CURVES+ USER GUIDE
V3.0, updated October 2016

Curves+ is a nucleic acid conformational analysis program that is applicable to a wide range of nucleic acid structures, including those with up to four strands and with either canonical or modified bases and backbones. The program is algorithmically simpler and computationally much faster than the earlier Curves approach, although it still provides both helical and backbone parameters, including a curvilinear axis and parameters relating the position of the bases to this axis, as well as the analysis of the local curvature of this axis, in terms of both its magnitude and its direction (register). It additionally provides a full analysis of groove widths and depths. Curves+ can also be used to analyze molecular dynamics trajectories. With the help of the accompanying program Canal, it is possible to produce a variety of graphical outputs including parameter variations along a given structure, time series and histograms of parameter variations during dynamics. Curves+ can also analyze the position of ions or other solvent or solute atoms in terms of curvilinear helicoidal coordinates with respect to the nucleic acid helical axis (see the Canion user guide).

References:

1) Conformational analysis of nucleic acids revisited: Curves+. R. Lavery, M. Moakher, J.H. Maddocks, D. Petkeviciute, K. Zakrzewka Nucleic Acids Res. 37 (2009) 5917
2) Curves+ web server for analyzing and visualizing the helical, backbone and groove parameters of nucleic acid structures. C. Blanchet, M. Pasi, K. Zakrzewska, R. Lavery Nucleic Acids Res. (web-server issue) 39 (2011) W68
3) Analyzing ion distributions around DNA. R. Lavery, J.H. Maddocks, M. Pasi, K. Zakrzewska Nucleic Acids Res. 42 (2014) 8130
4) Analyzing DNA curvature and its impact on the ionic environment: application to molecular dynamics simulations of minicircles. M. Pasi, K. Zakrzewska, J.H. Maddocks, R. Lavery (2016) submitted.

Compiling

Curves+ is distributed as Fortran code. The distribution includes a Makefile that uses the gfortran compiler. If you have this compiler on your system, you can simply type "make" to generate the executable file Cur+, otherwise you will have to modify the Makefile. The main vector and matrix dimensions of the code are contained in curves_data.inc. The program is currently set up to treat a maximum of 500 nucleotides, 15,000 heavy atoms and 100 ions (or ligands). You may need to change these limits in some cases (e.g. analyzing solvent water molecules). In this case, change the Makefile to compile with checks for overflowing dimensions (exchange the comment symbol "#" on the FFLAG lines of the Makefile) – note that compiling with this option will slow down the execution.

In addition to molecular dynamics trajectories in Amber .trj format, Curves+ can now optionally read Amber netCDF and Gromacs XTC formats. If you require one or both of these options, you will need to uncomment the appropriate sections within the Makefile. In the case of the netCDF option, you will need to have either netcdf-fortran (see http://www.unidata.ucar.edu/downloads/netcdf) or AmberTools (from Amber 12 or later, see http://ambermd.org/) already installed on your system.

Curves+ output can be analyzed directly when single structures are treated, but when it is used to analyze molecular dynamics trajectories, you will also need to use the companion program Canal - see the Canal user guide.

Input data

The example below shows a simple input file. In this example, the compiled program is in the directory /Users/RL/Code. The notation <<! allows the input data to be placed immediately after the line which calls the program. The input is then ended by the explanation mark. The example analyzes a duplex DNA from the file 1bna.pdb.

The initial input is in namelist format (beginning **&inp** and ending **&end**. See below for possible namelist variables. In the example, we give the input file name (the extension .pdb is assumed by default), the output file (r+bdna.lis, in this case it will be created in the /Code subdirectory) and the library file lib (base and backbone reference geometries will be taken from the files standard_b.lib and standard_s.lib, again in the /Code subdirectory).

```
 &inp file=1bna, lis=r+bdna,
 lib=/Users/RL/Code/standard, &end
```

The next line gives the number of strands, followed by the number of nucleotides in each strand (up to four strands can be analyzed). For each strand, the sign indicates whether the nucleotides are in the direction 5'-3' (positive) or 3'-5' (negative).

