

Supplementary material for ‘Drawing and editing the secondary structure(s) of RNA’: Command-lines and code listings

Yann Ponty*

CNRS, LIX – UMR 7161 & INRIA AMIB. Ecole Polytechnique, Palaiseau, France

Fabrice Leclerc

CNRS, IGM – UMR 8621, Université Paris Sud, Orsay, France

Contents

1	Running VARNA (Linear layout → SVG output)	2
2	Running R-chie (Linear layout → PDF output)	2
3	Running VARNA (Circular layout → PostScript output)	2
4	Running RNAView to extract a 2D structure from a 3D model (PDB → RNAML)	2
5	Running VARNA on the extended secondary structure (RNAML → Vector graphics)	3
6	Running VARNA (RNAML → VARNA session file)	3
7	Running VARNA (NAVview algorithm → PostScript)	3
8	Running VARNA to draw the shadow of a secondary structure	3
9	Running RNAPlot (NAVview algorithm → PostScript)	3
10	Running RNAPlot (Radial algorithm)	4
11	Running RNAPlot with additional Postscript annotations	4
12	Using RNAPlot to display probing values as a color map	4
13	Script for formatting accessibility values as a color map using RNAPlot	5
14	Script for preprocessing a Stockholm file as input accepted by R2R	5
15	Running R2R to precompute conservation levels	5

*To whom correspondence should be addressed (yann.ponty@lix.polytechnique.fr)

16 Simple input <code>meta</code> file for <code>R2R</code>	6
17 Running <code>R2R</code> (<code>Stockholm/meta</code> → <code>SVG/PDF</code>)	6
18 Running <code>Rchie</code> (<code>Fasta</code> multiple sequence alignment+ <code>DBN</code> → <code>PDF</code>)	6
19 Script to generate a tree representation using <code>GraphViz</code> (<code>DBN</code> → <code>DOT</code>)	6
20 Another simple input <code>meta</code> file for <code>R2R</code>	6
21 Advanced input <code>meta</code> file for <code>R2R</code>	7
22 Sample <code>Stockholm</code> file for interacting RNAs, usable as an input for <code>R2R</code>	7
23 Sample <code>Stockholm</code> file for a single RNA, usable as an input for <code>R2R</code>	8
24 Shell script demonstrating advanced usage of <code>R2R</code>	11
25 Shell script for the generation of simplified representations using <code>VARNA</code>	11
26 Shell script for using color maps and algorithms using <code>VARNA</code>	11

1 Running `VARNA` (Linear layout → `SVG` output)

```
# Running VARNA (Linear layout/SVG output)
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd
-i XXXX.yyy -o out.svg -algorithm line
```

2 Running `R-chie` (Linear layout → `PDF` output)

```
# Running R-chie (Linear layout/PDF output)
Rscript rchie.R --format1 "vienna" --pdf --output=out.pdf XXX.yyy
```

3 Running `VARNA` (Circular layout → `PostScript` output)

```
# Running VARNA (Circular layout/PostScript output)
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd
-i XXXX.yyy -o out.eps -algorithm circular
```

4 Running `RNAView` to extract a 2D structure from a 3D model (`PDB` → `RNAML`)

```
# Running RNAView on RNA 3D model (PDB format)
rnaview -p XXXX.pdb
```

5 Running VARNA on the extended secondary structure (RNAML → Vector graphics)

```
# Running VARNA on the ext. sec. str. (RNAML -> Vector graphics)
java -cp VARNAvx-Y.jar fr.orsay.lri.varna.applications.VARNAcmd
    -i XXXX.pdb.xml -o YYYY.svg
```

6 Running VARNA (RNAML → VARNA session file)

```
# Running \Software{VARNA} (RNAML -> VARNA session file)
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd
    -i XXXX.pdb.xml -o YYYY.varna
```

7 Running VARNA (NAVview algorithm → PostScript)

```
# Running VARNA (NAVview algorithm -> PostScript output)
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd
    -i XXXX.pdb.xml -o YYYY.eps -algorithm naview
```

8 Running VARNA to draw the shadow of a secondary structure

```
# Running VARNA (NAVview algorithm -> PostScript output)
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd -i XXXX -o YYYY
    -flat true -highlightRegion "1-ZZZ:fill=#AOAOAO,outline=#AOAOAO,radius=10"
    -drawBases false -backbone "#AOAOAO" -fillBases "#AOAOAO" -bpStyle simple
    -numPeriod 50
```

