HERD: The Highest Expected Reward Decoding for HMMs with Application to Recombination Detection

Broňa Brejová

Department of Computer Science

Comenius University in Bratislava

Joint work with Michal Nánási and Tomáš Vinař

Sequence annotation

Label each symbol in the input sequence by its function

Example: Finding genes (protein-coding regions) in DNA

Input: DNA sequence $X = x_1, \ldots, x_n$

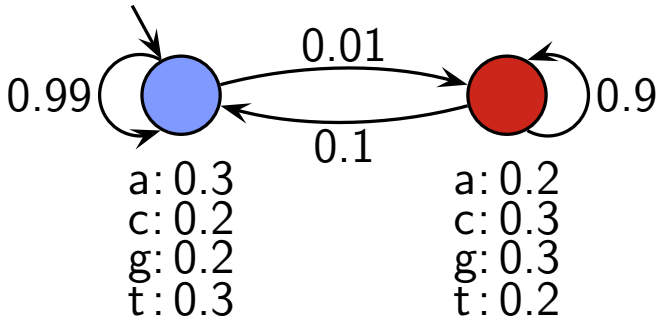
```
tgggcgtatttgcgctagtgttgggtgttccgctgtgctgttttccgtcatggctcgca
gtaaactacctttccagcgcctggtgcgcgagattgcgcaggactttaaaacagacctgc
acatccagctcgcccgccgcatccgcggagagagggcgtgattactgtggtctctctgac
```

Output: sequence of labels (colors) $A = a_1, \ldots, a_n$

red: protein coding, blue: non-coding

```
tgggcgtatttgcgctagtgttgggtgttccgctgtgctgttttccgtcatggctcgca
gtaaactacctttccagcgcctggtgcgcgagattgcgcaggactttaaaacagacctgc
acatccagctcgcccgccgcatccgcggagagagggcgtgattactgtggtctctctgac
```

Hidden Markov models (HMMs)



Sequence: $\{a, c, g, t\}^*$

$$X = x_1, x_2, \ldots, x_n$$

Annotation: $\{\blacksquare, \blacksquare\}^*$

$$A = a_1, a_2, \ldots, a_n$$

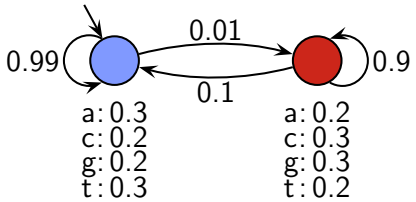
tatttagcgtcttctatcatccaatcactgcactttacacactataaatagagcagctca
tgggcgtatttgcgctagtgttgggtgttccgctgtgctgttttccgtcatggctcgca
ctaagcaaactgctcggaagtctactggtggcaaggcgccacgcaaacagttggccacta

HMM defines $P(X, A)$ for strings X and annotations A:

$$\begin{aligned} P(X, A) &= P(a_1)P(x_1|a_1)P(a_1 \to a_2)P(x_2|a_2)\cdots \\ &\quad P(a_{n-1} \to a_n)P(x_n|a_n) \end{aligned}$$

Viterbi algorithm [Forney 1973]

Input: string $X = x_1, \ldots, x_n$ and an HMM




```
tatttagcgtcttctatcatccaatcactgcactttacacactataaatagagcagctca
tgggcgtatttgcgctagtgttgggtgttccgctgtgctgttttccgtcatggctcgca
ctaagcaaactgctcggaagtctactggtggcaaggcgccacgcaaacagttggccacta
```

Goal: Find **the most probable annotation** $A^* = \arg\max_A P(X, A)$

```
tatttagcgtcttctatcatccaatcactgcactttacacactataaatagagcagctca
tgggcgtatttgcgctagtgttgggtgttccgctgtgctgttttccgtcatggctcgca
ctaagcaaactgctcggaagtctactggtggcaaggcgccacgcaaacagttggccacta
```

Simple dynamic programming

$O(n|E|)$ time (n = sequence length, $|E|$ = number of transitions)

This talk

- Viterbi is not the only algorithm for decoding HMMs
- Our new method: HERD, the highest expected reward decoding
- Motivated by application to HIV recombination detection

Gain function [Hamada et al. 2009]

$G(A, A')$ measures accuracy of A wrt. correct annotation A'

Examples:

