

**Supplementary material:**  
**The repertoire and prevalence of pseudoknots in RNA secondary structures**

Einar Andreas Rødland  
Rikshospitalet University Hospital

**Contents**

<b>1</b>	<b>Data material</b>	<b>1</b>
1.1	Structures with pseudoknots . . . . .	1
1.2	Orthodox structures . . . . .	2
<b>2</b>	<b>Maple code</b>	<b>2</b>
2.1	General structures . . . . .	2
2.1.1	Count pseudoknots . . . . .	3
2.2	Families of secondary structures . . . . .	3
2.2.1	Detailed structure counts . . . . .	4
2.2.2	Summary structure counts . . . . .	4
2.2.3	Asymptotics . . . . .	5
2.2.4	Sub-count asymptotics . . . . .	5
2.2.5	Pseudoknot frequencies . . . . .	6

**Email:** e.a.rodland@medisin.uio.no

**Address:** Dept. of Molecular Biology, Rikshospitalet, N-0027 Oslo, Norway.

# 1 Data material

Analyses are based on consensus structures from Rfam [1, 2] (<http://www.sanger.ac.uk/Software/Rfam/> or <http://rfam.wustl.edu/>). The 151 structures listed as published were included in the analyses; predicted structures were not included. The number of stems in each consensus structure was counted: this was done using the seed alignments. The seed data also contains pseudoknots; these were identified, classified and counted. Classification is at the stem level: e.g.  $P_{2,1}^2$  is an H-pseudoknot with 2 + 1 stems.

Additional pseudoknots have been added whenever genomes included in the Rfam full alignments have pseudoknots registered in PseudoBase [3] (<http://www.bio.LeidenUniv.nl/Batenburg/PKB.html>) within or immediately adjacent to the aligned region.

## 1.1 Structures with pseudoknots

The data material consists of 151 consensus structures, whereof 26 contain pseudoknots. Note that for 5 of these structures, the pseudoknots are not annotated in the Rfam seed alignments, and the pseudoknots have instead been obtained from PseudoBase. In these cases, the PKB code is given to identify the PseudoBase structures. Only experimentally verified structures are added, not predicted pseudoknots.

Rfam	Stems	Pseudoknots	Corrections from PseudoBase
RF00010	39	$P_{1,1}^2 + P_{1,1,3}^{3,1}$	
RF00011	17	$P_{1,1,1,3,1}^{5,1}$	
RF00023 *	13 (9+4)	$4 \times P_{1,1}^2$	PKB049+PKB050+PKB051+PKB052
RF00024	5	$P_{1,1}^2$	
RF00028	8	$P_{1,1}^2$	
RF00030	8	$P_{1,1}^2$	
RF00061	13	$P_{2,1}^2$	
RF00094	4	$P_{1,1}^2$	
RF00114 *	2	$P_{1,1}^2$	PKB072
RF00140	4	$P_{1,2,1}^{3,2}$	
RF00165	2	$P_{1,1}^2$	
RF00176	5	$P_{1,1}^2$	
RF00209 *	14 (13+1)	$P_{1,1}^2$	PKB209
RF00216	8	$P_{1,2,1,1}^{4,1}$	
RF00233	5	$P_{1,1}^2$	
RF00252 *	7	$P_{1,1}^2$	PKB191
RF00259	5	$P_{1,1}^2$	
RF00261	8	$P_{1,1,1}^{3,1}$	
RF00290 *	6 (4+2)	$P_{1,1}^2$	PKB172
RF00373	31	$P_{1,1}^2 + P_{1,1,2}^{3,1}$	
RF00381	2	$P_{1,1}^2$	
RF00390	2	$P_{1,1}^2$	
RF00458	8	$3 \times P_{1,1}^2$	
RF00499	3	$P_{1,1}^2$	
RF00505	2	$P_{1,1}^2$	
RF00507	2	$P_{1,1}^2$	

\*) Corrected structures: Pseudoknots and stems not annotated in Rfam have been added. E.g. for RF00023, 4 knotted stems have been added to the 9 in Rfam corresponding to each of the 4 H-pseudoknots. References are given to the corresponding PseudoBase structures (PKB codes).

Structures RF00114 and RF00252 both have two alternative folds, one of which is pseudoknotted; I have used the pseudoknotted structures in the analyses. Structure RF00290 is truncated in Rfam before the H-pseudoknot; I have added the two stems and this H-pseudoknot to the structure.