9 Running RNAPlot (NAVview algorithm → PostScript)

```
# Running RNAPlot (NAVview algorithm)
RNAPlot < xxxx.yyy
```

10 Running RNAPlot (Radial algorithm)

```
# Running RNAPlot (Radial algorithm)
RNAPlot --layout-type=0 < xxxx.yyy
```

11 Running RNAPlot with additional Postscript annotations

```
# Running RNAPlot from a DBN file 'XXX.txt'
# Marking nucleotide 10
RNAPlot --post "10 cmark" < XXX.txt
# Drawing backbone for region (10,15) using red color (r=1,g=b=0) and line thickness 2
RNAPlot --pre "10 15 2 1. 0. 0. omark" < XXX.txt
# Defining a custom PS macro and using it to fill base number 10 in blue (r=g=0, b=1)
RNAPlot --pre "/cfmark {setrgbcolor newpath 1 sub coor exch get aload pop fsize 2 div 0 360 arc fill} bind
def 10 0. 0. 1. cfmark" < XXX.txt
```

12 Using RNAPlot to display probing values as a color map

Listing 1: Using RNAPlot to display probing values as a color map.

```
import os,sys
def getColor(val, minval, maxval,col1=(1.,1.,1.),col2=(0.0,0.8,0.0)):
    (r1,g1,b1),(r2,g2,b2) = col1,col2
    span = float(maxval)-float(minval)
    k = (float(val)-float(minval))/(span if span!=0 else 1.0)
    l = 1.-k
    return (r1*l+r2*k,g1*l+g2*k,b1*l+b2*k)
def formatValuesAsPS(values):
    minval,maxval = min(values),max(values)
    valtab = [" %s %s cfmark"%(i+1,"%2f %.2f %.2f"%getColor(v,minval,maxval)) for i,v
              in enumerate(values)]
    return "\n".join(valtab)
# Invokes RNAPlot to display a color map
# Arg1: path to DBN file
# Arg2: list of comma-separated values (eg "1,2,3")
if __name__ == "__main__":
    inputFile = sys.argv[1]
    values = sys.argv[2].split(",")
    extraMacro = "/cfmark {setrgbcolor newpath 1 sub coor exch get aload pop fsize 2
                 div 0 360 arc fill} bind def"
    os.system("RNAPlot --pre \"%s %s\" < %s"%(extraMacro,formatValuesAsPS(values),
                                                inputFile))
```



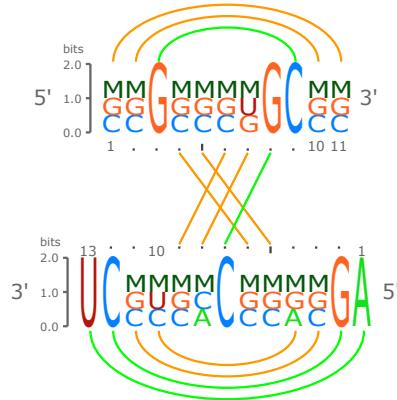
13 Script for formatting accessibility values as a color map using [RNAPlot](#)

Listing 2: Fasta-like input file XXXX.txt expected by RILogo

```
>organism1
GCGGGGUGCGC&AGGACCCACUCCU
>organism2
CGGCCCGGCCG&AGCGGGCCGCGCU
>structure
(((AABB))&(((aabbb))))
```

Listing 3: Invoking RILogo

```
rilogo XXXX.txt > output.svg
```



14 Script for preprocessing a [Stockholm](#) file as input accepted by [R2R](#)

Listing 4: Transforming an Stockholm-formatted RFAM multiple sequence alignment into an unblocked alignment accepted by R2R as input.

```
import os,sys,string
def unblockStockholm(inpath,outpath):
    outfile = open(outpath,"w")
    (seqIDs,seqContents) = ([],{})
    for l in open(inpath):
        if len(l)>12 and ((not l.startswith("#")) or (l[:12] in ["#=GC
SS_cons","#=GC RF      "])):
            (id,content) = (l[0:37].strip(),l[37:-1].strip())
            if id not in seqContents:
                seqIDs.append(id)
                seqContents[id] = ""
            seqContents[id] += content
        elif len(l)>1:
            for id in seqIDs:
                outfile.write(string.ljust(id, 37)+seqContents[id]
                             +"\n")
            seqIDs,seqContents = [],{}
            outfile.write(l)
    outfile.close()
# Transforms a 'blocked' STOCKHOLM file (eg RFAM alignment) into a linear one
# Arg1: path to 'blocked' STOCKHOLM file
# Arg2: path to 'linearized' STOCKHOLM output file
if __name__=="__main__":
    (inStoc,outFile) = (sys.argv[1],sys.argv[2])
    unblockStockholm(inStoc,outFile)
```

15 Running [R2R](#) to precompute conservation levels

```
# Compute conservation levels
```

```
r2r --GSC-weighted-consensus XXXX.sto XXXX.cons.sto 3 0.97 0.9 0.75 4 0.97 0.9
      0.75 0.5 0.1
```