The following two lines give the numbers corresponding to the order of the subunits containing the nucleotides which constitute each strand (1=1st, 2=2nd, ...). Note that proteins, water (recognized as unit names WAT, H2O or HOH) and any HETATM lines are automatically removed at input. The only exceptions to this rule are the solvent molecules specified by the option **sol**, or protein atoms (recognized by their amino acid subunit name) listed in the standard_i.lib file when the option **ions** is set to true.

Contiguous series of nucleotide numbers can be indicated using a colon (I:J ≡ I, I+1, i+2, ... J-1, J).

You can choose which nucleotides you wish to analyze. You are not obliged to analyze the complete nucleic acid fragment contained in the input file. Note also that bases can be excluded from axis calculations by including them in the I/P lines as negative numbers (useful for flipped out bases or abasic nucleotides) and missing nucleotides within a strand (gaps) are indicated by zeros.

Nucleotide numbers in each strand should be organized so that paired bases occur in identical positions (we term these positions "levels", typically a level will be equivalent to a base pair). For unpaired bases (which still constitute a level), you can use zeros for padding so that paired bases are correctly grouped in the same level (see the second example below).

Because the only subunits that are read from the input file are nucleotides, numbering is usually easy. For example, a canonical dodecamer would be:

```
2 1 -1 0 0
1:12
24:13
```

- even if it was complexed with a protein and the protein came before the nucleic acid in the input file.

Similarly, a decamer with unpaired 5'-terminal nucleotides (i.e. 11 nucleotides per strand) would be:

```
2 1 -1 0 0
1:11 0
0 22:12
```

- here the zeros specify that base pairs involve nucleotides 2-22, 3-21, ... 11:13. Note that if you want to ignore the hanging nucleotides, you can consider the input as a simple decamer:

2 1 -1 0 0
2:11
22:13

Note that at there must be at least one base (i.e. one positive number) at each level analyzed.

Curves+ namelist variables

CHARACTER (strings without quotes, maximum length 128 characters):

file: file name for input structure (.pdb and .mac extensions need not be given in input, use .trj, .nc or .xtc for MD trajectories)

ftop: name (with extension) of file with topological data (AMBER or PDB) for MD trajectory analysis

lis: root file name for all output (.lis, .cda, .cdi, .cdl, .fra, _X.pdb, _B.pdb, _C.pdb)

lib: root file name for base (_b.lib) and backbone (_s.lib) geometry files

ibld: name (.cdi extension assumed) of ion coordinates for reconstruction

sol: name of solute molecule in input .pdb file (or Amber topology file) to be analyzed if ions=.t.

back (P): atom used to define backbone. Different backbones can use different atoms if necessary (e.g. P/C5* - a slash is used to separate input names).

REAL DATA (default value of each variable is given):

wback (2.9Å) radius of backbone around spline through chosen backbone atoms (by default the phosphorus atoms) for groove width calculations

wbase (3.5Å) average half-width of base pairs for groove depth calculations

rvfac (7.5) factor multiplying the length of the radial vectors indicating curvature in the _r.pdb graphical ouput file

INTEGER DATA (default value of each variable is given):

isym (1): symmetry repeat for generating helical axis (1 = mononucleotide, 2 = dinucleotide e.g. ATATAT..., any positive value is allowed)

itst (0): first snapshot to analyze (if reading MD trajectory)

itnd (0): last snapshot to analyze (if reading MD trajectory, itst=itnd=0 implies full range of snapshots)

itdel (1): spacing of snapshots to analyze

itbkt (0): if itbkt is > 0 then read itbkt pairs of values itst, itnd (one per line) immediately following the namelist input. This enables chosen intervals of a trajectory to be analyzed

naxlim(0): if naxliim > 0, calculates overall bend skipping naxlim base levels at the 5'- and 3'-ends. This may give a more meaningful overall bend if the ends of an oligomer are irregular (e.g. suffer from base pair fraying)