The total number and types of pseudoknots is  $27 \times P^2$ ,  $3 \times P^{3,1}$ ,  $1 \times P^{3,2}$ ,  $1 \times P^{4,1}$  and  $1 \times P^{5,1}$ .

## 1.2 Orthodox structures

Of the 151 consensus structures included, 125 were orthodox. Below follows the Rfam identification codes of these together with the number of stems in each structure.

**Consensus structures without pseudoknots, Rfam code (stems):** RF00001 (3), RF00002 (4), RF00003 (5), RF00004 (5), RF00005 (4), RF00007 (5), RF00008 (3), RF00009 (7), RF00012 (4), RF00013 (1), RF00014 (3), RF00015 (4), RF00017 (6), RF00019 (1), RF00020 (2), RF00021 (3), RF00025 (4), RF00026 (1), RF00031 (1), RF00032 (1), RF00035 (3), RF00036 (8), RF00037 (1), RF00040 (9), RF00045 (4), RF00048 (1), RF00050 (5), RF00059 (5), RF00100 (6), RF00102 (4), RF00106 (3), RF00107 (2), RF00109 (3), RF00161 (3), RF00162 (4), RF00163 (3), RF00164 (1), RF00167 (3), RF00168 (5), RF00169 (1), RF00171 (5), RF00172 (3), RF00173 (1), RF00175 (4), RF00177 (18), RF00179 (1), RF00180 (1), RF00181 (1), RF00182 (1), RF00183 (3), RF00184 (2), RF00192 (2), RF00193 (10), RF00194 (3), RF00196 (1), RF00197 (1), RF00198 (3), RF00199 (3), RF00207 (1), RF00210 (13), RF00214 (2), RF00215 (2), RF00220 (1), RF00225 (4), RF00230 (2), RF00231 (4), RF00232 (5), RF00234 (4), RF00236 (2), RF00242 (2), RF00250 (1), RF00260 (1), RF00264 (2), RF00286 (2), RF00362 (1), RF00363 (1), RF00364 (1), RF00365 (1), RF00366 (1), RF00367 (1), RF00374 (3), RF00378 (3), RF00382 (1), RF00383 (3), RF00384 (2), RF00385 (1), RF00386 (4), RF00387 (5), RF00388 (4), RF00389 (4), RF00391 (3), RF00433 (3), RF00434 (4), RF00435 (4), RF00436 (1), RF00437 (2), RF00444 (4), RF00453 (1), RF00460 (2), RF00461 (4), RF00462 (1), RF00463 (5), RF00465 (1), RF00466 (3), RF00467 (1), RF00480 (1), RF00481 (3), RF00483 (4), RF00484 (3), RF00485 (1), RF00487 (5), RF00488 (9), RF00489 (1), RF00490 (1), RF00491 (1), RF00492 (1), RF00493 (1), RF00494 (1), RF00496 (1), RF00497 (2), RF00498 (1), RF00500 (1), RF00502 (1), RF00503 (17), RF00506 (2).

## 2 Maple code

Below follows the Maple code used to do the calculations. The code has been grouped into sections. Section 2.1 counts general secondary structures and their pseudoknots, and is not required by the calculations to follow. Section 2.2 gives the basic setup for counting secondary structures with only specific pseudoknots.

All power series will be expanded up to the desired order. This should be specified first using the following Maple code.

```
> Order := 9; # How many terms to include in power series
```

Order 9 means that knots of up to complexity 8, i.e. 8 ladders, stems or nucleotides, are counted. The order may be increased, though some of the calculations may become time consuming for higher orders: particularly power series expansions at the nucleotide level as these consist of power series in two variables,  $t$  and  $x$ .

Note that results of Maple expressions ending with semi-colon are printed, whereas those ending with colon are not printed.

