Wang-Landau Sampling of discrete Biopolymer Models

Michael Wolfinger

Institute for Theoretical Chemistry
University Vienna

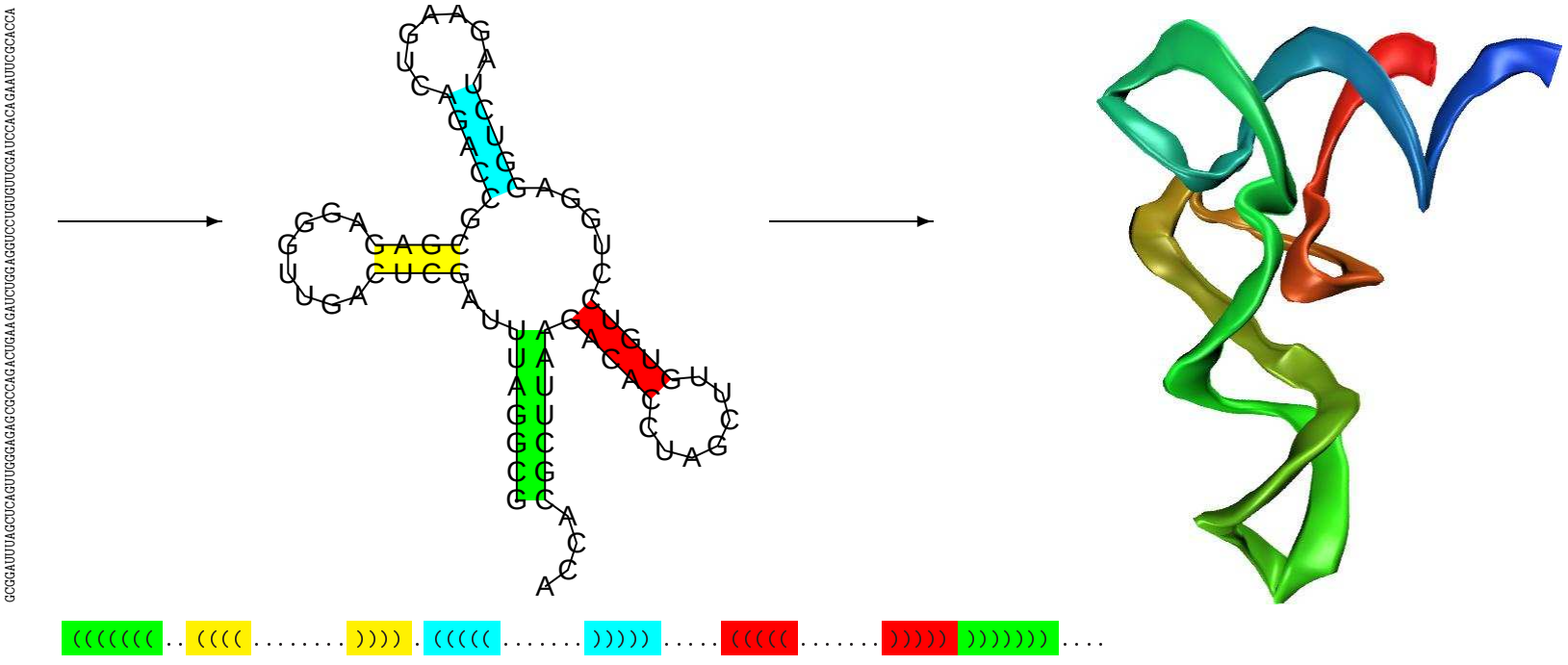
February 22, 2007

tbi

Outline

tbi

The RNA model



A secondary structure is a list of base pairs that fulfills two constraints:

- A base may participate in at most one base pair.
- Base pairs must not cross, i.e., no two pairs (i,j) and (k,l) may have $i < k < j < l$. (no pseudo-knots)

The optimal as well as the suboptimal structures can be computed recursively.

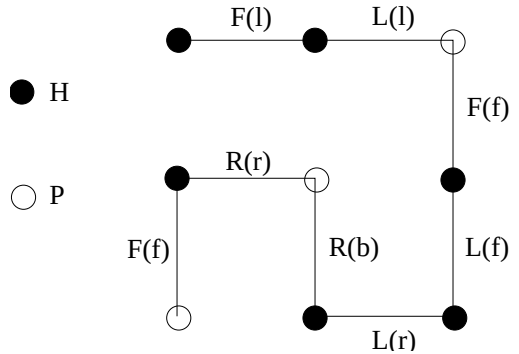
tbi

The HP-model

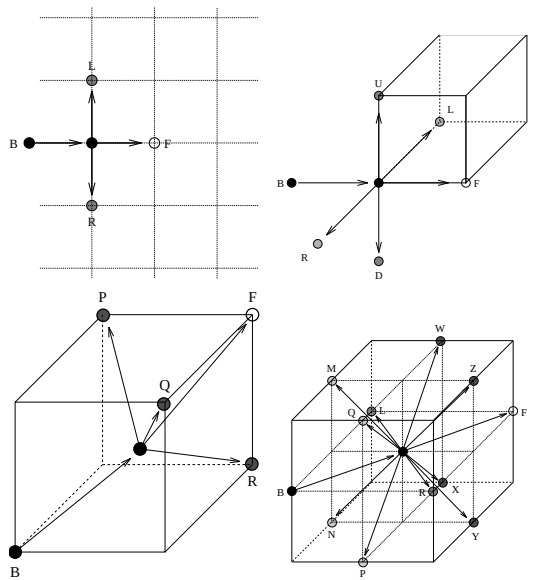
In this *simplified model*, a conformation is a *self-avoiding walk (SAW)* on a given lattice in 2 or 3 dimensions. Each bond is a straight line, bond angles have a few discrete values. The 20 letter alphabet of amino acids (monomers) is reduced to a two letter alphabet, namely **H** and **P**. H represents hydrophobic monomers, P represents hydrophilic or *polar* monomers.

Advantages:

- lattice-independent folding algorithms
- simple energy function
- hydrophobicity can be reasonably modeled



FRRLLFLF



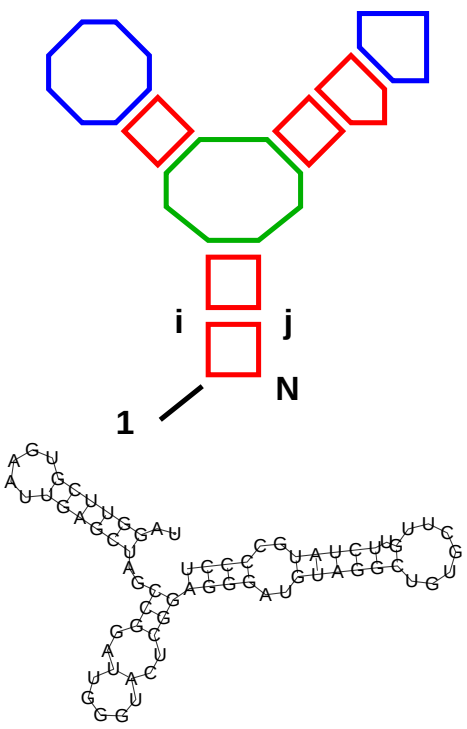
tbi

Energy functions

RNA

The standard energy model expresses the free energy of a secondary structure S as the sum of the energies of its loops l

$$E(\mathrm{S}) = \sum_{l \in \mathrm{S}} E(l)$$



$E = -17.5 kcal/mol$

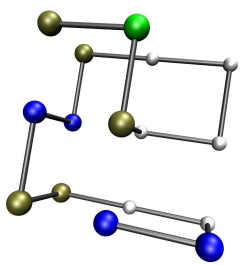
Lattice Proteins

The energy function for a sequence with n residues $\mathfrak{S} = \mathfrak{s}_1\mathfrak{s}_2\ldots\mathfrak{s}_n$ with $\mathfrak{s}_i \in \mathscr{A} = \{a_1, a_2, \ldots, a_b\}$, the alphabet of b residues, and an overall configuration $x = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ on a lattice $\mathscr{L}$ can be written as the sum of pair potentials

$$E(\mathfrak{S}, x) = \sum_{\substack{i < j-1 \\ |\mathbf{x}_i - \mathbf{x}_j| = 1}} \Psi[\mathfrak{s}_i, \mathfrak{s}_j].$$

	H	P
H	-1	0
P	0	0

	H	P	N	X
H	-4	0	0	0
P	0	1	-1	0
N	0	-1	1	0
X	0	0	0	0



$E = -16$

tbi

Folding landscape - energy landscape

The energy landscape of a biopolymer molecule is a complex surface of the (free) energy versus the conformational degrees of freedom.