16 Simple input `meta` file for R2R

```
XXXX.user.sto
XXXX.user.sto oneseq Pyrococcus_furiosus
XXXX.user.sto skeleton-with-pairbonds
```

17 Running R2R (Stockholm/meta → SVG/PDF)

```
r2r XXXX.meta XXXX.pdf
r2r XXXX.meta XXXX.svg
```

18 Running RChie (Fasta multiple sequence alignment+DBN → PDF)

```
Rscript rchie.R --msafile XXXX.txt --colour1 "#4DAF4A" --msacol "#00A651,#0072BC
, #00B9F2 ,#F15A22 ,#231F20 ,#AAAAAA ,#DA6FAB" --pdf --output="out.pdf" --format1
"vienna" YYYY.txt
```

19 Script to generate a tree representation using GraphViz (DBN → DOT)

```
import sys
# Get custom styles by changing these lines (cf GraphViz manual)
STYLE_DEFAULT = "shape=\\\"rectangle\\\",style=filled,margin=\\\"0,0\\\",fontsize=20,color=grey40,fontcolor=grey20,
fillcolor=grey90,fontname=\\\"Helvetica\\\""
STYLE_UNPAIRED = "shape=\\\"circle\\\",color=blue,fillcolor=aliceblue"
STYLE_PAIRED = "shape=\\\"hexagon\\\""
STYLE_EDGES = "color=grey50"
# Converts an RNA sequence/secondary structure (dot-parenthesis notation) into a DOT-formatted GraphViz input
# , written into a previously opened file f
def drawAsDOT (seq,secstr,f):
    print >> f, "digraph rna{\n    node [%s];\n    edge [%s];\n    -1 [label=\\\"Root\\\"];"%(STYLE_DEFAULT,STYLE_EDGES
    )
    stack = [-1]
    for i in range(len(secstr)):
        k = stack[-1]
        if secstr[i]== "(":
            stack.append(i)
            print >> f, " %s -> %s;"%(k,i)
        elif secstr[i]== ")":
            stack.pop()
            print >> f, " %s [label=%s,%s];"%(k,seq[k]+seq[i],STYLE_PAIRED)
        else:
            print >> f, " %s -> %s;"%(k,i)
            print >> f, " %s [label=%s,%s];"%(i,seq[i],STYLE_UNPAIRED)
    print >> f, " }"
#Typical usage of this script (RNAToDOT.py):
# python RNAToDOT.py (((...))) GGGAUACCC > rnaTree.dot
# dot -Tpdf -o rnaTree.pdf rnaTree.dot
if __name__=="__main__":
    struct,seq = sys.argv[1],sys.argv[2]
    drawAsDOT (seq,struct,sys.stdout)
```

20 Another simple input `meta` file for R2R

```
RF00065_guide_target.cons.sto
RF00065_guide_target.sto      oneseq  Pa21-S892
RF00065_guide_target-hybrid.cons.sto
```

21 Advanced input `meta` file for R2R

```
RF00065_seed.cons.sto
RF00065_seed.sto      oneseq  Pyrococcus_furirosus
RF00065_seed.sto      oneseq  Pyrococcus_abyssi_GE
RF00065_seed.sto      oneseq  Pyrococcus_horikoshi
RF00065_seed.sto      oneseq  Thermococcus_kodakar
RF00065_seed.cons.sto skeleton-with-pairbonds
RF00065_seed-IL00P75.cons.sto
RF00065_seed-IL00P85.cons.sto
```