Identity: score 1 iff A completely correct, 0 otherwise

Pointwise: score +1 for every correct label in A

Boundary: score +1 for every correct boundary, $-\gamma$ for incorrect boundary

	Identity	Pointwise	Boundary
$A =$  $A' =$ 	1	5	4
$A =$  $A' =$ 	0	4	$3-\gamma$

Optimizing expected gain

Goal: find annotation A that maximizes

$$E_{A'|X}[G(A, A')] = \sum_{A'} G(A, A')P(A'|X)$$

Identity gain function: Viterbi algorithm

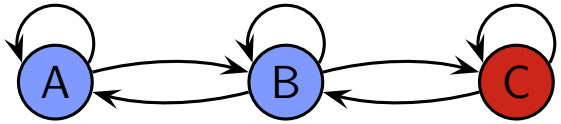
Pointwise gain function: Posterior decoding (forward-backward)

Boundary gain function: [Gross et al. 2007]

The choice of gain function is application-dependent

More complex HMMs

May have more states of the same color



Annotation: a set of state paths

$$P(\blacksquare\square\square\square) = P(CBBB) + P(CBBA) + P(CBAB) + P(CBAA)$$

$$P(\blacksquare\blacksquare\blacksquare\blacksquare) = P(CCCC)$$

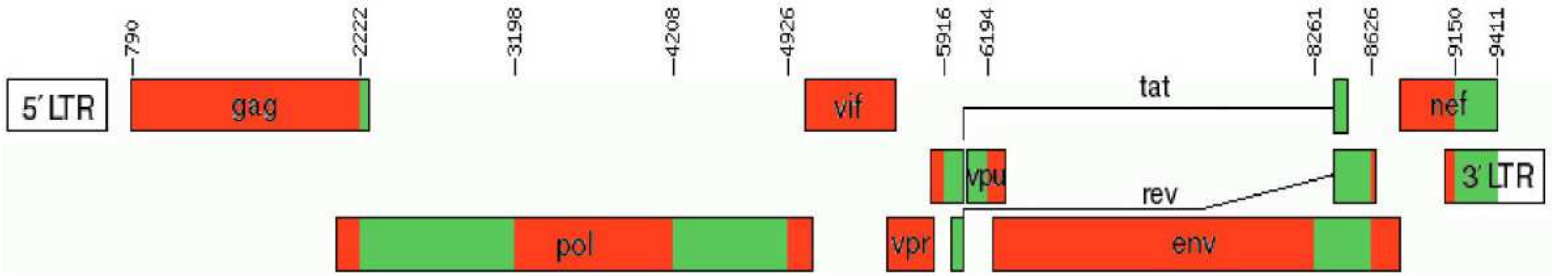
- Viterbi algorithm finds **the most probable state path**,
  not the most probable annotation
- Finding most probable annotation is NP-hard
  [Lyngso and Pedersen, 2002]
- Pointwise and boundary gain functions can be optimized efficiently

Our application: HIV recombination detection

Different subtypes of HIV (A, B, C, . . .)

Sometimes genomes recombine

Labels: one for each subtype



Source: Schultz et al 2006; subtypes A and G

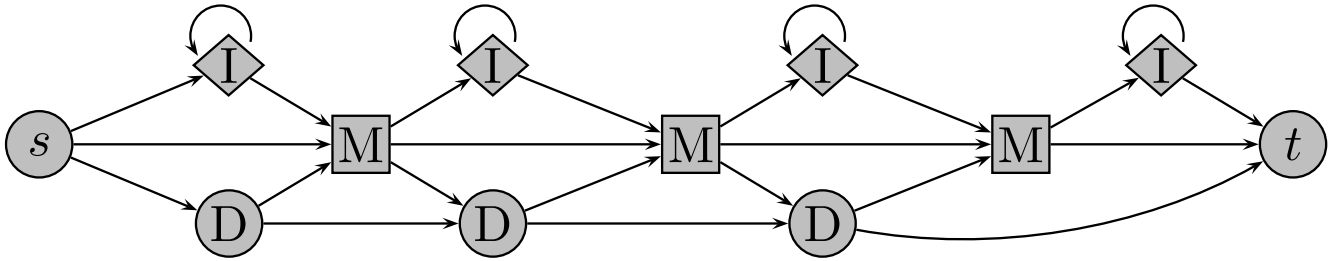
Jumping HMMs for HIV recombination detection

[Schultz et al 2006]