Number of RNA secondary structures

$c_n \sim 1.86^n \cdot n^{-\frac{3}{2}}$

dynamic programming algorithms available

Number of LP structures

$c_n \sim \mu^n \cdot n^{\gamma-1}$

problem is NP-hard

dim	Lattice Type	μ	γ
2	SQ	2.63820	1.34275
	TRI	4.15076	1.343
	HEX	1.84777	1.345
3	SC	4.68391	1.161
	BCC	6.53036	1.161
	FCC	10.0364	1.162

Formally, three things are needed to construct an energy landscape:

- A set X of configurations
- an energy function $f : X \to \mathbf{R}$
- a symmetric neighborhood relation $\mathfrak{N} : X \times X$

tbi

Folding landscape - energy landscape

The energy landscape of a biopolymer molecule is a complex surface of the (free) energy versus the conformational degrees of freedom.

Number of RNA secondary structures

$$c_n \sim 1.86^n \cdot n^{-\frac{3}{2}}$$

dynamic programming algorithms available

Number of LP structures

$$c_n \sim \mu^n \cdot n^{\gamma-1}$$

problem is NP-hard

dim	Lattice Type	μ	γ
2	SQ	2.63820	1.34275
	TRI	4.15076	1.343
	HEX	1.84777	1.345
3	SC	4.68391	1.161
	BCC	6.53036	1.161
	FCC	10.0364	1.162

Formally, three things are needed to construct an energy landscape:

- A set X of configurations
- an energy function $f : X \to \mathbf{R}$
- a symmetric neighborhood relation $\mathfrak{N} : X \times X$

tbi

Folding landscape - energy landscape

The energy landscape of a biopolymer molecule is a complex surface of the (free) energy versus the conformational degrees of freedom.

Number of RNA secondary structures

$c_n \sim 1.86^n \cdot n^{-\frac{3}{2}}$

dynamic programming algorithms available

Number of LP structures

$c_n \sim \mu^n \cdot n^{\gamma-1}$

problem is NP-hard

dim	Lattice Type	μ	γ
2	SQ	2.63820	1.34275
	TRI	4.15076	1.343
	HEX	1.84777	1.345
3	SC	4.68391	1.161
	BCC	6.53036	1.161
	FCC	10.0364	1.162

Formally, three things are needed to construct an energy landscape:

- A set X of configurations
- an energy function $f : X \to \mathbf{R}$
- a symmetric neighborhood relation $\mathfrak{N} : X \times X$

tbi

Folding landscape - energy landscape

The energy landscape of a biopolymer molecule is a complex surface of the (free) energy versus the conformational degrees of freedom.

Number of RNA secondary structures

$c_n \sim 1.86^n \cdot n^{-\frac{3}{2}}$

dynamic programming algorithms available

Number of LP structures

$c_n \sim \mu^n \cdot n^{\gamma-1}$

problem is NP-hard

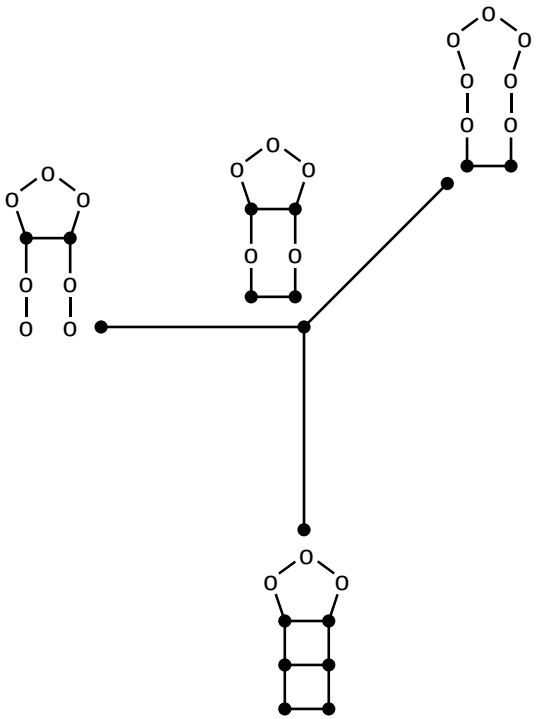
dim	Lattice Type	μ	γ
2	SQ	2.63820	1.34275
	TRI	4.15076	1.343
	HEX	1.84777	1.345
3	SC	4.68391	1.161
	BCC	6.53036	1.161
	FCC	10.0364	1.162

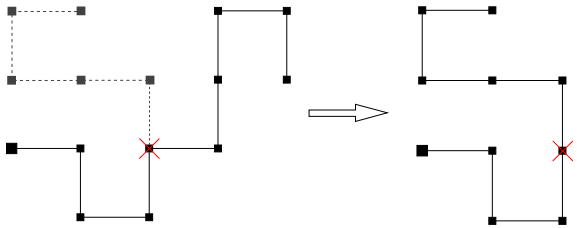
Formally, three things are needed to construct an energy landscape:

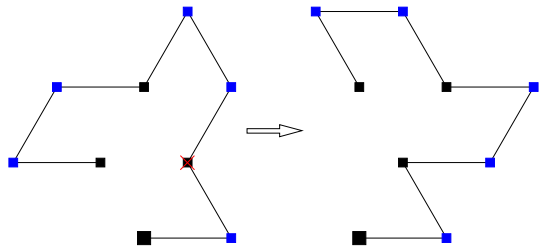
- A set X of configurations
- an energy function $f : X \rightarrow \mathbf{R}$
- a symmetric neighborhood relation $\mathfrak{N} : X \times X$

tbi

The move set







- For each move there must be an inverse move
- Resulting structure must be in X
- Move set must be *ergodic*

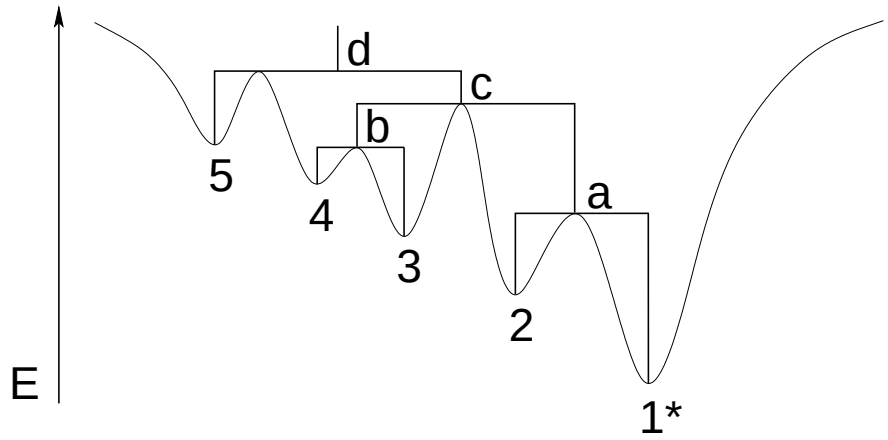
tbi

Energy barriers and barrier trees

Some topological definitions:

A structure is a

- local minimum if its energy is lower than the energy of **all** neighbors
- local maximum if its energy is higher than the energy of **all** neighbors
- saddle point if there are at least two local minima thar can be reached by a downhill walk starting at this point



C. Flamm, I. L. Hofacker, P. F. Stadler, and M. T. Wolfinger.
Barrier trees of degenerate landscapes.
Z. Phys. Chem., 216:155–173, 2002.

tbi

The algorithm of Barriers

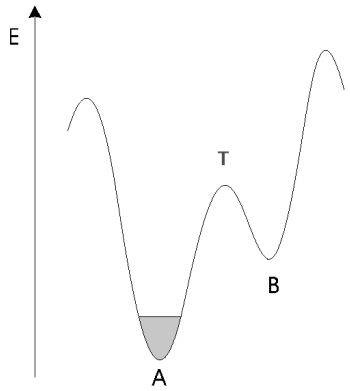
Barriers

Require: all suboptimal secondary structures within a certain energy range from mfe

```
 1: ℬ ⇐ ∅
 2: for all x ∈ subopt do
 3:     ℋ ⇐ ∅
 4:     𝒩 ⇐ generate_neighbors(x)
 5:     for all y ∈ 𝒩 do
 6:         if b ⇐ lookup_hash(y) then
 7:             ℋ ⇐ ℋ ∪ b
 8:         end if
 9:     end for
10:     if ℋ = ∅ then
11:         ℬ ⇐ ℬ ∪ {x}
12:     end if
13:     if |ℋ| ≥ 2 then
14:         merge_basins(ℋ)
15:     end if
16:     write_hash(x)
17: end for
```