LOGICAL SWITCHES (enter as .t. or .f., the default value of each variable is given):

circ (.f.): .t. implies a closed circular nucleic acid

line (.f.): if .t. find the best linear helical axis

zaxe (.f.): if .t. use the Cartesian Z-axis as the helical axis

fit (.t.): if .t. fit a standard bases to the input coordinates (important for MD snapshots to avoid base distortions leading to noisy helical parameters)

test (.f.): if .t. provide additional output in .lis file on base least squares fitting and details of the helical axis reference points and vectors per base level

ions (.f.): if .t. curvilinear helicoidal coordinate analysis of ions (or other solvent/solute atoms) is carried out (see Canion user guide)

refo (.f.): if .t. use the old Curves convention for defining the reference frame of each base, rather than the more recent Tsukuba convention

axfrm (.f.): if .t. generates closely spaced helical axis frames as input for Canal and Canion

frames (.f.): if .t. outputs base frames in the *lis*.fra file (as nearest integer to value x 1000)

Input files

STRUCTURE

Curves+ can analyze single structures from .pdb or .mac files: (the latter is only of interest for users of JUMNA) or a series of snapshots from an MD simulation (an AMBER .trj or netCDF file, or a GROMACS XTC file – the latter two options requiring modifications of the Makefile before compilation, see above). Analyzing a trajectory also requires reading the molecular topology that defines the atom types and the order of atoms that will be found in the trajectory input. This information can be provided as an AMBER top file, or as a single PDB file (extension .pdb). It is also possible to analyze a PDB file containing several conformations of the same molecular system (each bracketed by the PDB code words MODEL and ENDMDL). In this case, the input file should have the extension .pgp, otherwise only the first MODEL will be read.

Note that only ATOM lines are used in .pdb files (unless solute molecules are analyzed, in which case HETATM lines are also used, e.g. by putting **sol**=H2O in the namelist input).

Remember that PDB unit numbers are NOT used in Curves+ input (see Section "Input data"):

- the numbers in the input refer to order in which the nucleic acid subunits appear in the .pdb file (i.e. 1,2,3 implies 1st, 2nd, and 3rd subunits encountered in the input file) – whether or not other molecules appear before the nucleic acid.

LIBRARY FILES (note that all .lib files can be edited by the user to suit individual needs)

standard_b.lib contains the standard geometry for the bases which can ben analyzed. Below is the data for cytosine:

```
C Y 7 'Cytosine'
 1.94866 -1.45161 -0.19029 'C1*'
 0.74385 -0.58352 -0.07229 'N1'
 0.93020  0.79520 -0.12929 'C2'
-0.15547  1.60399 -0.02329 'N3'
-1.37969  1.08626  0.13371 'C4'
-1.59254 -0.32937  0.19471 'C5'
-0.49500 -1.12092  0.08671 'C6'
```

The first line gives the one-letter code for the base, R/Y specifying purine or pyrimidine, the number of ring atoms (plus the sugar C1*) and the full base name (≤ 8 charas.). The following lines give the x,y,z coordinates of each atom and the atom name. The first three atoms must be C1*, the base atom bound to C1* and the atom used to define the base normal using the vector product (1-2)x(3-2). This data is used for least squares fitting of standard base geometries. The base atoms can be specified in any order.

standard_s.lib contains the description of the backbone geometry to be analyzed. Standard lines start with a blank and define a single torsion by giving 4 atom names (≤ 4 chars) followed by the name of the torsion name (≤ 6 chars). Lines starting with 'B' define torsions which depend on the type of base: atoms 1-4 are used with purines and atoms 5-8 are used with pyrimidines

Lines starting with 'S' define sugar rings and give the names of the 5 ring atoms in the order used for pseudorotation calculations. the data below is for standard DNA. Non-standard bases or backbones can be analyzed by modifying the .lib files.

```
 '-O3*'  'P'   'O5*' 'C5*'  'Alpha'
  'P'   'O5*' 'C5*' 'C4*'  'Beta'
  'O5*'  'C5*' 'C4*' 'C3*'  'Gamma'
  'C5*'  'C4*' 'C3*' 'O3*'  'Delta'
  'C4*'  'C3*' 'O3*' '+P'   'Epsil'
  'C3*'  'O3*' '+P'  '+O5*' 'Zeta'
B 'O4*'  'C1*' 'N9'  'C4'  'O4*'  'C1*'
            ... 'N1'  'C2'   'Chi'
S 'C1*' 'C2*' 'C3*'  'C4*' 'O4*'
```

standard_i.lib contains the description of the ions, or atoms belonging to ligands, proteins or solute molecules (nucleic acid) to be analyzed. Each line gives the name of the ion or atom in single quotes followed by its formal charge (which can be used in Canion for choosing positive, neutral or negative ions/solute atoms).

```
'Na+' 1
'K+' 1
'P' -1
```