### 2.1 General structures

This section shows the calculation of general secondary structures. First, the number of general secondary structures at different levels of detail.

```
> Fl := sum((2*n)! / (2**n*n!) * 1**n, n=0..Order-1); # All structures (ladder level)
fl := series(1-1/Fl, l); # Irreducible structures (ladder level)
Fs := series(subs(l=s/(1+s), Fl), s); # All structures (stem level)
fs := series(1-1/Fs, s); # Irreducible structures (stem level)
```

$$Fl := 1 + l + 3l^2 + 15l^3 + 105l^4 + 945l^5 + 10395l^6 + 135135l^7 + 2027025l^8$$

$$fl := l + 2l^2 + 10l^3 + 74l^4 + 706l^5 + 8162l^6 + 110410l^7 + 1708394l^8 + O(l^9)$$

$$Fs := 1 + s + 2s^2 + 10s^3 + 68s^4 + 604s^5 + 6584s^6 + 85048s^7 + 1269680s^8 + O(s^9)$$

$$fs := s + s^2 + 7s^3 + 49s^4 + 463s^5 + 5281s^6 + 70687s^7 + 1084609s^8 + O(s^9)$$

### 2.1.1 Count pseudoknots

Solve  $F(l) = 1 + q_F(l \cdot F(l)^2)$  where  $q_F$  (denoted `q_F` in the Maple code) counts knot-components, then use this to derive the number of collapsed pseudoknots  $\tilde{p}_F$  (denoted `ps_F` in the Maple code).

```
> l_of_t:=solve(t=series(1*F1**2,1),1); # Solve t = l · F(l)2
q_F:=subs(t=1,series(subs(l=l_of_t,F1-1),t)); # Number of knot-components
p_F:=series(q_F-1,1); # Number of pseudoknots
ps_F:=series(subs(l=s/(1+s),p_F),s); # Number of collapsed pseudoknots
```

$$l\_of\_t := t - 2t^2 + t^3 - 6t^4 - 34t^5 - 356t^6 - 4299t^7 - 60558t^8 + O(t^9)$$

$$q\_F := l + l^2 + 4l^3 + 27l^4 + 248l^5 + 2830l^6 + 38232l^7 + 593859l^8 + O(l^9)$$

$$p\_F := l^2 + 4l^3 + 27l^4 + 248l^5 + 2830l^6 + 38232l^7 + 593859l^8 + O(l^9)$$

$$ps\_F := s^2 + 2s^3 + 18s^4 + 160s^5 + 1825s^6 + 24486s^7 + 377853s^8 + O(s^9)$$

### 2.2 Families of secondary structures

This section is used to analyse secondary structures containing only specific kinds of pseudoknots. The allowed pseudoknots are specified at the level of collapsed pseudoknots, and encoded into  $\tilde{p}(s)$  (denoted `ps` in the Maple code): e.g.  $\tilde{p}(s) = s^2$  for counting structures with H-pseudoknots.

```
> ps:=s**2; # Collapsed pseudoknots allowed
```

$$ps := s^2$$

Calculate  $q(l) = l + \tilde{p}(l/(1-l))$ , and define the equation in  $G(l)$ ,  $\tilde{G}(s)$  or  $\hat{G}(t, x)$  (all denoted by `G` in the Maple code).

```
> ql:=1+subs(s=1/(1-l),ps):
Eq:=1+subs(l=1*G**2,ql)-G:
eq1:=numer(simplify(Eq)): # Ladder level
eqs:=numer(simplify(subs(l=s/(1+s),Eq))): # Stem level
eqt:=numer(simplify(subs(l=s/(1+s), # convert to stems first
G=G/v, # G̃ → Ĝ/v
s=v**2*1/(1-w*1), # separate stems by v, ladders by w
l=(t**2*x)**m/(1-t**2*x), # ladders of length ≥ m
u=t**k, w=v**2-1, v=1/(1-t), # hairpin loops of length ≥ k
m=2, k=3, # set minimal lengths (m and k)
Eq+1*G*(u-1))): # Nucleotide level (modified equation)
subvar:=x: # Sub-count variable (e.g. x=bonds)
var:=t; eq:=eval(cat('eq',var)): # Set var:=l or s or t (eq:=eq1 or eqs or eqt)
```

$$var := t$$

Here, `var:=t` was used to select the nucleotide level count. This will cause the equation  $eqt = 0$  to be solved for  $G$  in variable  $t$ . The expression  $eqt$  also contains the variable  $x$  which is used to count the number of bonds: the variable used in sub-counts is specified through `subvar:=x`. Please note that `subvar` should be defined even when not in use.

Two other alternatives may be used instead of `var:=t`: `var:=1` for ladder level counts using equation  $eq1 = 0$ , and `var:=s` for stem level counts using equation  $eqs = 0$ .