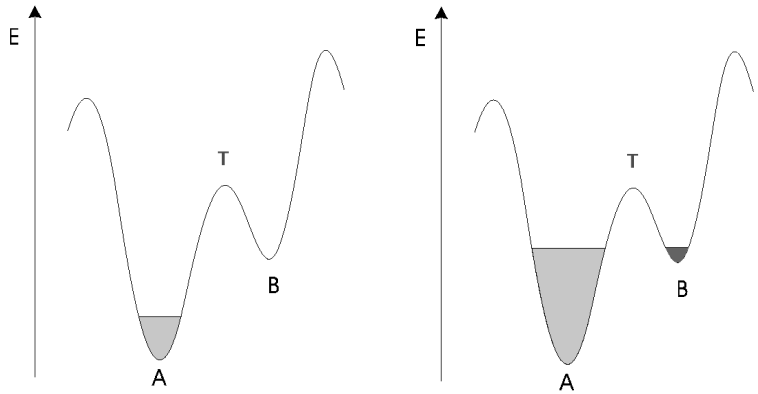
tbi

The flooding algorithm

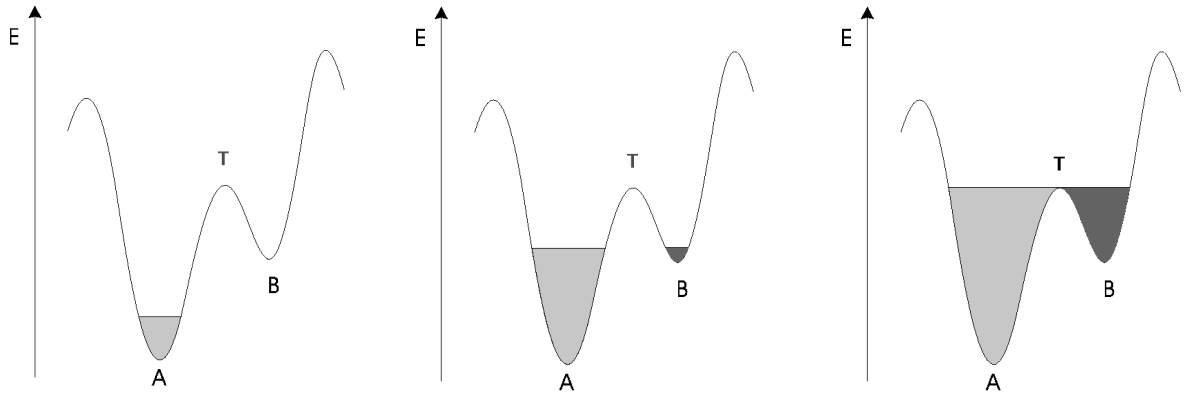


1

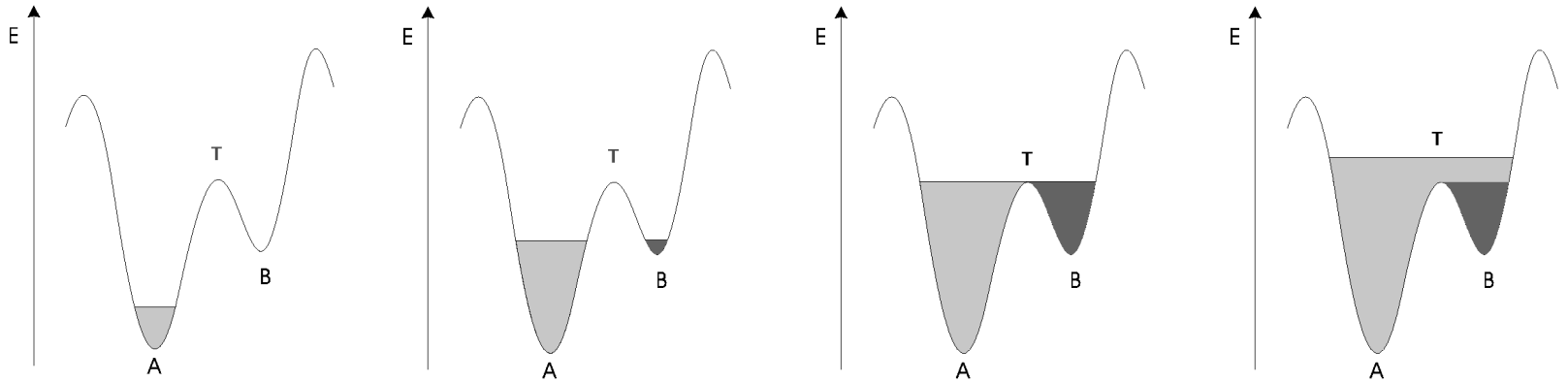
The flooding algorithm



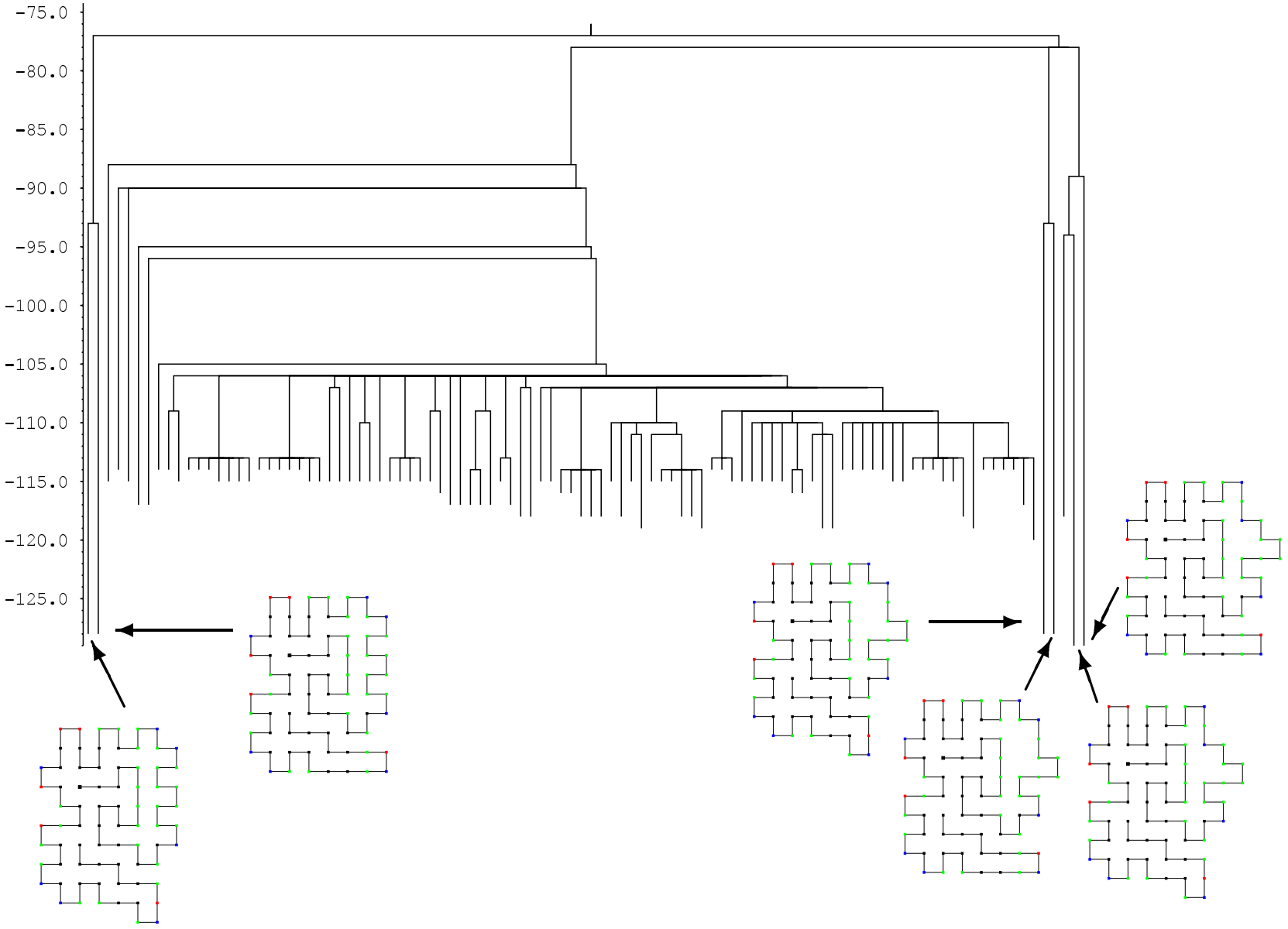
The flooding algorithm



The flooding algorithm



Barrier tree example



Information from the barrier trees

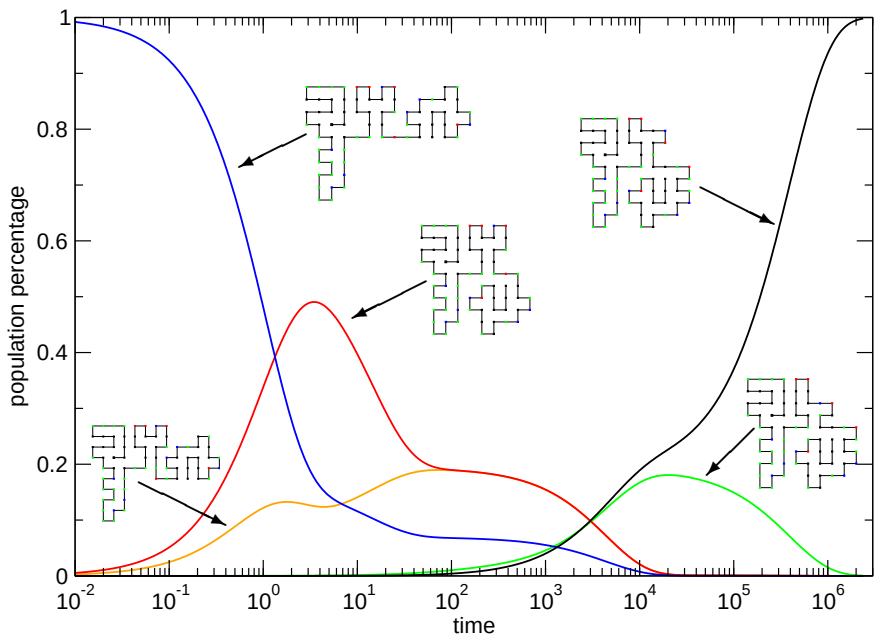
- Local minima
- Saddle points
- Barrier heights
- Gradient basins
- Partition functions
- Free energies of (gradient) basins

With this information, a reduced dynamics can be formulated as a Markov process by means of macrostates (i.e. basins in the barrier tree) and Arrhenius-like transition rates between them.

$$\frac{d}{dt}P_t = \mathbf{U}P_t \quad \Longrightarrow \quad P_t = e^{t\mathbf{U}}P_0$$

- macro-states form a partition of the full configuration space
- transition rates between macro-states

$$r_{\beta\alpha} = \Gamma_{\beta\alpha} \exp\left(-(E^*_{\beta\alpha} - G_\alpha)/kT\right)$$



M. T. Wolfinger, W. A. Svrcek-Seiler, C. Flamm, I. L. Hofacker, and P. F. Stadler.
Efficient computation of RNA folding dynamics.
J. Phys. A: Math. Gen., 37(17):4731–4741, 2004.

tbi

Barrier tree dynamics - problems and pitfalls

The method works fine for moderately sized systems.

Currently, we consider approx. 100 million structures within a single run of BARRIERS to calculate the topology of the landscape.

However, we are interested in larger systems:

- biologically relevant RNA switches
- large 3D lattice proteins

The next steps:

- use high-level diagonalization routines for sparse matrices
- calculate low-energy structures
- sample (thermodynamics properties of) individual basins

tbi

Barrier tree dynamics - problems and pitfalls

The method works fine for moderately sized systems.

Currently, we consider approx. 100 million structures within a single run of BARRIERS to calculate the topology of the landscape.

However, we are interested in larger systems:

- biologically relevant RNA switches
- large 3D lattice proteins

The next steps:

- use high-level diagonalization routines for sparse matrices
- calculate low-energy structures
- sample (thermodynamics properties of) individual basins

Wang-Landau sampling

"A dynamic Monte Carlo algorithm to estimate the density of states by performing a random walk in energy space with a flat histogram"

F. Wang and D. P. Landau.
Efficient, Multiple-Range Random Walk Algorithm to Calculate the Density of States.
Phys. Rev. Lett., 86:2050–2053, 2001.

The classical partition function can be written as the sum over all states, *or* over all energies, i.e.

$$Z = \sum_i e^{-E_i/kT} \equiv \sum_E g(E) e^{-E/kT}$$

Wang-Landau sampling estimates $g(E)$ directly, instead of trying to extract it from a 'standard' Monte Carlo probability distribution.

