case, the overall non-uniform x,y scale for the $B(u)$ curves is set to 1.5,1.8. For the Bezier,B-spline case, a scale of 1.25,1.4 is used.

Implementation

The prototype version of the software was developed with IRIS Inventor, a three-dimensional, object-oriented graphics tool kit supplied by Silicon Graphics. This C++ programing environment provides a wide variety of geometric primitives and methods for rendering them, including such advanced features as true transparency and texture mapping. Figures 1–5 were created by auxiliary programs using Inventor with its PostScript output.

The coding was done in C++ on a low-end Indigo machine. The style is basically "C," with small portions done in true object-oriented fashion (e.g., $B(u)$ is implemented as a class). This is part of the *Ribbons++* software, an extension of the *Ribbons 2.0* program[12] currently under development. The prototype was shown in a poster session at the 1993 Molecular Graphics Society annual meeting.

The auxiliary programs are run to determine optimal placement of control points and scale factors. The program reads two PDB files, each containing a complementary strand of DNA. It then reads N, the number of patches per base pair, a key specifying whether Bezier or B-splines are to be used and a scale factor to control the placement of control points. Next, the program reads the N u-values that position the control points and the N integer keys setting the color and texture of each patch for the four types of base pairs. The base pairs are processed, with the ideal B-DNA coordinates of the appropriate $B(u)$ fit in a least-squares sense to the actual PDB coordinates. The resultant geometry can be imported/exported between other SGI Inventor and Explorer applications. Work on a more integrated control point, color and texture editor is in progress.

RESULTS

Figure 6 is a stereo line drawing of DNA as stick bonds with the patch boundary curves of the DNurbs. Bezier, B-spline patches with six patches per base pair are employed.

Figure 7 presents the rendering of a model of tRNA.[12] Transparent DNurbs are overlaid on the complementary regions using B-spline, B-spline patches with four patches per pair.

Color Plate 1 shows B-spline, B-spline DNurbs with eight patches per base pair. Color and texture are applied to illustrate the sequence and direction of the strands. Cutaway views give a feel for the fit to the underlying structure.

Color Plate 2 illustrates more textured DNurbs. A Bezier, B-spline with only two patches per base pair gives the geometry. Textures are applied to indicate the chemical properties of atoms in the major and minor groves. The eight required texture patterns are also shown. For comparison, a space-filling model is drawn to scale.

The time required for computation of the control points is trivial after the coordinates are read into memory. The images shown in the Color Plates cannot be rotated smoothly in real time except on high-end workstations. Table 2 gives rendering times in seconds for the boundary curves in Figure 6, the averages for the textured and nontextured DNurbs of

Table 2. Rendering performance. Time (seconds) required to draw the images. Only the Indigo^2 workstation is equipped with a special graphics board.

Workstation CPU, graphics	Space-filling spheres	Textured DNurbs	Nontextured DNurbs	Boundary line DNurbs
Indigo R3000, low-end graphics	15	7	2.2	0.2
Indigo R4000, XS24 graphics	6	3	0.8	0.0
Indigo^2 R4000, extreme graphics	1	2	0.2	0.0