22 Sample [Stockholm](#) file for interacting RNAs, usable as an input for R2R

```

#=GF SUBFAM_hybrid_R2R box_nuc b rgb:200,0,0
#=GF SUBFAM_hybrid_R2R box_nuc f rgb:200,0,0
#=GF SUBFAM_hybrid_R2R box_nuc g rgb:200,0,0
#=GF SUBFAM_hybrid_R2R box_nuc c rgb:200,0,0
#=GF SUBFAM_hybrid_R2R outline_nuc E
#=GF SUBFAM_hybrid_R2R outline_nuc D
#=GF SUBFAM_hybrid_R2R outline_nuc m
#=GF SUBFAM_hybrid_R2R outline_nuc n
#=GF SUBFAM_hybrid_R2R outline_nuc i
#=GF SUBFAM_hybrid_R2R set_dir pos0 -90
#=GF SUBFAM_hybrid_R2R place_explicit b b-- +45 3 0 0 0 0
#=GF SUBFAM_hybrid_R2R place_explicit g g-- +90 1.5 0 0 0 90
#=GF SUBFAM_hybrid_R2R place_explicit c c-- 90 1 0 0 0 0
#=GF SUBFAM_hybrid_R2R place_explicit h h-- +45 3 0 0 0 90
#=GF SUBFAM_hybrid_R2R place_explicit m m-- 0 1 0 0 0 -90
#=GF SUBFAM_hybrid_R2R place_explicit o o-- 0 2.5 0 0 0 90
#=GF SUBFAM_hybrid_R2R tick_label i target
#=GF SUBFAM_hybrid_R2R tick_label a guide
#=GF SUBFAM_hybrid_R2R shade_along_backbone A rgb:193,255,193
#=GF SUBFAM_hybrid_R2R shade_along_backbone a rgb:193,255,193
#=GF SUBFAM_hybrid_R2R shade_along_backbone E rgb:193,255,193
#=GF SUBFAM_hybrid_R2R shade_along_backbone n rgb:193,255,193
#=GF SUBFAM_hybrid_R2R shade_along_backbone D rgb:255,228,196
#=GF SUBFAM_hybrid_R2R shade_along_backbone d rgb:255,228,196
#=GF SUBFAM_hybrid_R2R shade_along_backbone h rgb:255,228,196
#=GF SUBFAM_hybrid_R2R shade_along_backbone H rgb:255,228,196

#=GF R2R_oneseq Pa21-S892 shade_along_backbone KLOOP:K rgb:0,129,255
#=GF R2R_oneseq Pa21-S892 shade_along_backbone ILOOP:P rgb:193,255,193
#=GF R2R_oneseq Pa21-S892 shade_along_backbone ILOOP:Q rgb:193,255,193
#=GF R2R_oneseq Pa21-S892 shade_along_backbone ILOOP:X rgb:193,255,193
#=GF R2R_oneseq Pa21-S892 shade_along_backbone ILOOP:Y rgb:255,228,196
#=GF R2R_oneseq Pa21-S892 shade_along_backbone ANA:C rgb:200,200,200

//
```