Output files

1) The .lis output from Curves+ lists the input parameters, the base sequence of each strand, then: (A) base pair-axis parameters; (B) intra-base pair parameters; (C) inter-base pair parameters: (D) backbone parameters, (E) groove parameters and (F) local curvature and register (curvature is given as 40/r where r is the local radius of curvature and register indicates the direction of curvature with respect to the base pair reference frame, using the vector pointing towards the strand whose 5'-3' direction is aligned with the helical axis, 90° therefore indicates bending into the minor groove and -90° bending into the major groove). In section (C), the first six parameters, including rise and twist correspond to the transformation between successive base pairs. H-Ris and H-Twi correspond to the translation and rotation of successive base pairs along and around the helical axis. With the base pair-axis parameters, H-Ris and H-Twi are useful for understanding the overall helical structure of the fragment analyzed.

2) If axfrm=.t., a set of Cartesian coordinate frames describing the helical axis is output in a .afr file that is used by Canion in performing ion distribution analyses. This option is only used for a single nucleic acid structure, either an experimental structure, or the average structure from an MD trajectory.

3) If ions=.t. the curvilinear helicoidal coordinates of the ions are output in the (I) section of the .lis output, as well as in a .cdi file that can subsequently be used by Canion to analyze their distribution.

Graphic output:

1) The _X.pdb file containing the helical axis
2) The _B.pdb file showing the backbone splines and the vectors defining groove widths.
3) The _R.pdb file showing the radial vectors that indicate the direction and magnitude of local curvature. The magnitude of curvature is scaled so that 1.0 corresponds roughly to the curvature of DNA wrapped around the histone core of a nucleosome. The length of the vectors in the _R.pdb file is determined by multiplying the scaled curvature by the namelist variable rvfac.

Analyzing snapshots from an MD trajectory (which requires both trajectory and topology I/P files) suppresses the parameter output in the .lis output. Either Canal (for structure, using the .cda file) or Canion (for ions/solute molecules, using the .cdi file) should then be used for parameter analysis.