tbi

Monte Carlo basics

Generally, any (probability) distribution can be sampled by a Monte Carlo-type algorithm. Prerequisites: Detailed Balance

$$\pi(x)\, p(x \to y) = \pi(y)\, p(y \to x)$$

The probability of state x occuring in a classical system, is $\pi(x) = \frac{1}{Z} e^{-E_x/kT}$ ("Boltzmann-sampling")

Metropolis rule:

$$p(x \to y) = \min\left(1, \frac{\pi(y)}{\pi(x)}\right)$$

In Wang-Landau sampling, we have $\pi(x) = \frac{1}{g(E_x)}$ and thus

$$p(x \to y) = \min\left(1, \frac{g(E_x)}{g(E_y)}\right)$$

tbi

Monte Carlo basics

Generally, any (probability) distribution can be sampled by a Monte Carlo-type algorithm. Prerequisites: Detailed Balance

$$\pi(x)\, p(x \to y) = \pi(y)\, p(y \to x)$$

The probability of state x occuring in a classical system, is $\pi(x) = \frac{1}{Z} e^{-E_x/kT}$ ("Boltzmann-sampling")

Metropolis rule:

$$p(x \to y) = \min\left(1, \frac{\pi(y)}{\pi(x)}\right)$$

In Wang-Landau sampling, we have $\pi(x) = \frac{1}{g(E_x)}$ and thus

$$p(x \to y) = \min\left(1, \frac{g(E_x)}{g(E_y)}\right)$$

tbi

Monte Carlo basics

Generally, any (probability) distribution can be sampled by a Monte Carlo-type algorithm. Prerequisites: Detailed Balance

$$\pi(x)\, p(x \to y) = \pi(y)\, p(y \to x)$$

The probability of state x occuring in a classical system, is $\pi(x) = \frac{1}{Z} e^{-E_x/kT}$ ("Boltzmann-sampling")

Metropolis rule:

$$p(x \to y) = \min\left(1, \frac{\pi(y)}{\pi(x)}\right)$$

In Wang-Landau sampling, we have $\pi(x) = \frac{1}{g(E_x)}$ and thus

$$p(x \to y) = \min\left(1, \frac{g(E_x)}{g(E_y)}\right)$$

tbi

Wang-Landau basics

Wang-Landau sampling assumes a crude 'guess' for the density of states, i.e. $g(E_x) = 1$ for all x.
Starting from an (arbitrary) initial state, a random neighbor is chosen with a transition probability

$$p(x \to y) = \min\left(1, \frac{g(E_x)}{g(E_y)}\right)$$

- If the move is accepted, the value of $g(E_y)$ is multiplied with a modfication factor $f > 1$ and the histogram entry $h(E_y)$ is updated.
- If the move is rejected, $g(E_x)$ is multiplied with f and $h(E_x)$ is incremented.

In practice, we work with the logarithm of the density of states, i.e. an update of the density of states yields $ln[g(E)] \to ln[g(E)] + ln(f)$.

tbi

Wang-Landau basics

Wang-Landau sampling assumes a crude 'guess' for the density of states, i.e. $g(E_x) = 1$ for all x.
Starting from an (arbitrary) initial state, a random neighbor is chosen with a transition probability

$$p(x \to y) = \min\left(1, \frac{g(E_x)}{g(E_y)}\right)$$

- If the move is accepted, the value of $g(E_y)$ is multiplied with a modfication factor $f > 1$ and the histogram entry $h(E_y)$ is updated.
- If the move is rejected, $g(E_x)$ is multiplied with f and $h(E_x)$ is incremented.