Profile HMM for each subtype

— represents multiple alignment of sequences in the subtype

— match, insert, delete state for each alignment column

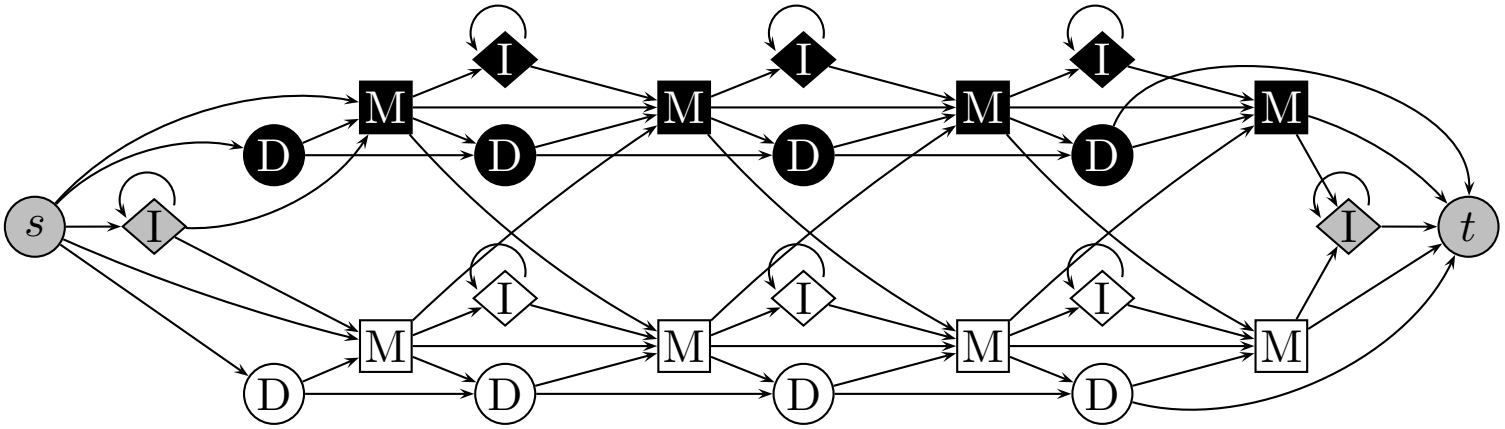



```
TTTTGGCTGAGGCAATGAGCCAAGCAACAAATGC
TTTTGGCCGAGGCAATGAGTCAAGCA---AATTC
TTTTGGCTGAGGCAATGAGCCAAGCA---AATAC
```

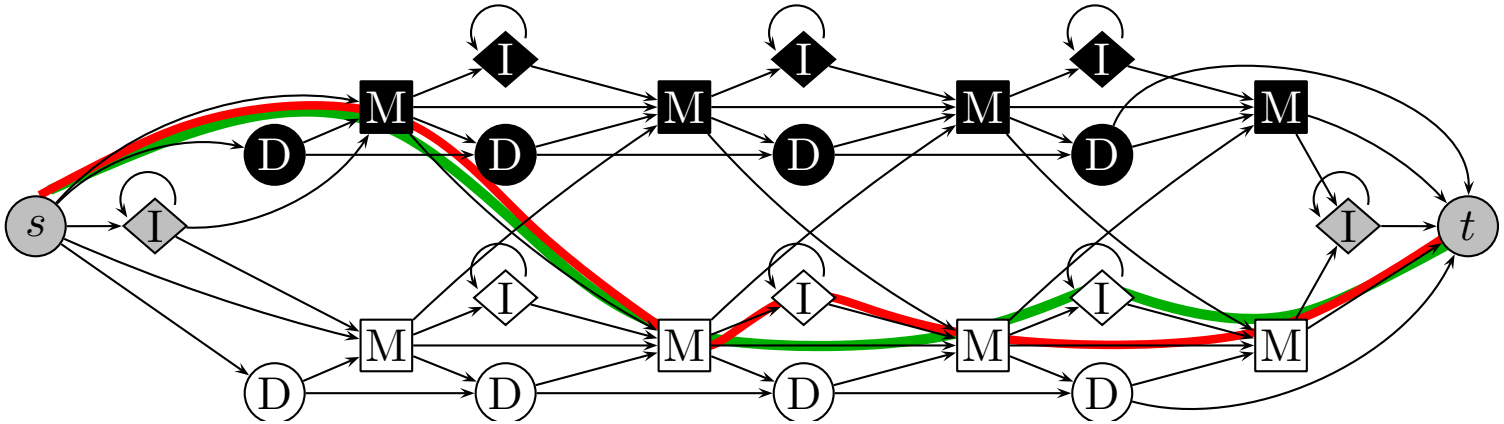
Jumping HMMs for HIV recombination detection (cont.)