Sample output for 1bna.pdb

```
    **************************************               **************
    **** CURVES+ Version 3.0nc 09/2016 ***               *  25 Oct 16 *
    **************************************               **************


 FILE : 1bna.pdb                          ftop :
 LIS  : test                              LIB  : standard
 ibld :                                   sol  :
 back : P

 wback :  2.90  wbase :  3.50  rvfac :  7.50

 isym  :     1  itst  :     0  itnd  :     0  itdel :     1  itbkt :     0
 naxlim:     0

 circ  :     F  line  :     F  zaxe  :     F  fit   :     T  test  :     F
 IONS  :     T  refo  :     F  axfrm :     F  frames:     F


 LS fitting of standard bases ...RMS max =  0.051

 Strands =    2 Atoms =    486 Units =    24

 Combined strands have   12 levels ...

 Strand  1 has  12 bases (5'-3'): CGCGAATTCGCG
 Strand  2 has  12 bases (3'-5'): GCGCTTAAGCGC

 (A) BP-Axis        Xdisp   Ydisp   Inclin    Tip  Ax-bend

   1) C   1-G  24    0.15    0.08     6.1     1.2     ---
   2) G   2-C  23    0.02   -0.01     4.4     3.1     0.4
   3) C   3-G  22    0.43   -0.36     4.7    -5.8     0.5
   4) G   4-C  21   -0.18    0.02     6.4     0.2     0.5
   5) A   5-T  20   -0.33   -0.02     1.8    -1.3     1.0
   6) A   6-T  19   -0.37    0.06    -1.0    -1.3     1.0
   7) T   7-A  18   -0.02    0.02    -1.7    -4.2     1.0
   8) T   8-A  17   -0.07    0.26    -1.3    -2.6     1.0
   9) C   9-G  16    0.36    0.17    -2.0    -1.7     0.7
  10) G  10-C  15    1.45    0.64    -3.9     5.6     0.7
  11) C  11-G  14    0.52    0.34    -8.3    -3.7     0.3
  12) G  12-C  13    1.25    0.09    -6.6    -1.9     0.3

      Average:       0.27    0.11    -0.1    -1.0  Total bend =     6.8 (  1 to  12)

 (B) Intra-BP parameters

 Strands 1-2       Shear  Stretch Stagger  Buckle  Propel Opening

   1) C   1-G  24   -0.61   -0.20    0.09     3.8   -14.7    -2.7
   2) G   2-C  23    0.11   -0.18    0.26    -4.5   -10.9    -2.9
   3) C   3-G  22   -0.15   -0.17    0.22    -7.6    -4.0    -0.9
   4) G   4-C  21   -0.22   -0.35   -0.12    10.1   -11.7     0.3
   5) A   5-T  20    0.28   -0.21    0.14     4.8   -18.3     3.7
   6) A   6-T  19   -0.09   -0.03    0.28     3.4   -20.2     7.4
   7) T   7-A  18    0.32   -0.10    0.23     0.7   -19.3    10.2
   8) T   8-A  17    0.23   -0.20    0.02    -1.8   -19.8     3.0
   9) C   9-G  16   -0.17   -0.16    0.04   -10.8   -19.3     0.5
  10) G  10-C  15    0.23   -0.20    0.32     2.4    -6.2     0.8
  11) C  11-G  14   -0.06   -0.21    0.67    -4.0   -19.7    -4.7
  12) G  12-C  13   -0.36   -0.02    0.32     7.0     0.6    -2.9

      Average:      -0.04   -0.17    0.21     0.3   -13.6     1.0


 (C) Inter-BP        Shift   Slide    Rise    Tilt    Roll   Twist   H-Ris   H-Twi

   1) C   1/G   2   -0.39    0.27    3.54    -3.5     6.1    42.5    3.57    42.9
   2) G   2/C   3    0.52    0.18    3.54     1.0    -5.3    36.1    3.55    35.9
   3) C   3/G   4   -0.32    0.78    3.00     3.2     9.1    26.7    3.04    27.3
   4) G   4/A   5   -0.00    0.07    3.38    -3.3     2.1    40.0    3.37    40.1
   5) A   5/A   6    0.11   -0.31    3.31    -0.8     0.4    35.3    3.31    35.3
   6) A   6/T   7    0.35   -0.61    3.34     2.0    -3.5    34.4    3.36    34.6
   7) T   7/T   8   -0.28   -0.17    3.33     3.2    -0.0    35.2    3.31    35.3
   8) T   8/C   9    0.02   -0.07    3.39     0.9    -0.9    38.6    3.39    38.6
   9) C   9/G  10    0.39    0.93    3.21    -2.9     4.8    31.6    3.12    31.5
  10) G  10/C  11   -1.36    0.31    3.71    -5.0   -13.5    38.5    3.67    39.2
  11) C  11/G  12    0.79    0.09    3.22     3.1    -2.9    34.6    3.21    34.8

      Average:      -0.02    0.13    3.36    -0.2    -0.3    35.8    3.35    36.0

 (D) Backbone Parameters

  Strand 1     Alpha  Beta   Gamma  Delta  Epsil  Zeta   Chi    Phase  Ampli  Puckr

   1) C   1     ----   ----  174.2  156.8 -141.3 -143.9 -105.0  161.6   56.6  C2'en
   2) G   2    -65.6  169.8   40.1  128.1  174.2  -97.8 -110.5  139.8   42.1  C1'ex
   3) C   3    -62.6  171.8   58.8   98.3 -176.7  -87.6 -135.1   92.8   38.5  O1'en
   4) G   4    -62.9 -179.9   57.2  155.7 -155.3 -152.5  -93.4  166.6   49.8  C2'en
   5) A   5    -43.0  142.8   52.4  119.6  179.9  -92.2 -126.3  128.8   46.8  C1'ex
   6) A   6    -73.3  179.7   66.0  121.1  173.7  -88.5 -122.2  127.3   50.2  C1'ex
   7) T   7    -56.6 -179.2   52.2   98.9  173.6  -85.9 -127.3  101.5   47.6  O1'en