### 2.2.1 Detailed structure counts

The code below calculates the power series up to the specified order. For high orders, in particular if sub-counts are used, this may be computationally demanding.

```
> G_sol:=solve(series(eq+var**Order, var), G); # Allowed structures
  if var=t then G_eq_g:=subs(v=1/(1-t), v/(1-g/v))
  else G_eq_g:=1/(1-g) fi; # G expressed in terms of g
  g_eq_G:=solve(G=G_eq_g, g): # g expressed in terms of G
  g_sol:=series(subs(G=G_sol, g_eq_G), var): # Irreducible structures
  g_sol:=collect(g_sol, var, expand);
```

$$G_{sol} := 1 + t + t^2 + t^3 + t^4 + t^5 + t^6 + (1 + x^2) t^7 + (3x^2 + x^4 + 1) t^8 + O(t^9)$$

$$g_{sol} := x^2 t^7 + (3x^2 + x^4) t^8 + O(t^9)$$

Note that for counts at the nucleotide level, the relation between  $\hat{G}$  and  $\hat{g}$  is  $\hat{G} = v + \hat{g} + \hat{g}^2/v + \dots$  rather than  $G = 1 + g + g^2 + \dots$ . In the above calculation, at the nucleotide level, the structures counted are those without bonds ( $t^n$ ), the structure ((...)) of length 7 with one length 2 ladder ( $x^2 t^7$ ), the three structures with a length 2 ladder and 4 unbonded nucleotides ( $3t^8 x^2$ ) and the H-pseudoknot with two ladders of length 2 ( $t^8 x^4$ ).

### 2.2.2 Summary structure counts

Should the detailed structure count of section 2.2.2 be too demanding, it is possible to find the number of structure ignoring the sub-count by making the substitution `subvar=1` before solving the equation: e.g.  $t^n x^k \rightarrow t^n$ . If the solution to  $\hat{E}(\hat{G}, t, x) = 0$  is expressed

$$\hat{G}(t, x) = \sum_{n=0}^{\infty} \hat{G}_n(x) t^n, \quad (1)$$

this corresponds to finding  $\hat{G}'_n(1)$ . The average sub-count, i.e. the average power of  $x$ , is then given by  $\hat{G}'_n(1)/G_n(1)$ , where  $\hat{G}'_n(1)$  will be the coefficients of  $(\partial/\partial x)\hat{G}(t, x)|_{x=1}$ . This may be found by using

$$\frac{d}{dx} \hat{E}(\hat{G}, t, x) = \frac{\partial \hat{E}}{\partial x}(\hat{G}, t, x) + \frac{\partial \hat{E}}{\partial \hat{G}}(\hat{G}, t, x) \cdot \frac{\partial \hat{G}}{\partial x}(t, x) = 0 \implies \frac{\partial \hat{G}}{\partial x} = -\frac{\partial \hat{E}/\partial x}{\partial \hat{E}/\partial \hat{G}} \quad (2)$$

and substituting  $x = 1$  and the known expression for  $\hat{G}(t, 1)$ .

```
> eq0:=subs(subvar=1, eq): # Equation ignoring sub-counts
  G_sol0:=solve(series(eq0+var**Order, var), G); # Count ignoring sub-counts (subvar->1)
  G_sol1:=series(subs(subvar=1, G=G_sol0,
    -diff(eq, subvar)/diff(eq, G)), var); # Sum all sub-counts (subvar**n->n)
  seq('mu'[i]=coeff(G_sol1, var, i)/coeff(G_sol0, var, i), i=1..Order-1);
```

$$1 + t + t^2 + t^3 + t^4 + t^5 + t^6 + 2t^7 + 5t^8 + O(t^9)$$

$$2t^7 + 10t^8 + O(t^9)$$

$$\mu_1 = 0, \mu_2 = 0, \mu_3 = 0, \mu_4 = 0, \mu_5 = 0, \mu_6 = 0, \mu_7 = 1, \mu_8 = 2$$

In some cases the series expansion of `G_sol1` may be speeded up considerably by delaying the `G=G_sol1` substitution. The Maple code for doing this more efficiently is:

```
> series(subs(subvar=1, -diff(eq, subvar)/diff(eq, G)), var):
  G_sol1:=series(subs(G=G_sol0, collect(convert(%, polynom), G)), s):
```

