pknotsRG 1.3 manual

Jens Reeder Robert Giegerich Peter Steffen

September 19, 2006

1 NAME

pknotsRG 1.3

2 SYNOPSIS

pknotsRG: Folding canonical RNA secondary structures including pseudoknots

3 INTRODUCTION

pknotsRG is a tool for thermodynamic folding of RNA secondary structures, including the class of canonical simple recursive pseudoknots. This class and the algorithms are described in detail in:

"Design, Implementation and Evaluation of a Practical Pseudoknot Folding Algorithm based on Thermodynamics", Jens Reeder and Robert Giegerich, BMC Bioinformatics, 5:104, 2004

Output:

The output consist of the input sequence, the secondary structure, and the minimum free energy. Basepairs of the first pseudoknot helix are denoted by '{' and '}', the second helix by '[' and ']'. Basepairs not involved in a pseudoknot correspond to normal brackets, '(' and ')'.

Example:

```
UCAAGUAUUCCGAAGCUCAACGGGAAAAUGAGCUA
.......[[[[[.{{{{{{.]]]]].}}}}}}. ( -14.9)
```

Since pknotsRG-loc (option -l) finds the best local pseudoknot, the output of this variant also contains the start and end position of the corresponding subsequence:

```
7.......................34
UUCCGAAGCUCAACGGGAAAAUGAGCU
[[[[[.{{{{{{.]]]]]...}}}}}} (-14.51)
```

When option -s is used, pknotsRG computes a list of unordered suboptimal results.

The energy model we use for pseudoknots:

Destabilizing:

```
creating a new pseudoknot :          9.0   (pkinit)
basepair inside pseudoknot:          0.0
not paired base in pk     :          0.3   (npp)
```

The parameters pkinit and npp have major influence on the pseudoknot energy. The values shown above seem to be the best choice for our test sets. However, feel free to experiment with these values.

Stabilizing:

```
stacking of basepairs     :          stack
base dangling of a pk pair:          dangle
coaxial stacking          :          stack
```

all values in (kcal/mole)

stack and dangle are the normal energies for nested structures like in mfold-3.1 or RNAfold. If nested or unnested structures occur inside a pseudoknot, their energy, of course, contributes to the overall pseudoknot energy.

4 OPTIONS

-h Display this information

Option -h displays a command option overview.

-H `option` Display detailed information on <option>

This displays the corresponding section of the pknotsRG manual for the given command line option.

-v Show version

This shows the version number of pknotsRG.

-m Use mfe strategy

Compute the best structure (the structure with minimum free energy). This is the default strategy.

-f Use enf strategy

Compute the best structure that actually contains at least one pseudoknot.

-l Use loc strategy

Compute the best "compact" pseudoknot, defined as the structure with the lowest energy to length ratio.

-s Show suboptimals

Show suboptimals. If neither **-e** nor **-c** is specified, default is use with **-c** 10. This option can be combined with **-m**, **-f** and **-l**.

-u No dangling bases (implies -s)

This calculates all structures without dangling bases. Setting this option reduces the number of suboptimals drastically.

-o No suboptimals inside pknots (implies -s -l)

Used with loc-strategy, this option discards all suboptimal structures inside of pseudoknots.

-e `value` Set energy range (kcal/mol)

This sets the energy range for suboptimal results. value is the difference to the minimum free energy for the sequence.

-c `value` Set energy range (%%)

This sets the energy range as percentage value of the minimum free energy. For example, when -c 10 is specified, and the minimum free energy is -10.0 kcal/mol, the energy range is set to -9.0 to -10.0 kcal/mol.

-n `value` Set npp-value [0.3]

This option sets the energy penalty for unpaired bases inside of pseudoknots.

-p `value` Set pkinit-value [9]

This options sets the energy penalty for creating a new pseudoknot.

-k `value` Set maximal pknot-length

This option sets the maximal pseudoknot-length allowed.

-w `value` Set window size

Beginning with position 1 of the input sequence, the analysis is repeatedly processed on subsequences of the specified size. After each calculation, the results are printed out and the window is moved by the window position increment (-W), until the end of the input sequence is reached.

-W `value` Set window position increment

This option specifies the increment for the window analysis mode (-w).

-S `value` Specify output width for structures

This splits the structure strings into parts of the specified length. This option is useful when displaying results for long sequences that would otherwise not fit onto the screen.

-F `filename` Read input from file

Let pknotsRG load its input data from file. file can contain a plain single sequence, or multiple sequences in fasta format. When given multiple sequences, each sequence is processed separately in the order of input.

Valid characters in an input sequence are "ACGU" and "acgu". "T" and "t" will be converted to "U". Other letters are mapped to "N" and will not be paired. All other characters are ignored.

-z Colored output

This option enables colored output. In interactive mode, this is the default setting, so use -z to disable colors here.

4.1 Additional interactive mode commands

:s Show current configuration

This command shows the current settings in an interactive pknotsRG session.

:d Reset configuration

This command sets all settings to their default values.

:e `string` Execute system command

Command **:e** executes a system command.

:q Quit

This command quits an interactive pknotsRG session.

5 EXAMPLES

Start pknotsRG in interactive mode:

```
pknotsRG
```

Pipe file input.seq into the program:

```
cat input.seq | pknotsRG
```

6 EXIT STATUS

pknotsRG returns a zero exit status if it succeeds. Non zero is returned in case of failure.

7 AUTHORS

Copyright (C) 2004-2006 Jens Reeder, Robert Giegerich, Peter Steffen

Faculty of Technology, Bielefeld University, 33594 Bielefeld, Germany

Send comments/bug reports to: J.Reeder <jreeder@techfak.uni-bielefeld.de>.
Updates: http://bibiserv.techfak.uni-bielefeld.de/pknotsrg/

pknotsRG contains code from the following contributors:

Editline library (libedit)