   8) T   8    -59.2  173.4   64.1  108.9  170.6  -89.3 -125.7  115.9   49.7  C1'ex
   9) C   9    -58.5 -179.5   60.5  128.7 -156.9  -94.0 -119.5  140.7   46.9  C1'ex
  10) G  10    -67.3  169.1   47.2  142.9 -103.3  150.2  -89.6  146.5   54.7  C2'en
  11) C  11    -73.9  139.3   56.3  135.7 -161.8  -89.6 -125.1  147.7   47.7  C2'en
  12) G  12    -81.5  175.7   57.2  110.7   ----   ---- -112.0  114.1   52.1  C1'ex

  Strand 2     Alpha  Beta   Gamma  Delta  Epsil  Zeta   Chi    Phase  Ampli  Puckr

   1) G  24    -65.0  170.6   46.6   78.7   ----   ---- -135.2   34.2   46.4  C3'en
   2) C  23    -72.2  138.5   44.6  112.8 -174.4  -96.8 -125.3  117.3   44.6  C1'ex
   3) G  22    -66.8  179.1   50.2  149.7 -100.1  171.6  -88.4  156.8   52.3  C2'en
   4) C  21    -59.1 -175.4   45.0  110.3 -176.7  -86.5 -114.3  113.9   43.3  C1'ex
   5) T  20    -58.6  179.5   55.3  122.4  178.5  -94.5 -120.5  129.9   50.5  C1'ex
   6) T  19    -58.3  173.6   60.0  109.2  178.8  -88.3 -131.3  116.7   47.9  C1'ex
   7) A  18    -57.1 -173.6   47.7  130.2  174.4 -101.3 -108.3  147.6   43.0  C2'en
   8) A  17    -56.6 -169.5   53.8  146.6  176.9  -97.1 -106.4  169.4   43.1  C2'en
   9) G  16    -69.2  171.1   73.2  135.9  174.1  -98.4 -114.8  149.5   41.5  C2'en
  10) C  15    -63.0  168.8   60.4   85.7  174.8  -85.5 -133.8   67.5   44.1  C4'ex
  11) G  14    -51.3  163.9   49.0  121.9  177.7  -93.0 -116.4  128.5   45.4  C1'ex
  12) C  13     ----   ----   55.9  136.7 -158.6 -124.9 -127.6  153.5   43.7  C2'en

 (E) Groove parameters

  Level           W12     D12     W21     D21

   1.5
   2.0  G   2
   2.5
   3.0  C   3
   3.5           8.1     5.1
   4.0  G   4     7.2     4.9    11.4     5.0
   4.5           6.2     5.1    11.5     5.1
   5.0  A   5     5.2     5.1    11.5     5.2
   5.5           4.6     5.1    11.2     5.7
   6.0  A   6     4.2     4.9    10.5     5.4
   6.5           4.1     5.2    10.4     5.0
   7.0  T   7     4.0     5.4    10.6     4.9
   7.5           3.3     5.6    11.6     4.5
   8.0  T   8     3.1     5.6    12.1     3.5
   8.5           4.1     5.5    12.4     1.8
   9.0  C   9     5.2     5.7    12.4     2.9
   9.5
  10.0  G  10
  10.5
  11.0  C  11
  11.5

 (F) Curvature analysis

    N    BP step        Cur       Rad       Reg
 ---------------------------------------------
   2) G   2/C   3      0.323  123.991  -76.077
   3) C   3/G   4      0.257  155.398 -106.005
   4) G   4/A   5      0.366  109.312 -149.701
   5) A   5/A   6      0.867   46.147 -171.414
   6) A   6/T   7      0.549   72.831  167.453
   7) T   7/T   8      0.190  211.009   62.570
   8) T   8/C   9      0.209  191.772 -131.858
   9) C   9/G  10      0.417   95.978  174.935
  10) G  10/C  11      0.122  327.071  135.795

 <curvature> =   0.367 std. dev. =   0.215
```

```
 (I)  46 ions input (  0+    0++  22-   0-- )

 Located      I-Axe   I-Dis   I-Ang

  1) C1*   1    1.15    5.97    17.6
  2) P     2    1.64    9.18    11.0
  3) C1*   2    2.07    5.85    20.2
  4) P     3    2.68    9.26     5.3
  5) C1*   3    2.95    5.37    24.2
  6) P     4    3.82    8.93     6.1
  7) C1*   4    4.22    5.86    20.3
  8) P     5    4.81    9.14     7.7
  9) C1*   5    4.97    5.91    29.5
 10) P     6    5.49    9.54    22.9
 11) C1*   6    5.89    6.15    35.3
 12) P     7    6.37    9.77    28.9
 13) C1*   7    6.80    5.74    35.9
 14) P     8    7.28    9.70    28.0
 15) C1*   8    7.76    6.00    34.0
 16) P     9    8.22    9.63    28.4
 17) C1*   9    8.64    5.86    36.7
 18) P    10    8.90    9.75    38.8
 19) C1*  10   10.01    5.91     9.3
 20) P    11   10.40    9.12    13.0
 21) C1*  11   10.66    5.94    29.7
 22) P    12   10.86    9.54    37.4
 23) C1*  12   11.93    5.70    15.5
 24) P    14   11.76    8.86   162.8
 25) C1*  14   11.18    5.37   155.7
 26) P    15   10.71    8.98   163.6
 27) C1*  15   10.20    4.76   158.8
 28) P    16    9.45    8.48   174.8
 29) C1*  16    9.01    5.54   157.9
 30) P    17    8.38    8.76   167.8
 31) C1*  17    8.11    5.67   146.2
 32) P    18    7.67    9.32   151.4
 33) C1*  18    7.11    5.91   144.7
 34) P    19    6.59    9.62   153.5
 35) C1*  19    6.17    5.93   142.8
 36) P    20    5.58    9.73   153.9
 37) C1*  20    5.11    6.05   145.4
 38) P    21    4.62    9.57   154.5
 39) C1*  21    4.08    5.78   150.9
 40) P    22    3.45    9.70   164.9
 41) C1*  22    2.70    5.87   171.9
 42) P    23    2.17    9.27   176.6
 43) C1*  23    1.89    5.86   160.2
 44) P    24    1.15    9.28  -176.9
```