### 2.2.3 Asymptotics

When the relation between  $G$  and a variable, say  $l$ , is given by the equation  $E(G, l) = 0$ , the asymptotics of the power series coefficients  $G_n$  of  $G = G(l)$  is given by

$$G_n \approx \sqrt{\frac{l_0 \cdot \frac{\partial E}{\partial l}(G_0, l_0)}{2\pi \cdot \frac{\partial^2 E}{\partial G^2}(G_0, l_0)}} \cdot \frac{l_0^{-n}}{n^{3/2}}. \quad (3)$$

where  $(G, l) = (G_0, l_0)$  solves  $E = \partial E / \partial G = 0$ . When there are more than one solution, the appropriate one is determined by which one on the component of the curve containing  $(G, l) = (1, 0)$ : viewing the implicit plot which graphs the colution curve can help determine which this is.

To solve for irreducible secondary structures, use  $G = 1/(1-g)$  and make the corresponding approximation for  $g_n$ ; at the nucleotide level,  $\hat{G} = v/(1-\hat{g}/v)$ . For the general case, if  $G = \gamma(g, l)$ ,

$$\left. \frac{\partial}{\partial l} E(\gamma(g, l), l) \right|_{\substack{g=g_0 \\ l=l_0}} = \frac{\partial E}{\partial l} + \frac{\partial E}{\partial G} \cdot \left. \frac{\partial \gamma}{\partial l} \right|_{\substack{g=g_0 \\ l=l_0}} = \left. \frac{\partial E}{\partial l} \right|_{\substack{G=G_0 \\ l=l_0}} \quad (4)$$

and

$$\left. \frac{\partial^2}{\partial g^2} E(\gamma(g, l), l) \right|_{\substack{g=g_0 \\ l=l_0}} = \frac{\partial^2 E}{\partial G^2} \cdot \left( \frac{\partial \gamma}{\partial g} \right)^2 + \frac{\partial E}{\partial G} \cdot \left. \frac{\partial^2 \gamma}{\partial g^2} \right|_{\substack{g=g_0 \\ l=l_0}} = \frac{\partial^2 E}{\partial G^2} \cdot \left. \frac{\partial^2 \gamma}{\partial g^2} \right|_{\substack{g=g_0 \\ l=l_0}}. \quad (5)$$

Thus, the asymptotic ratio  $g_n/G_n \approx 1/(\partial \gamma / \partial g)|_{g=g_0, l=l_0}$ . For  $G = 1/(1-g)$ , this ratio is  $1/G_0^2$ .

If there is a sub-count, e.g. using  $x$  to count the number of bonds or a coefficient in  $\tilde{p}(s)$  to count the number of pseudoknots of a given type, this must be set to 1 to count the total number of structures. The subvar :=x used below specifies the name of this sub-count variable.

```
> eq0:=subs(subvar=1, eq) : # Equation ignoring sub-counts
EQ_sol:=fsolve(eq0, diff(eq0, G), G, var, G=1..4, var=0..0.7) ; # Find critical point
#plots[implicitplot](eq0, var=0..1, G=1..4) ; # Run to view curve
G_asymp:=evalf(subs(EQ_sol,
  sqrt(var*diff(eq0, var)/(2*Pi*diff(eq0, G, G)))
  *(1/var)**(n)/n**(3/2))) :
ratio_irr:=subs(g=g_eq_G, EQ_sol, 1/diff(G_eq_g, g)) :
G[n]=G_asymp, g[n]=G_asymp*ratio_irr, ratio=ratio_irr;
```

$$EQ\_sol := \{G = 2.286192146, t = 0.4609569094\}$$

$$G_n = \frac{0.7053422968 \cdot 2.169400175^n}{n^{3/2}}, g_n = \frac{0.4644377414 \cdot 2.169400175^n}{n^{3/2}}, ratio = 0.6584572378$$

The limits  $1 < G \leq 4$  and  $0 \leq t \leq 0.7$  are used to help pick the correct solution in cases when multiple solutions to the equations exist. When in doubt, uncomment the `plots[implicitplot]` to plot the solution curve and control the solution by inspecting the curve.

### 2.2.4 Sub-count asymptotics (cont'd from 2.2.3)

These calculations produce mean, standard deviation and higher cumulants for the sub-counts, and depend on the Maple code in section 2.2.3 which must be run first.