23 Sample Stockholm file for a single RNA, usable as an input for R2R

```

# STOCKHOLM 1.0
#=GF ID snoR9
#=GF AC RF00065
#=GF DE Small nucleolar RNA snoR9
#=GF AU Bateman A, Daub J
#=GF GA 50.0
#=GF NC 49.8
#=GF TC 68.7
#=GF SE Bateman A
#=GF SS Published; PMID:12032319
#=GF TP Gene; snRNA; snoRNA; CD-box;
#=GF BM cmbuild -F CM SEED; cmcalibrate --mpi -s 1 CM
#=GF BM cmsearch -Z 274931 -E 1000000 --toponly CM SEQDB
#=GF DR SO:0000593 SO:C_D_box_snoRNA
#=GF DR GO:0006396 GO:RNA processing
#=GF DR GO:0005730 GO:nucleolus
#=GF RN [1]
#=GF RM 12032319
#=GF RT Noncoding RNA genes identified in AT-rich hyperthermophiles.
#=GF RA Klein RJ, Misulovin Z, Eddy SR;
#=GF RL Proc Natl Acad Sci U S A 2002;99:7542-7547.
#=GF CC snoRNA R9 is a member of the C/D class of snoRNA which contain
the C (UGAUGA) and D (CUGA) box motifs. R9 was identified in a
#=GF CC computational screen in AT-rich hyperthermophiles [1]. R9 was
#=GF CC found to overlap with the smaller snoRNA R19 which is currently a
#=GF CC member of Pyrococcus C/D box snoRNA family Rfam:RF00095.
#=GF WK http://en.wikipedia.org/wiki/Small_nucleolar_RNA_snoR9
#=GF SQ 5

#=GS Pyrococcus_furius AC AE009950.1/163991-163864
#=GS Pyrococcus_abysse GE AC AJ248283.1/230575-230449
#=GS Pyrococcus_horikoshi AC BA000001.2/215834-215709
#=GS P.furius AC AF468960.1/1-128
#=GS Thermococcus_kodakar AC AP006878.1/47908-47779

Pyrococcus_furius GGGCCCGGUU.CCCGCCCUUCGGGGAAUCGUGAACCGGGGUUCGGACCGGGCCCACA..
AUGGGAUGAUGACCUUUUGCUUACUGAACACAUGAUGACCACGCCCUUCGCUGAC.CUAAAUAUUGAC
Pyrococcus_abysse GE GGGCCCGGUU.CCCGCCCUUCGGGGAAUCGUGAACCGGGGUUCGGCCGUACAG..G..
UUUAUGAUGAACUUUUGCUUUGCUGAUGUGGUGAUGAGCACGCCCUUCGCUGAUACUCUCUGCUCAU
Pyrococcus_horikoshi CGGCCCGGUU.CCCGCCCUUCGGGGAAUCGUGAACCGGGGUUCGGACCGGGCGACA..GG.
GGAUGAAGAGCUUUUGCUUUGCUGAGCAGAUGAUGACACGCCCUUCGCUGAC.CU.GCUAAUUGAC
P.furius GGGCCCGGUU.CCCGCCCUUCGGGGAAUCGUGAACCGGGGUUCGGACCGGGCCCACA..
AUGGGAUGAUGACCUUUUGCUUACUGAACACAUGAUGACCACGCCCUUCGCUGAC.CUAAAUAUUGAC
```

```

Thermococcus_kodakar
    GGGCCUGGCGUCCCCGGGAAACGUGAACCGGGGUUCCUGCCAGGCCUACACCGGGGAUGAAGAGCUUUUGCUUGCUGAC..
    UGUGAUGAGCACGCCCUUCACUGACCCGUACAGCUU
#=GC SS_cons <<<<<<.....<.<<<<<<.....>>>>>>.....>>>>>>.....
#=GC RF gGCCCGGu..CCGCCCUCGGGAUCGUGAACCGGGGuUCCggCCGGGCCcACA..
    auguuAUGAUGAaCUUUUGCUUUaCUGAagagaUGAUGAgCACGCCUUCGCUAu..CuaaaUauUugAu
#=GR Pyrococcus_furiosus DEL_COLS .....
-----.
#=GR Pyrococcus_abyssi_GE DEL_COLS .....
-----.
#=GR Pyrococcus_horikoshi DEL_COLS .....
-----.
#=GR P.furiosus DEL_COLS .....
-----.
#=GR Thermococcus_kodakar DEL_COLS .....
-----.
#=GC R2R_LABEL p.....1.....2.....3..4...QQQqQQ5...6RRRrRRRs...7...
    .....8
#=GC R2R_XLABEL_KLOOP .....KKKKKKKK...
-----.
#=GC R2R_XLABEL_KLOOPL .....k...
-----.
#=GC R2R_XLABEL_ILOOP .....IIIIIIII...JJJJJ...
-----.
#=GC R2R_XLABEL_ILOOPL .....i.....j...
-----.
#=GC R2R_XLABEL_ANA .....CCC...
-----.
#=GC R2R_XLABEL_ANAL .....c...
-----.
#=GC SUBFAM_LABEL_TERM .....x...
-----.
#=GC SUBFAM_ILOOP75_R2R_LABEL -----<.....B>-----<.KLMNO.>...
-----.
#=GC SUBFAM_ILOOP85_R2R_LABEL -----<.....B>-----<.KLMNO.>...
-----.

#=GF R2R keep p s
#=GF R2R var_backbone_range 1 2
#=GF R2R var_backbone_range 3 4
#=GF R2R var_backbone_range 5 6
#=GF R2R var_backbone_range 7 8
#=GF R2R shade_along_backbone ILOOP:I rgb:0,255,0
#=GF R2R shade_along_backbone ILOOP:J rgb:0,255,0
#=GF R2R shade_along_backbone KLOOP:K rgb:0,129,255
#=GF R2R shade_along_backbone ANA:C rgb:200,200,200
#=GF R2R tick_label ANAL:c ANA box
#=GF R2R tick_label KLOOPL:k K-Loop
-----.
#=GF SUBFAM_REGEX_PRED ILOOP75 TERM [.]
#=GF SUBFAM_REGEX_PRED ILOOP85 TERM [A-Z]

#=GF SUBFAM_ILOOP75_R2R no5
#=GF SUBFAM_ILOOP75_R2R shade_along_backbone ILOOP:I rgb:0,255,0
#=GF SUBFAM_ILOOP75_R2R tick_label ILOOPL:i 7 nt
#=GF SUBFAM_ILOOP75_R2R outline_nuc ILOOP:I
#=GF SUBFAM_ILOOP75_R2R var_backbone_range_size_fake_nucs 1 M M 5 nt
#=GF SUBFAM_ILOOP75_R2R var_backbone_range_size_fake_nucs 1 B B
#=GF SUBFAM_ILOOP75_R2R var_backbone_range_size_fake_nucs 1 K K
#=GF SUBFAM_ILOOP75_R2R var_backbone_range_size_fake_nucs 1 L L
#=GF SUBFAM_ILOOP75_R2R var_backbone_range_size_fake_nucs 1 N N
#=GF SUBFAM_ILOOP75_R2R var_backbone_range_size_fake_nucs 1 O O
-----.
#=GF SUBFAM_ILOOP85_R2R no5
#=GF SUBFAM_ILOOP85_R2R shade_along_backbone ILOOP:I rgb:0,255,0
#=GF SUBFAM_ILOOP85_R2R tick_label ILOOPL:i 8 nt
#=GF SUBFAM_ILOOP85_R2R outline_nuc ILOOP:I
#=GF SUBFAM_ILOOP85_R2R var_backbone_range_size_fake_nucs 1 M M 5 nt
#=GF SUBFAM_ILOOP85_R2R var_backbone_range_size_fake_nucs 1 B B
#=GF SUBFAM_ILOOP85_R2R var_backbone_range_size_fake_nucs 1 K K
#=GF SUBFAM_ILOOP85_R2R var_backbone_range_size_fake_nucs 1 L L
#=GF SUBFAM_ILOOP85_R2R var_backbone_range_size_fake_nucs 1 N N
#=GF SUBFAM_ILOOP85_R2R var_backbone_range_size_fake_nucs 1 O O
-----.
#=GF R2R_oneseq Pyrococcus_furiosus shade_along_backbone ILOOP:I rgb:0,255,0
#=GF R2R_oneseq Pyrococcus_furiosus tick_label ILOOPL:i 5' guide sequence
#=GF R2R_oneseq Pyrococcus_furiosus shade_along_backbone ILOOP:J rgb:0,255,0
#=GF R2R_oneseq Pyrococcus_furiosus tick_label ILOOPL:j 3' guide sequence
#=GF R2R_oneseq Pyrococcus_furiosus shade_along_backbone KLOOP:K rgb:0,129,255
#=GF R2R_oneseq Pyrococcus_furiosus var_backbone_range_size_fake_nucs 10 7 8
#=GF R2R_oneseq Pyrococcus_furiosus inline_nuc ILOOP:I
#=GF R2R_oneseq Pyrococcus_furiosus inline_nuc ILOOP:J