In practice, we work with the logarithm of the density of states, i.e. an update of the density of states yields $ln[g(E)] \to ln[g(E)] + ln(f)$.

tbi

Wang-Landau continued

A reasonable initial value for the modification factor is $f = e^1 \simeq 2.71828$. The random walk is continued until the histogram $h(E)$ is 'flat'[1], which is typically checked for every 10^6 iterations.
$\rightarrow$ all energy bins have been visited an euqal number of times.
$\rightarrow$ the density of states converges to the true value prop. $ln(f)$.

Then, f is reduced to $f^{1/2}$, i.e. $f_1 = \sqrt{f_0}$ and $h(E)$ is reset to 0.

The random walk is continued, until the histogram becomes 'flat' again, in which case we reset $h(E)$ and modify f to $\sqrt{f}$. This is done, until a final value of $f = \exp(10^{-8}) \simeq 1.00000001$ is reached.

After many iterations, $g(E)$ converges to the true value as f approaches 1. At that point, the random walk satifies detailed balance:

$$\frac{1}{g(E_x)} p(E_x \rightarrow E_y) = \frac{1}{g(E_y)} p(E_y \rightarrow E_x)$$

[1] 'flat' means that $h(E) \geq 0.8 \langle h(E) \rangle$ for all E

Wang-Landau continued

A reasonable initial value for the modification factor is $f = e^1 \simeq 2.71828$. The random walk is continued until the histogram $h(E)$ is 'flat'[1], which is typically checked for every 10^6 iterations.
$\rightarrow$ all energy bins have been visited an euqal number of times.
$\rightarrow$ the density of states converges to the true value prop. $ln(f)$.

Then, f is reduced to $f^{1/2}$, i.e. $f_1 = \sqrt{f_0}$ and $h(E)$ is reset to 0.

The random walk is continued, until the histogram becomes 'flat' again, in which case we reset $h(E)$ and modify f to $\sqrt{f}$. This is done, until a final value of $f = \exp(10^{-8}) \simeq 1.00000001$ is reached.

After many iterations, $g(E)$ converges to the true value as f approaches 1. At that point, the random walk satifies detailed balance:

$$\frac{1}{g(E_x)} p(E_x \rightarrow E_y) = \frac{1}{g(E_y)} p(E_y \rightarrow E_x)$$

[1] 'flat' means that $h(E) \geq 0.8 \langle h(E) \rangle$ for all E

Wang-Landau continued

A reasonable initial value for the modification factor is $f = e^1 \simeq 2.71828$. The random walk is continued until the histogram $h(E)$ is 'flat'[1], which is typically checked for every 10^6 iterations.
$\rightarrow$ all energy bins have been visited an euqal number of times.
$\rightarrow$ the density of states converges to the true value prop. $ln(f)$.

Then, f is reduced to $f^{1/2}$, i.e. $f_1 = \sqrt{f_0}$ and $h(E)$ is reset to 0.

The random walk is continued, until the histogram becomes 'flat' again, in which case we reset $h(E)$ and modify f to $\sqrt{f}$. This is done, until a final value of $f = \exp(10^{-8}) \simeq 1.00000001$ is reached.

After many iterations, $g(E)$ converges to the true value as f approaches 1. At that point, the random walk satifies detailed balance:

$$\frac{1}{g(E_x)} p(E_x \rightarrow E_y) = \frac{1}{g(E_y)} p(E_y \rightarrow E_x)$$

[1] 'flat' means that $h(E) \geq 0.8 \langle h(E) \rangle$ for all E

tbi

Wang-Landau continued

A reasonable initial value for the modification factor is $f = e^1 \simeq 2.71828$. The random walk is continued until the histogram $h(E)$ is 'flat'[1], which is typically checked for every 10^6 iterations.
$\rightarrow$ all energy bins have been visited an euqal number of times.
$\rightarrow$ the density of states converges to the true value prop. $ln(f)$.

Then, f is reduced to $f^{1/2}$, i.e. $f_1 = \sqrt{f_0}$ and $h(E)$ is reset to 0.

The random walk is continued, until the histogram becomes 'flat' again, in which case we reset $h(E)$ and modify f to $\sqrt{f}$. This is done, until a final value of $f = \exp(10^{-8}) \simeq 1.00000001$ is reached.

After many iterations, $g(E)$ converges to the true value as f approaches 1. At that point, the random walk satifies detailed balance:

$$\frac{1}{g(E_x)} p(E_x \rightarrow E_y) = \frac{1}{g(E_y)} p(E_y \rightarrow E_x)$$

[1] 'flat' means that $h(E) \geq 0.8 \langle h(E) \rangle$ for all E

tbi

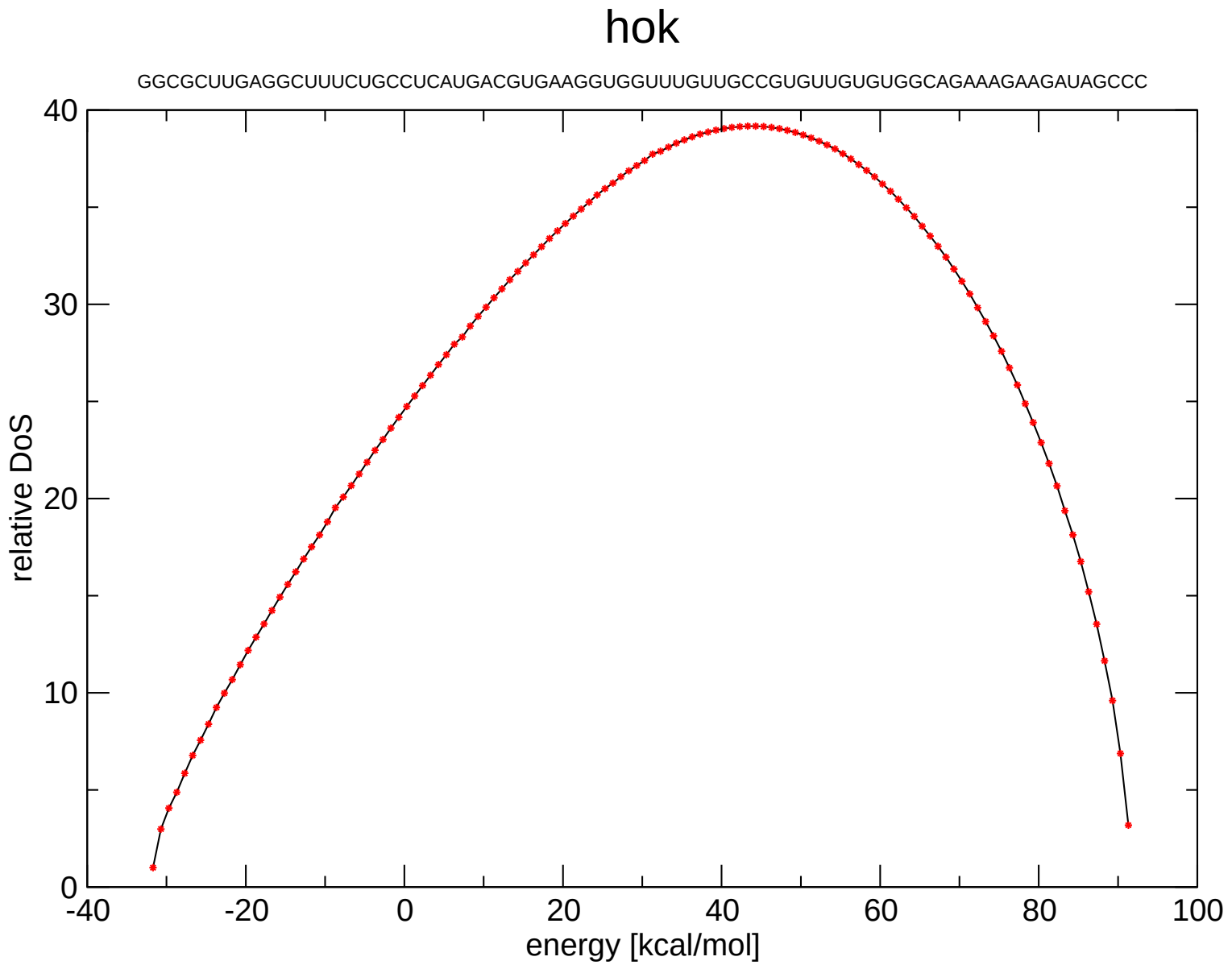
Wang-Landau sampling - pseudocode

Require: a start structure x compatible with sequence $\mathfrak{S}$
Ensure: **for all** E : $g(E) \leftarrow 1$ and **for all** E : $h(E) \leftarrow 0$