For the equation  $\hat{E}(\hat{G}, t, x) = 0$ , the solution as a power series in  $t$  may be expressed as in equation (1). Then,  $\hat{G}_n(1)$  is the number of structures with  $n$  nucleotides. Given  $n$ , the probability generating function for the number of bonds in a random structure with  $n$  nucleotides is  $P_n(x) = \hat{G}_n(x) / \hat{G}_n(1)$ . The corresponding moment generating function  $M_n(\nu) = P_n(e^\nu)$  and the cumulant generating function  $C_n(\nu) = \ln M_n(\nu)$ .

One reason for being interested in the cumulant generating function is that the coefficients  $c_{n,k}$  in the series expansion  $C_n(\nu) = \sum_{k=1}^{\infty} c_{n,k} \nu^k / k!$  are the cumulants:  $c_{n,1} = \mu_n$  is the mean,  $c_{n,2} = \sigma_n^2$  is the variance,  $c_{n,3}$  is the skewness, etc.

For low  $n$ , the above cumulant generating function can be calculated explicitly from  $\hat{G}_n(x)$ . For higher  $n$ , obtaining  $\hat{G}_n(x)$  may be computationally demanding; yet, asymptotics can be obtained using equation (3). The solution in  $(\hat{G}, t)$  of  $\hat{E} = \partial\hat{E}/\partial\hat{G} = 0$  now depends on  $x$ : i.e. the solution is  $(\hat{G}, t) = (\hat{G}_0(x), t_0(x))$ . By equation (3),

$$P_n(x) = \frac{\hat{G}_n(x)}{\hat{G}_n(1)} \approx \left( \frac{t_0(1)}{t_0(x)} \right)^n \Rightarrow C_n(\nu) \approx n \cdot \ln \left( \frac{t_0(1)}{t_0(e^\nu)} \right). \quad (6)$$

The large  $n$  limit may then be conveniently expressed through

$$C(\nu) = \lim_{n \rightarrow \infty} \frac{C_n(\nu)}{n} = \ln \left( \frac{t_0(1)}{t_0(e^\nu)} \right) = \sum_{k=1}^{\infty} \frac{c_k \nu^k}{k!} = \lim_{n \rightarrow \infty} \sum_{k=1}^{\infty} \frac{c_{n,k} \nu^k}{k!}. \quad (7)$$

Thus,  $c_1 = \lim_{n \rightarrow \infty} \mu_n/n$ ,  $c_2 = \lim_{n \rightarrow \infty} \sigma_n^2/n$ , etc.

I will be reading off the coefficients as the  $k^{\text{th}}$  derivative at  $\nu = 0$ : i.e.  $c_k = (d/d\nu)^k C(\nu)|_{\nu=0}$ . The cumulant generating function  $C(\nu)$  may be expressed as the function  $(\hat{G}, t, \nu) \mapsto \ln(t_0(1)/t)$  composed by the parametrisation  $\nu \mapsto (\hat{G}_0(e^\nu), t_0(e^\nu), \nu)$  of the curve defined by  $\hat{E}(\hat{G}, \hat{t}, e^\nu) = \frac{\partial\hat{E}}{\partial\hat{E}}(\hat{G}, \hat{t}, e^\nu) = 0$ . The term  $t_0(1)$  is denoted by the undefined constant `var0` in the Maple code: as it is a constant, it which will disappear after the first differentiation. This makes

$$\frac{d}{d\nu} = \frac{\partial}{\partial\nu} + \left( \frac{d}{d\nu} \hat{G}_0(e^\nu) \right) \cdot \frac{\partial}{\partial\hat{G}} + \left( \frac{d}{d\nu} t_0(e^\nu) \right) \cdot \frac{\partial}{\partial t} \quad (8)$$

where, for  $\check{E}(\hat{G}, t, \nu) = \hat{E}(\hat{G}, t, e^\nu)$ . Differentiation of  $\check{E}$  and  $\partial\check{E}/\partial\hat{G}$  along the curve  $(\hat{G}_0(\nu), t_0(\nu), \nu)$  yields