```

```

#=GF R2R_oneseq Pyrococcus_furius inline_nuc KLOOP:K
#=GF R2R_oneseq Pyrococcus_furius outline_nuc Q
#=GF R2R_oneseq Pyrococcus_furius outline_nuc q
#=GF R2R_oneseq Pyrococcus_furius outline_nuc R
#=GF R2R_oneseq Pyrococcus_furius outline_nuc r
#=GF R2R_oneseq Pyrococcus_furius outline_nuc s
#=GF R2R_oneseq Pyrococcus_furius tick_label q 8 bp
#=GF R2R_oneseq Pyrococcus_furius tick_label r 9 bp

#=GF R2R_oneseq Pyrococcus_abysii_GE shade_along_backbone ILOOP:I rgb:0,255,0
#=GF R2R_oneseq Pyrococcus_abysii_GE tick_label ILOOPL:i 5' guide sequence
#=GF R2R_oneseq Pyrococcus_abysii_GE shade_along_backbone ILOOPJ:J rgb:0,255,0
#=GF R2R_oneseq Pyrococcus_abysii_GE tick_label ILOOPL:j 3' guide sequence
#=GF R2R_oneseq Pyrococcus_abysii_GE shade_along_backbone KLOOP:K rgb:0,129,255
#=GF R2R_oneseq Pyrococcus_abysii_GE var_backbone_range_size_fake_nucs 10 7 8
#=GF R2R_oneseq Pyrococcus_abysii_GE inline_nuc ILOOP:I
#=GF R2R_oneseq Pyrococcus_abysii_GE inline_nuc ILOOP:J
#=GF R2R_oneseq Pyrococcus_abysii_GE inline_nuc KLOOP:K
#=GF R2R_oneseq Pyrococcus_abysii_GE outline_nuc Q
#=GF R2R_oneseq Pyrococcus_abysii_GE outline_nuc q
#=GF R2R_oneseq Pyrococcus_abysii_GE outline_nuc R
#=GF R2R_oneseq Pyrococcus_abysii_GE outline_nuc r
#=GF R2R_oneseq Pyrococcus_abysii_GE outline_nuc s
#=GF R2R_oneseq Pyrococcus_abysii_GE tick_label q 8 bp
#=GF R2R_oneseq Pyrococcus_abysii_GE tick_label r 9 bp

#=GF R2R_oneseq Pyrococcus_horikoshi shade_along_backbone ILOOP:I rgb:0,255,0
#=GF R2R_oneseq Pyrococcus_horikoshi tick_label ILOOPL:i 5' guide sequence
#=GF R2R_oneseq Pyrococcus_horikoshi shade_along_backbone ILOOPJ:J rgb:0,255,0
#=GF R2R_oneseq Pyrococcus_horikoshi tick_label ILOOPL:j 3' guide sequence
#=GF R2R_oneseq Pyrococcus_horikoshi shade_along_backbone KLOOP:K rgb:0,129,255
#=GF R2R_oneseq Pyrococcus_horikoshi var_backbone_range_size_fake_nucs 10 7 8
#=GF R2R_oneseq Pyrococcus_horikoshi inline_nuc ILOOP:I
#=GF R2R_oneseq Pyrococcus_horikoshi inline_nuc ILOOP:J
#=GF R2R_oneseq Pyrococcus_horikoshi inline_nuc KLOOP:K
#=GF R2R_oneseq Pyrococcus_horikoshi outline_nuc Q
#=GF R2R_oneseq Pyrococcus_horikoshi outline_nuc q
#=GF R2R_oneseq Pyrococcus_horikoshi outline_nuc R
#=GF R2R_oneseq Pyrococcus_horikoshi outline_nuc r
#=GF R2R_oneseq Pyrococcus_horikoshi outline_nuc s
#=GF R2R_oneseq Pyrococcus_horikoshi tick_label q 8 bp
#=GF R2R_oneseq Pyrococcus_horikoshi tick_label r 9 bp

#=GF R2R_oneseq P.furius shade_along_backbone ILOOP:I rgb:0,255,0
#=GF R2R_oneseq P.furius tick_label ILOOPL:i 5' guide sequence
#=GF R2R_oneseq P.furius shade_along_backbone ILOOPJ:J rgb:0,255,0
#=GF R2R_oneseq P.furius tick_label ILOOPL:j 3' guide sequence
#=GF R2R_oneseq P.furius shade_along_backbone KLOOP:K rgb:0,129,255
#=GF R2R_oneseq P.furius var_backbone_range_size_fake_nucs 10 7 8
#=GF R2R_oneseq P.furius inline_nuc ILOOP:I
#=GF R2R_oneseq P.furius inline_nuc ILOOP:J
#=GF R2R_oneseq P.furius inline_nuc KLOOP:K
#=GF R2R_oneseq P.furius outline_nuc Q
#=GF R2R_oneseq P.furius outline_nuc q
#=GF R2R_oneseq P.furius outline_nuc R
#=GF R2R_oneseq P.furius outline_nuc r
#=GF R2R_oneseq P.furius outline_nuc s
#=GF R2R_oneseq P.furius tick_label q 8 bp
#=GF R2R_oneseq P.furius tick_label r 9 bp

#=GF R2R_oneseq Thermococcus_kodakar shade_along_backbone ILOOP:I rgb:0,255,0
#=GF R2R_oneseq Thermococcus_kodakar tick_label ILOOPL:i 5' guide sequence
#=GF R2R_oneseq Thermococcus_kodakar shade_along_backbone ILOOPJ:J rgb:0,255,0
#=GF R2R_oneseq Thermococcus_kodakar tick_label ILOOPL:j 3' guide sequence
#=GF R2R_oneseq Thermococcus_kodakar shade_along_backbone KLOOP:K rgb:0,129,255
#=GF R2R_oneseq Thermococcus_kodakar var_backbone_range_size_fake_nucs 10 7 8
#=GF R2R_oneseq Thermococcus_kodakar inline_nuc ILOOP:I
#=GF R2R_oneseq Thermococcus_kodakar inline_nuc ILOOP:J
#=GF R2R_oneseq Thermococcus_kodakar inline_nuc KLOOP:K
#=GF R2R_oneseq Thermococcus_kodakar outline_nuc Q
#=GF R2R_oneseq Thermococcus_kodakar outline_nuc q
#=GF R2R_oneseq Thermococcus_kodakar outline_nuc R
#=GF R2R_oneseq Thermococcus_kodakar outline_nuc r
#=GF R2R_oneseq Thermococcus_kodakar outline_nuc s
#=GF R2R_oneseq Thermococcus_kodakar tick_label q 8 bp
#=GF R2R_oneseq Thermococcus_kodakar tick_label r 9 bp

#=GF Makefile_skeleton-with-pairbonds
//
```