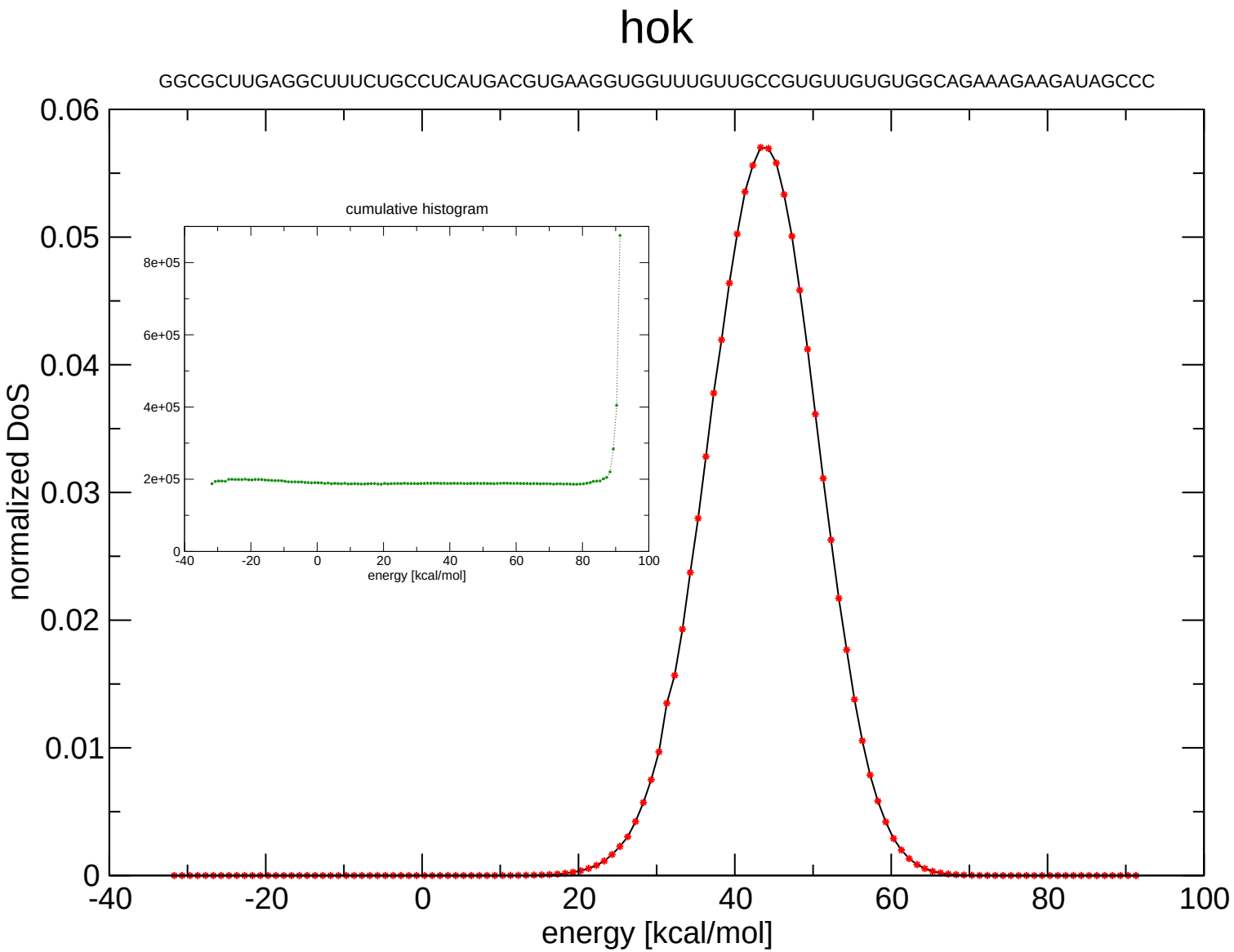
```
 1: f ← f_0 = exp(1)
 2: E_1 ← energy(x)
 3: 𝒩 ← generate_neighbors(x)
 4: repeat
 5:     y ← get_random_neighbor(𝒩)
 6:     E_2 ← energy(y)
 7:     ξ ← g(E_1)/g(E_2)
 8:     r ← random_number()  // from [0;1]
 9:     if r < ξ then  // accept the move
10:         E_1 ← E_2
11:         𝒩 ← generate_neighbors(y)
12:     end if
13:     g(E_1) ← g(E_1) * f
14:     h(E_1) ← h(E_1) + 1
15:     if histogram_is_flat() then
16:         f ← f^(1/2)
17:         reset_histogram()
18:     else
19:         goto 5
20:     end if
21: until f ← f_min ~ exp(10^-8)
```