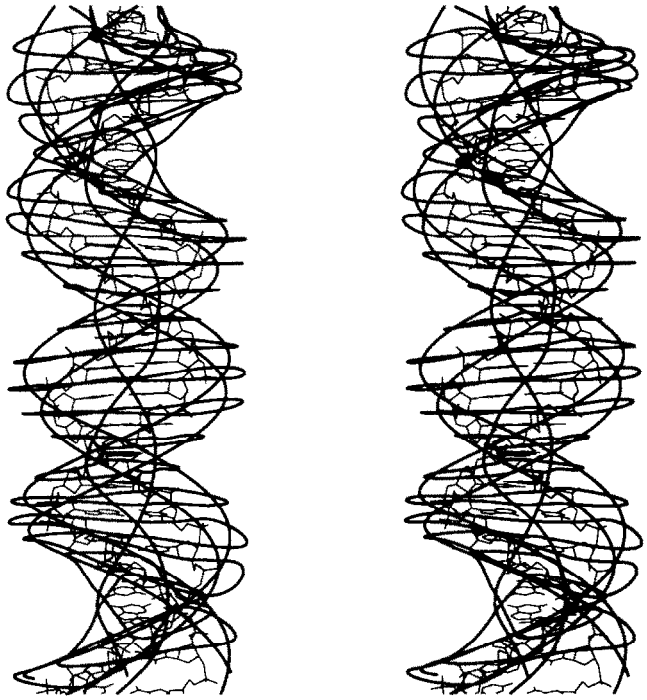

Figure 6. Comparison of DNurbs with the atomic model. A stereo view presents the atomic model as thin lines. The thick lines are the cubic curves on the edges of the bicubic patches. Bezier curves are employed around the base pairs. The u-values are the same for all base pairs (see the N = 6 example in Figure 4.) B-spline curves are used along the helix axis. The B(u) scale factors are 1.25,1.4.

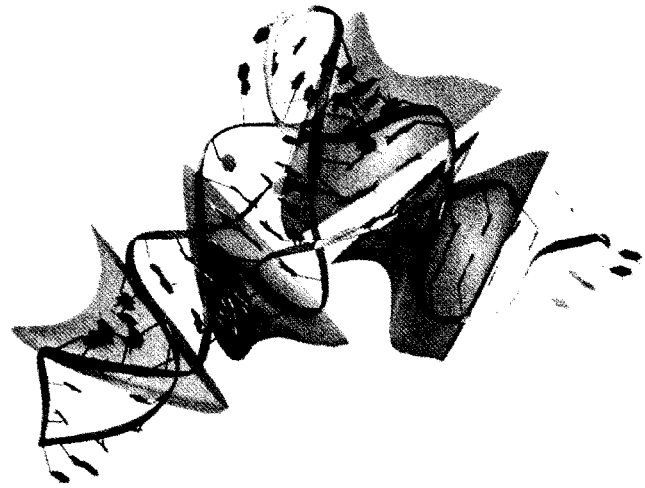

Figure 7. DNurbs with RNA. A tRNA model is rendered as a ribbon with polygonal bases connected by cylinders. Transparent DNurbs are overlaid on the four regions of complementary structure. B-spline curves are used for the base pairs, with N = 4 and the u-values = 0.12,0.38,0.59,0.92. The B(u) scale factors are increased to 1.8,2.1.

the Color Plates, and the atomic spheres of Color Plate 2. A value of zero means real-time transformation.

There appears to be a slight performance penalty in the Inventor environment compared to programs written directly in GL. Line drawings may be manipulated in near real time on even the lowest end workstation. This is the tack with *Ribbons 2.0* on Indigo machines—manipulate the image as a line drawing, then turn on the solid rendering. The Sphere and NurbSurface objects in the Inventor environment may be rendered at a user-chosen "complexity." The Color Plates have the complexity of the NurbSurface objects set very high, while the complexity of the Sphere objects are set to the default. The DNurbs still look quite smooth with the default values and the interactive response increases about threefold compared to the results given in Table 2.

DISCUSSION

The accessible surface of a macromolecule is a key determinant of its action. The definition[13] and computer implementation[14] of such surfaces have had a profound effect on subsequent visualizations of molecular form and function. Usually such surfaces are displayed on interactive graphics devices as dots[11] or triangles[15] and may be color coded to highlight chemical properties.

Splines have also been used to model molecular surfaces. Triangular surface patches were used to accurately model small molecule shapes.[16] Spline "contour lines" have been constructed from dot surfaces to represent the shapes of proteins.[17] A "sphere" of 64 patches collapsed onto the dot surface of a protein gave a smooth but "blobby" approximation of the shape (see "Methods"). The images are somewhat similar to those produced by modeling protein shape with spherical harmonics.[18]

This method seeks not so much an exact surface construction, but an attractive approximate surface representation described with a minimal number of parameters. As mentioned in a previous paper,[7] nucleic acid ribbons "could be sectioned to show such information as base type, base tilt-angle or sugar conformation." This method further allows the display of key DNA surface features—those involved in the "reading" of the DNA by enzymes and regulatory proteins.

B-DNA is about 20 Å in diameter, with very close to ten base pairs per helical repeat. The base pair planes are tilted about 6° with respect to the helix axis and the spacing between adjacent pairs is about 3.4 Å. The regularity of the structure allows one to infer the atomic positions from just the sequence or, alternatively, from a reduced atom representation used in simulations where only three "atoms" are used per base.[2] The latter case is fit by approximating the local helix axis.