24 Shell script demonstrating advanced usage of R2R

```

#!/bin/tcsh -f
#
set R2RPATH = ""
if ( $R2RPATH == "" ) then
echo "Usage: please set the R2RPATH variable"
exit
endif

set r2r = "$R2RPATH/src/r2r"
set subfam = "perl $R2RPATH/src>SelectSubFamilyFromStockholm.pl"
set GSCparams = '3 0.97 0.9 0.75 4 0.97 0.9 0.75 0.5 0.1'

$r2r --GSC-weighted-consensus RF00065_seed.sto RF00065_seed.cons.sto $GSCparams

$subfam RF00065_seed.cons.sto ILOOP75 > RF00065_seed-ILOOP75.sto
$r2r --GSC-weighted-consensus RF00065_seed-ILOOP75.sto RF00065_seed-ILOOP75.cons.sto $GSCparams
$subfam RF00065_seed.cons.sto ILOOP85 > RF00065_seed-ILOOP85.sto
$r2r --GSC-weighted-consensus RF00065_seed-ILOOP85.sto RF00065_seed-ILOOP85.cons.sto $GSCparams

setenv R2R_DUMP_FILE RF00065_seed.txt
$r2r RF00065_seed.meta RF00065_seed.pdf
$r2r RF00065_seed.meta RF00065_seed.svg

$r2r --GSC-weighted-consensus RF00065_guide_target.sto RF00065_guide_target.cons.sto $GSCparams

$subfam RF00065_guide_target.cons.sto hybrid > RF00065_guide_target-hybrid.sto
$r2r --GSC-weighted-consensus RF00065_guide_target-hybrid.sto RF00065_guide_target-hybrid.cons.sto $GSCparams

setenv R2R_DUMP_FILE RF00065_guide_target.txt
$r2r RF00065_guide_target.meta RF00065_guide_target.pdf
$r2r RF00065_guide_target.meta RF00065_guide_target.svg