$$\begin{bmatrix} 0 & \frac{\partial\check{E}}{\partial t} \\ \frac{\partial^2\check{E}}{\partial\hat{G}^2} & \frac{\partial^2\check{E}}{\partial\hat{G}\partial t} \end{bmatrix} \cdot \frac{\partial}{\partial\nu} \begin{bmatrix} \hat{G}_0(e^\nu) \\ t_0(e^\nu) \end{bmatrix} + \begin{bmatrix} \frac{\partial\check{E}}{\partial\nu} \\ \frac{\partial^2\check{E}}{\partial\hat{G}\partial\nu} \end{bmatrix} = 0 \Rightarrow \begin{aligned} \frac{d}{d\nu} \hat{G}_0(e^\nu) &= \frac{\frac{\partial^2\check{E}}{\partial\hat{G}\partial t} \frac{\partial\check{E}}{\partial\nu} - \frac{\partial\check{E}}{\partial t} \frac{\partial^2\check{E}}{\partial\hat{G}\partial\nu}}{\frac{\partial\check{E}}{\partial t} \frac{\partial^2\check{E}}{\partial\hat{G}^2}} \\ \frac{d}{d\nu} t_0(e^\nu) &= -\frac{\frac{\partial\check{E}}{\partial\nu}}{\frac{\partial\check{E}}{\partial t}} \end{aligned} \quad (9)$$

as  $\partial\check{E}/\partial\hat{G} = 0$  along the curve.

```
> if (diff(eq, subvar) <> 0) then # Run only if sub-counts are in use
  eq:=subs(subvar=exp(nu), eq);
  Dnu:=f->diff(f, nu)-diff(eq, nu)/diff(eq, var)*diff(f, var)
    +(diff(eq, nu)*diff(eq, G, var)-diff(eq, G, nu)*diff(eq, var))
    / (diff(eq, var)*diff(eq, G, G))*diff(f, G);
  Cum:=n->simplify(subs(EQ_sol, nu=0, (Dnu@@n)(ln(var0/var))));
  mu:=Cum(1);
  sigma2:=Cum(2); sigma:=sqrt(sigma2);
  print('mu'=mu, 'sigma'=sigma);
end;
```

$$\mu = .3498817169, \quad \sigma = .2673655791$$

## 2.2.5 Pseudoknot frequencies

The above calculations were all done at the nucleotide level for secondary structures having at most H-pseudoknots. A different level of detail may be chosen by setting `var:=l` or `var:=s` when setting up the calculations.

An alternative kind of analysis is to assess the frequencies of various types of pseudoknots. This may in theory be done at any level of detail, though it is not practical to do it at the level of nucleotides while keeping track of the number of bonds as this would produce as a result a power series  $\hat{G}$  in three variables rather than just two.

Assume, as an example, that `var:=1` is chosen. To count the expected number of double hairpin pseudoknots ( $P^{3,1}$ ) in a secondary structure having at most H-pseudoknots ( $P^2$ ) and double hairpin pseudoknots, specify `ps:=s**2+c*s**3` instead of `ps:=s**2`. I now use  $c$  instead of  $x$  for sub-counts, so I must also change the specification of the sub-count variable to `subvar:=c`. I.e., the modified lines in the Maple code are:

```
> ps:=s**2+c*s**3:
  subvar:=c:
  var:=t:
```

To count the frequencies of pseudoknots in general secondary structures, the pseudoknot count  $\tilde{p}_F$  from section 2.1.1 may be used. E.g. to count pseudoknots of complexity  $dg = 4$  (i.e. collapsed pseudoknot has  $dg$  bonds), use the following code to specify  $\tilde{p}$ :

```
> dg:=4:
  ps:=convert(ps_F, polynom)+coeff(ps_F, s, dg)*(c-1):
```

Note that the use of  $\tilde{p}_F$  (denoted `ps_F` in the Maple code) requires that the code in section 2.1 be run first.

## References

- [1] Griffiths-Jones, S., Bateman, A., Marshall, M., Khanna, A., and Eddy, S. (Jan, 2003) Rfam: an RNA family database.. *Nucleic Acids Res*, **31**(1), 439–41.
- [2] Griffiths-Jones, S., Moxon, S., Marshall, M., Khanna, A., Eddy, S., and Bateman, A. (Jan, 2005) Rfam: annotating non-coding rnas in complete genomes.. *Nucleic Acids Res*, **33 Database Issue**, D121–4.
- [3] vanBatenburg, F., Gulyaev, A., Pleij, C., Ng, J., and Oliehoek, J. (Feb, 2000) Pseudobase: a database with RNA pseudoknots.. *Nucleic Acids Res*, **28**(1), 201–4.