tbi

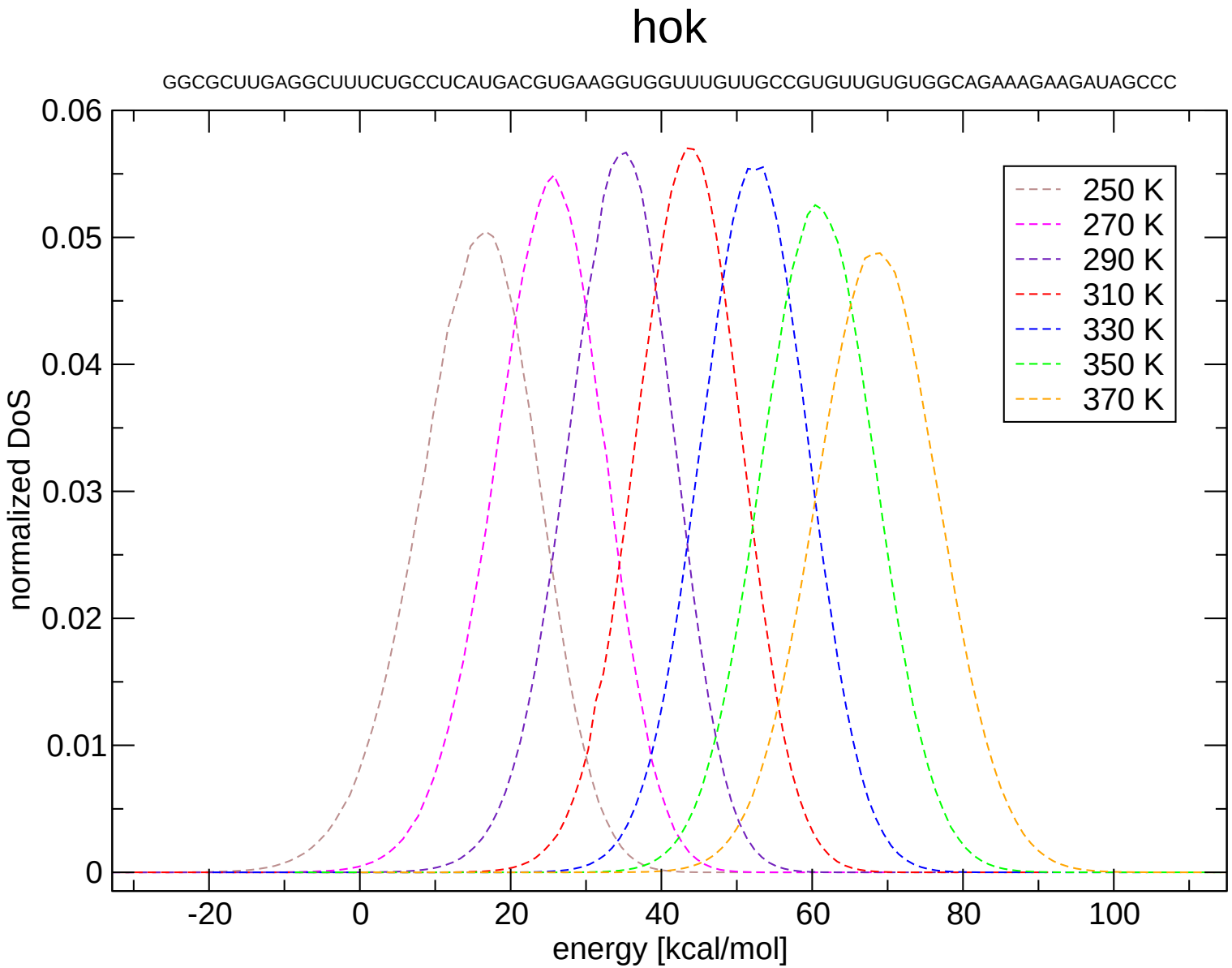
Wang-Landau - RNA example



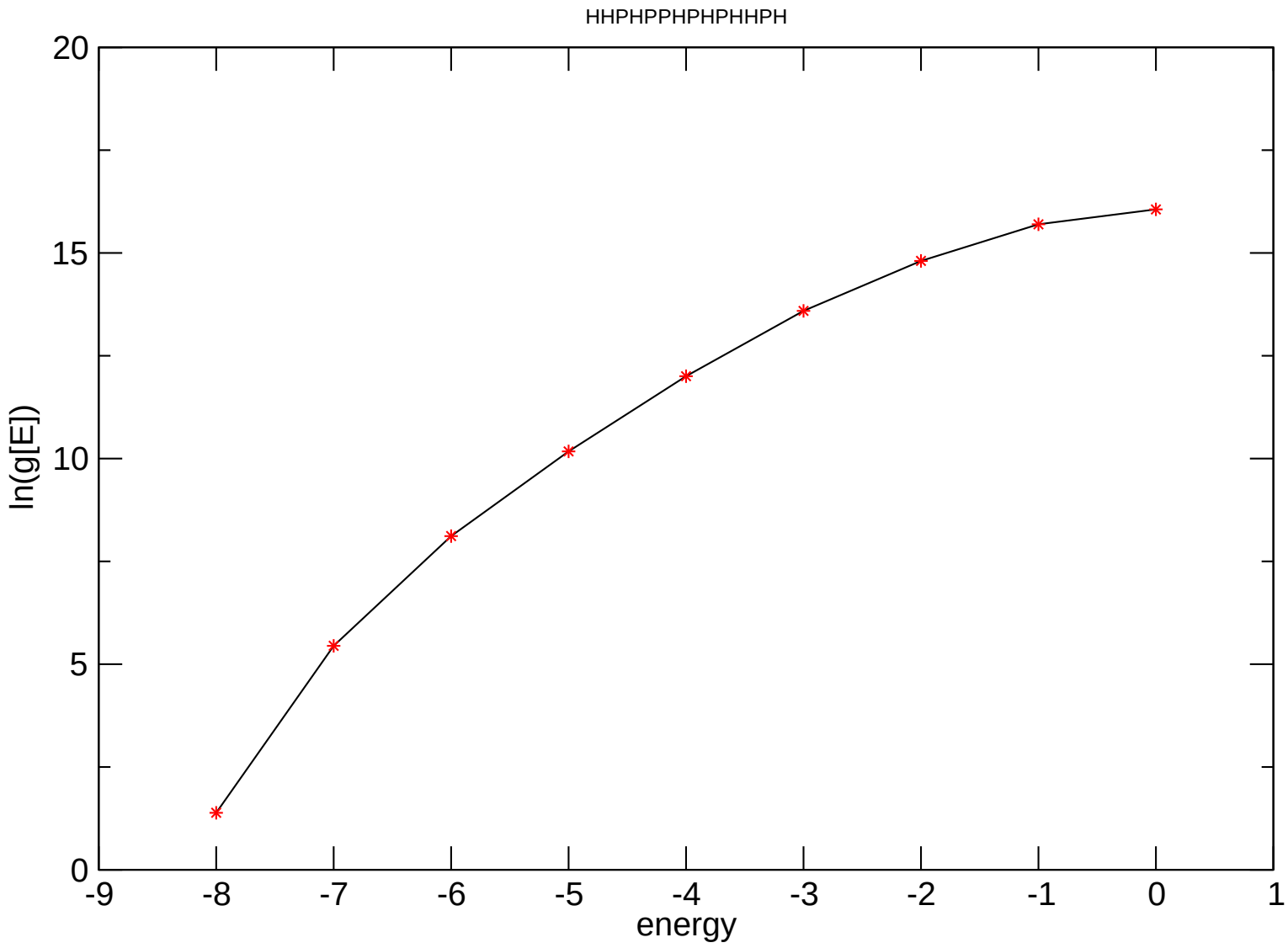
Wang-Landau - Density of States



Wang-Landau - DoS at different T



Wang-Landau - LP example



Thermodynamics

Relevant thermodynamic quantities can easily be calutaed from the DoS

$$U(T) = \frac{\sum_E E\, g(E)\, e^{-E/kT}}{\sum_E g(E)\, e^{-E/kT}} \equiv \langle E \rangle_T$$

$$C(T) = \frac{\partial U(T)}{\partial T} = \frac{\langle E^2 \rangle_T - \langle E \rangle_T^2}{kT^2}$$

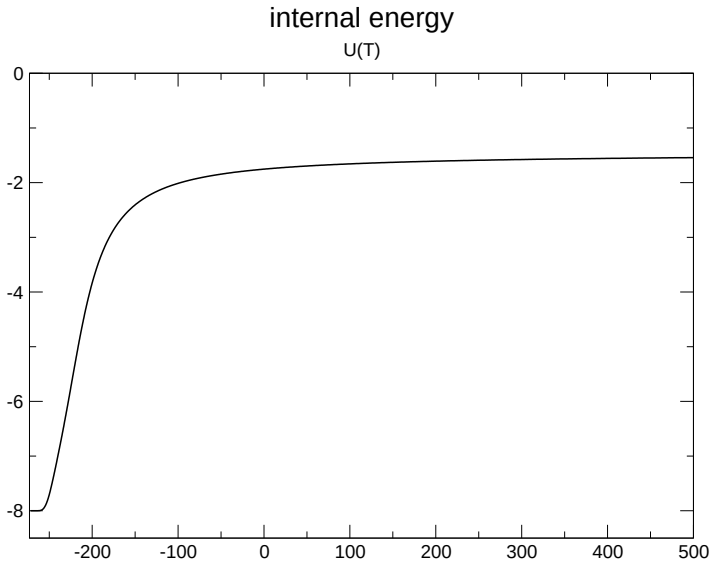
$$F(T) = -kT \ln(Z) = -kT \ln\left(\sum_E g(E)\, e^{-E/kT}\right)$$