```

25 Shell script for the generation of simplified representations using
VARNA

```

# Top
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd \
-sequenceDBN "GGGCCGGCUCCCGCCUCUCCGGGAACUGGUAACCGGGGUUCGGCCGGCCUACA" \
-structureDBN "((((((.....(.....))))....))))...." \
-highlightRegion "10-16:radius=10,fill=#acf269,outline=#acf269;43-47:radius=10,fill=#acf269,outline=#acf269"
\ 
-o RF00065_testarc1.svg -algorithm line -drawBases False -spaceBetweenBases 0.6 -bpStyle line

# Bottom
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd \
-sequenceDBN "GGGCCGGCUCCCGCCUCUCCGGGAACUGGUAACCGGGGUUCGGCCGGCCUACA" \
-structureDBN "((((((.....(.....))))....))))...." \
-chemProb "10-11:glyph=pin,dir=out,intensity=3,color=#f90000;17-18:glyph=pin,dir=out,intensity=1,color=#
f90000;" \
-highlightRegion "26-27:radius=10,fill=#d2ccf4,outline=#d2ccf4;31-37:radius=10,fill=#d2ccf4,outline=#d2ccf4"
\ 
-o RF00065_testarc2.svg -algorithm line -drawBases False -spaceBetweenBases 0.6 -bpStyle line

```

26 Shell script for using color maps and algorithms using VARNA

```

# panel A
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd \
-sequenceDBN "GGGCCGGGCUCCGCCCUUCGGGGAAUCUGUAACCGGGGUUCGGCCGGCCUACA" \
-structureDBN "((((((.....(.....))))))))...." \
-highlightRegion "10-16:radius=10,fill=#acf269,outline=#acf269;43-47:radius=10,fill=#acf269,outline=#acf269"
 \
-o RF00065_panelA.svg -algorithm radiate -flat true -drawBases False -spaceBetweenBases 0.6 \
-title "RF00065: Pyrococcus abyssi GE" -titleSize 12

# panel B
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd \
-sequenceDBN "GGGCCGGGCUCCGCCCUUCGGGGAAUCUGUAACCGGGGUUCGGCCGGCCUACA" \
-structureDBN "((((((.....(.....))))))))...." \
-chemProb "9-10:glyph=pin,dir=out,intensity=1,color=#3e844b;10-11:glyph=pin,dir=out,intensity=1,color=#3e844b
;" \

```

```

-o RF00065_panelB.svg -algorithm radiate -flat true -drawBases False -spaceBetweenBases 0.6
# panel C
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd \
-sequenceDBN "GGGCCGCGCUCCGCCCUUCGGGGAAUCGUGAACCGGGGUUCGGCCGGCCUACAC" \
-structuredBn "((((((.....(((((.....)))))))).....))))...." \
-colorMap "0;0;0;0;0;0;0;0;1;1;2;3;2;1;1;1;3;3;1;1;0;0;0;2;2;2;3;3;1;3;2; \
1;1;1;1;0;0;0;1;3;3;1;0;0;0;0;0;0;1;1;3" \
-colorMapMax 3 -colorMapMin 0 -colorMapStyle green \
-o RF00065_panelC.svg -algorithm radiate -flat true -drawBases True -spaceBetweenBases 0.6

# panels D, E, F
java -cp VARNAvx-y.jar fr.orsay.lri.varna.applications.VARNAcmd \
-sequenceDBN "GGGCCGCGCUCCGCCCUUCGGGGAAUCGUGAACCGGGGUUCGGCCGGCCUACAC" \
-structuredBn "((((((.....(((((.....)))))))).....))))...." \
-chemProb "9-10:glyph=pin,dir=out,intensity=1,color=#3e844b;10-11:glyph=pin,dir=out,intensity=1,color=#3e844b" \
;" \
-highlightRegion "26-27:radius=10,fill=#d2ccf4,outline=#d2ccf4;31-37:radius=10,fill=#d2ccf4,outline=#d2ccf4" \
\
-o RF00065_panelD.svg -algorithm radiate -flat true -drawBases False -spaceBetweenBases 0.6

```