Notes

1) The analysis of axis curvature is presented in section “(F)” with the curvature, the radius of curvature and the register (the orientation of curvature) for each base pair step. For details see the corresponding publication listed on page 1.

2) This analysis has the option ions=.t. and both "P3 and "CI*" atoms are included in the standard_i.lib file. Consequently, their curvilinear helicoidal coordinates appear in this listing (section I).

Curves+ program subunits

aacur	main program
axis	generate helical axis frame at each base level
axref	output axis frames if test = .t.
backbo	calculate backbone torsions and sugar puckers
bisection	bisection algorithm used in determining ion positions
curpar	calculate magnitude and direction of local curvature
dotdelta	dot product used by bisection algorithm
eigen	eigenvalue calculation used for fitting standard base geometries
findaxis	locate screw axis to transform one helical axis frame to the next
input	read input conformation (.pdb or .mac)
intaxe	use screw axis to generate helical axis frames between base levels
intop	reads Amber topology file
ionbld	reconstructs ion positions using a .cdi file as input
ionpar	calculate helicoidal ion coordinates
locate	find base and backbone atoms and generate base reference frames
lsfit	least-squares fit of bases to standard geometries
manta	groove geometry calculation
ncerror	error routine for netCDF input
nml	parsing namelist input
params	calculation of helical parameters
pdbout	output of .pdb format files
screw	apply a screw axis transformation on a helical axis reference frame
setup	read library files for bases (and optionally, ions and ligands)
smooth	polynomial smoothing of helical axis frames at base levels
torp	calculate torsion angles
xtcerror	error routine for XTC input