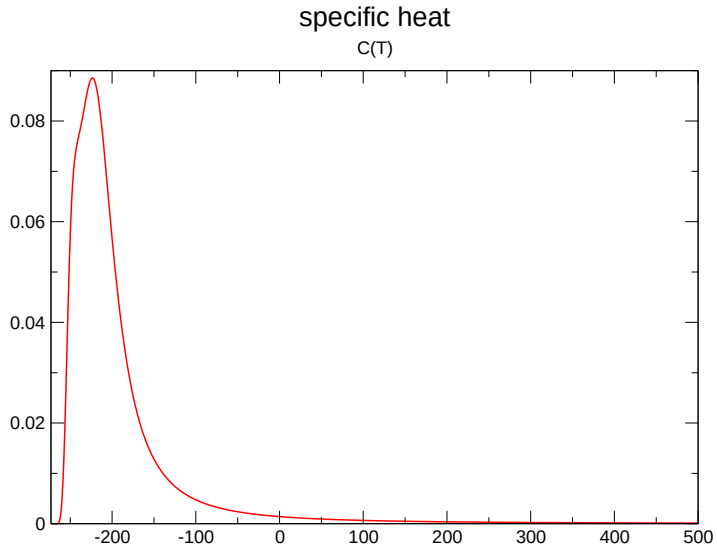
$$S(T) = \frac{U(T) - F(T)}{T}$$

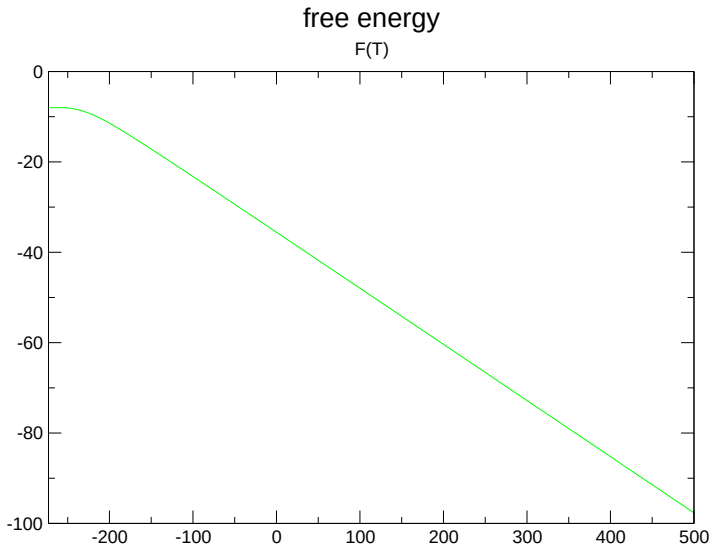
tbi

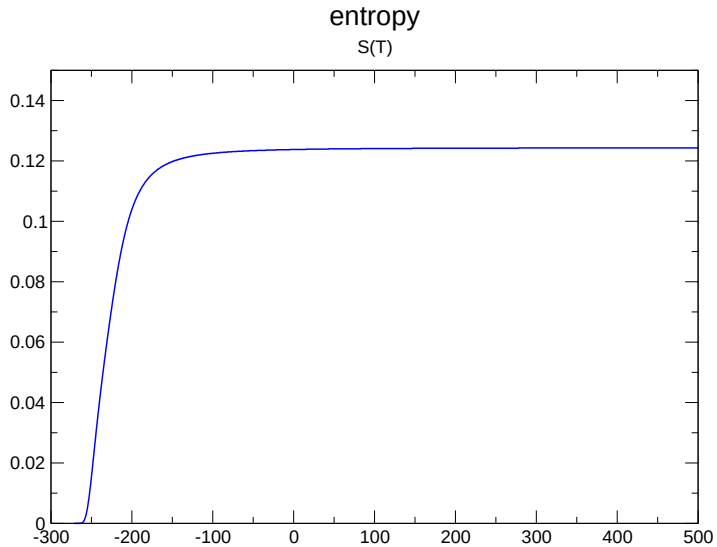
Thermodynamics of a short, artificial LP

HHPHPPHPHPHHPH $n = 14$

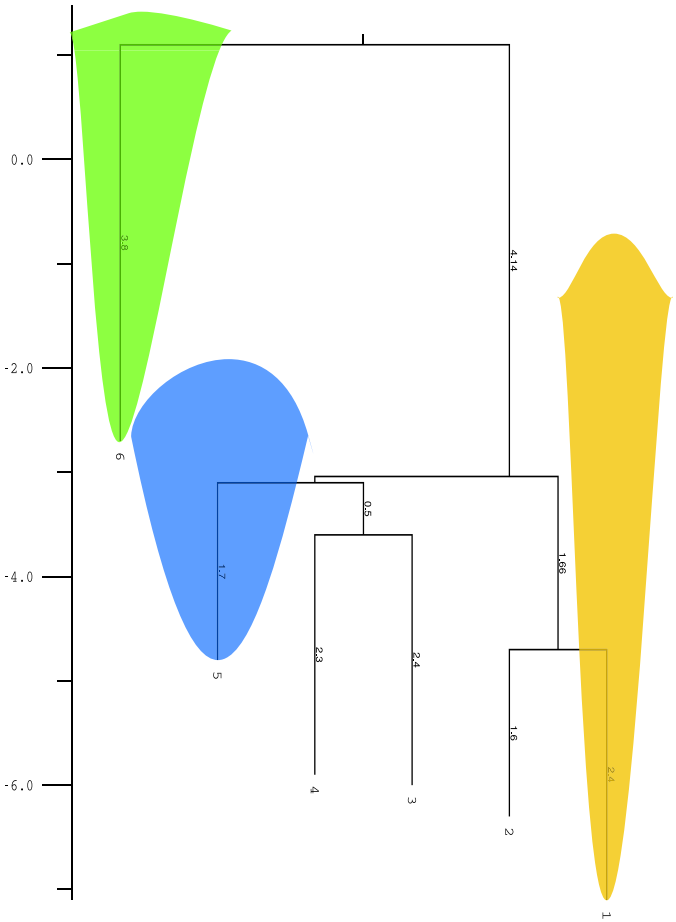








Basin sampling




```
1:  f ← f_0 = exp(1)
2:  E_1 ← energy(x)
3:  α ← get_gradient_basin(x)
4:  𝒩 ← generate_neighbors(x)
5:  repeat
6:      y ← get_random_neighbor(𝒩)
7:      E_2 ← energy(y)
8:      ξ ← g(E_1)/g(E_2)
9:      r ← random_number()   // from [0;1]
10:     if r < ξ then   // accept the move
11:         β ← get_gradient_basin(y)
12:         if α != β then
13:             continue
14:         E_1 ← E_2
15:         𝒩 ← generate_neighbors(y)
16:     end if
17:     g(E_1) ← g(E_1) * f
18:     h(E_1) ← h(E_1) + 1
19:     if histogram_is_flat() then
20:         f ← f^(1/2)
21:         reset_histogram()
22:     else
23:         goto 6
24:     end if
25: until f ← f_min ~ exp(10^-8)
```

tbi

Conclusion

- Discrete models allow a detailed study of the energy surface.
- Barrier trees represent the landscape topology.
- A macrostate approach of folding kinetics reduces simulation time drastically.
- Wang-Landau sampling approximates the density of states and allows the calculation of thermodynamic quantities.
- Basin sampling is a promosing approach towards simulation of biologically relevant molecules.

Conclusion

- Discrete models allow a detailed study of the energy surface.
- Barrier trees represent the landscape topology.
- A macrostate approach of folding kinetics reduces simulation time drastically.
- Wang-Landau sampling approximates the density of states and allows the calculation of thermodynamic quantities.
- Basin sampling is a promosing approach towards simulation of biologically relevant molecules.

tbi

Conclusion

- Discrete models allow a detailed study of the energy surface.
- Barrier trees represent the landscape topology.
- A macrostate approach of folding kinetics reduces simulation time drastically.
- Wang-Landau sampling approximates the density of states and allows the calculation of thermodynamic quantities.
- Basin sampling is a promosing approach towards simulation of biologically relevant molecules.

Conclusion

- Discrete models allow a detailed study of the energy surface.
- Barrier trees represent the landscape topology.
- A macrostate approach of folding kinetics reduces simulation time drastically.
- Wang-Landau sampling approximates the density of states and allows the calculation of thermodynamic quantities.
- Basin sampling is a promosing approach towards simulation of biologically relevant molecules.

Conclusion

- Discrete models allow a detailed study of the energy surface.
- Barrier trees represent the landscape topology.
- A macrostate approach of folding kinetics reduces simulation time drastically.
- Wang-Landau sampling approximates the density of states and allows the calculation of thermodynamic quantities.
- Basin sampling is a promosing approach towards simulation of biologically relevant molecules.

Conclusion

- Discrete models allow a detailed study of the energy surface.
- Barrier trees represent the landscape topology.
- A macrostate approach of folding kinetics reduces simulation time drastically.
- Wang-Landau sampling approximates the density of states and allows the calculation of thermodynamic quantities.
- Basin sampling is a promosing approach towards simulation of biologically relevant molecules.

Acknowledgments & References

Christoph Flamm, Ivo Hofacker, Peter Stadler
Sebastian Will, Martin Mann


M. T. Wolfinger, W. A. Svrcek-Seiler, C. Flamm, I. L. Hofacker, and P. F. Stadler.
Efficient computation of RNA folding dynamics.
J. Phys. A: Math. Gen., 37(17):4731–4741, 2004.


M. T. Wolfinger, S. Will, I. L. Hofacker, R. Backofen, and P. F. Stadler.
Exploring the lower part of discrete polymer model energy landscapes.
Europhys. Lett., 74(4):725–732, 2006.


C. Flamm, I. L. Hofacker, P. F. Stadler, and M. T. Wolfinger.
Barrier trees of degenerate landscapes.
Z. Phys. Chem., 216:155–173, 2002.


D. P. Landau, S. H. Tsai, and M. Exler.
A new approach to Monte Carlo simulations in statistical physics: Wang-Landau sampling.
Am. J. Phys., 72(10):1294–1302, 2004.

tbi