The method is developed from ideal B-DNA. Although tRNA does not quite adopt this ideal configuration, the DNurbs based on the DNA fit rather well (Figure 7). Close inspection of Figure 7 reveals a small section in the acceptor stem where the ribbon is not enclosed by the DNurbs.

Both the Bezier and B-spline surface loop approximations are specified with N u-values. However, Bezier curves require three times the number of control points. Also, the Bezier curve segment connections are not as smooth (C^1 versus C^2 continuity.) The Bezier form can be used if greater fidelity to the actual surface tangent is desired. For a more faithful representation of the surface, the nonplanarity of the sugar rings in the minor groove should be taken into account. There are doubtless many variations on the theme of this work.

The data structures employed here could be used to create polygonal approximations such as triangular or quadrilateral meshes, but NURBS offer several advantages: a surface patch is specified with a small number of parameters, the NURBS are automatically decomposed into triangles, which are a staple of rendering systems and NURBS provide a particularly easy texture mapping. Such texturing represents the trend in evolving graphics standards.

The DNurbs offer a simplified, informative and aesthetically pleasing representation of complementary DNA. They may also offer a performance advantage on available hardware.

ACKNOWLEDGMENTS

Steve Harvey had suggested the idea of a "barber's pole" representation of DNA several years ago. Charlie Bugg and NASA grant NAGW-813 and NCI grant CA-13148 have provided continuing encouragement and support.

REFERENCES

1 Tan, R.K.-Z. and Harvey, S.C. Molecular mechanics model of supercoiled DNA. *J. Mol. Biol.* 1989, **205,** 573–591

2 Milner, J.D., Wood, Z.A., Byers, T.J., Harvey, S.C., Park, K., Griffith, J. and Hajduk, S.L. Characterization of an intrinsically bent fragment of *Bodo caudatus* kinetoplast DNA. (submitted to *Nucleic Acids Res.*)

3 Voet, D. and Voet, J.G. *Biochemistry;* John Wiley & Sons: New York, 1990.

4 Wing, R., Drew, H., Takano, T., Broka, C., Tanaka, S., Itakura, K. and Dickerson, R.E. Crystal structure analysis of a complete turn of B-DNA. *Nature* 1980, **287,** 755–758

5 Lesk, V.I. and Lesk, A.M. Schematic diagrams of nucleic acids and protein-nucleic acid complexes. *J. Appl. Cryst.* 1989, **22,** 569–571

6 Carson, M. and Bugg, C.E. Algorithm for ribbon models of proteins. *J. Mol. Graphics* 1986, **4,** 121–122

7 Carson, M. Ribbon models of macromolecules. *J. Mol. Graphics* 1987, **5,** 103–106

8 Carson, M. SigGraph. 1988, Unpublished results

9 Foley, J., van Dam, A., Feiner, S. and Hughes, J. *Computer Graphics;* Addison-Wesley: New York, 1990.

10 Bartels, R., Beatty, J. and Barsky, B. *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling;* Morgan Kaufmann: Los Altos, CA, 1987

11 Connoly, M.L. Solvent-accessible surfaces of proteins and nucleic acids. *Science* 1983, **221,** 709–713

12 Carson, M. Ribbons 2.0. *J. Appl. Cryst.* 1991, **24,** 958–961

13 Lee, B. and Richards, F.M. The interpretation of protein structures: Estimation of static accessibility. *J. Mol. Biol* 1971, **55,** 379–400

14 Greer, J. and Bush, B.L. Macromolecular shape and surface maps by solvent exclusion. *Proc. Nat. Acad. Sci. USA* 1978, **75,** 303–307

15 Connoly, M.L. Molecular surface triangulation. *J. Appl. Cryst.* 1985, **18,** 499–505

16 Klein, T.E., Huang, C.C., Petterrsen, E.F., Cough, G.S., Ferrin, T.E. and Langridge, R. A real-time malleable molecular surface. *J. Mol. Graphics* 1990, **8,** 16–24

17 Colloc'h, N. and Mornon, J-P. A new tool for the qualitative and quantitative analysis of protein surfaces using B-spline and density of surface neighbourhood. *J. Mol. Graphics* 1990, **8,** 133–140

18 Max, N.L. and Getzoff, E.D. Spherical harmonic molecular surfaces. *IEEE Comp. Graphics Appl.* 1988, **8** (4)42–50