[Schultz et al 2006]

- Profile HMM for each subtype
- Profiles connected by jumping transitions



Annotation issues in jumping HMMs



State path: alignment of sequence to subtype profiles

Annotation: segments of input emitted by subtype profiles

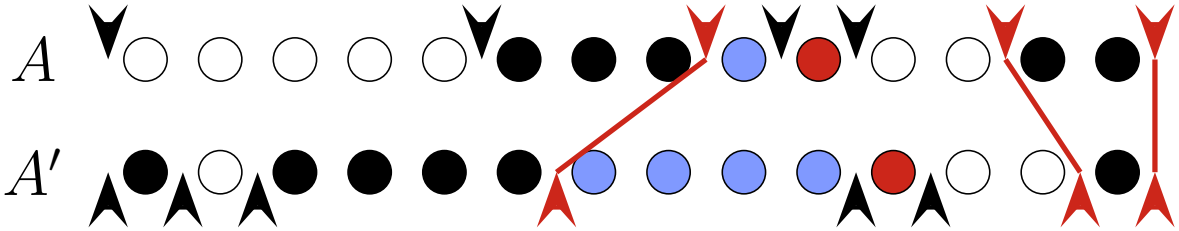
- Many state paths for each annotation
- Many annotations with slightly shifted boundaries

Change the objective function for decoding

HERD: highest expected reward decoding

Boundaries are **buddies** if they are closer than W,

and their flanking regions have the same colors and overlap



gain: $+3 - 4\gamma$

Our gain function $G(A, A')$:

+1 for each boundary in A with a buddy in A'

$-\gamma$ for each boundary without a buddy

Appropriate for recombination detection

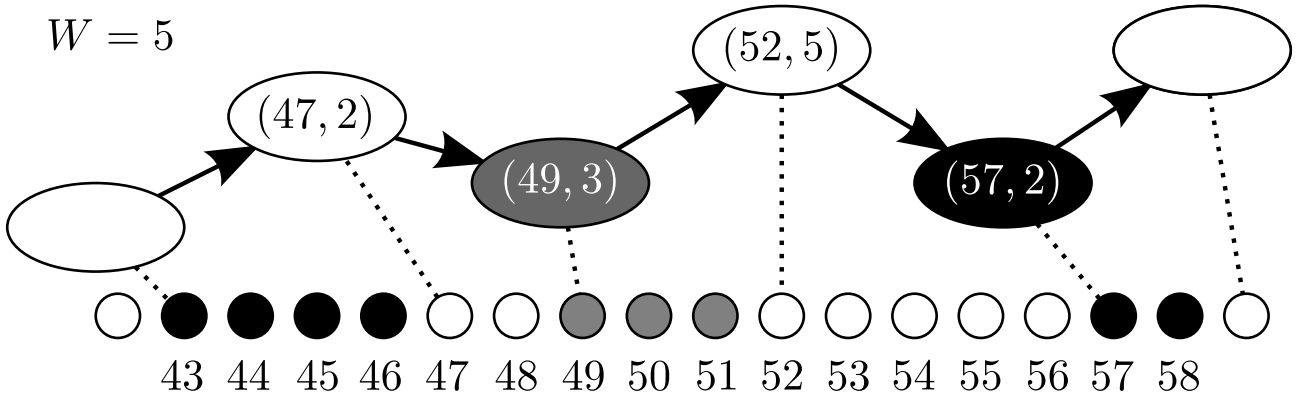
Similar gain function [Brown and Truszkowski 2010]:

score 1 if all boundaries have a buddy

HERD: highest expected reward decoding

Our gain function can be optimized in $O(nW|E| + nW^2C^2)$:

- Compute $P(a_{i...j} = c_1^{j-i}c_2 \,|\, X)$ by a modified forward/backward alg.
- Compute expected gain at each potential boundary
- Find the highest-weight path in a DAG



Comparison on HIV recombination data

Algorithm	% bases correct	Feature sp.	Feature sn.
Artificial recombinants, $P_j = 10^{-5}$			
HERD $W = 10, \gamma = 1$	95.5%	62.1%	57.6%
HERD $W = 1, \gamma = 0.1$	81.3%	37.2%	29.3%
Viterbi	96.3%	51.8%	48.2%
Posterior	95.8%	60.4%	57.7%
No recombination			
HERD, $W = 10, \gamma = 1, P_j = 10^{-9}$	100.0%	100.0%	100.0%
HERD, $W = 10, \gamma = 1, P_j = 10^{-5}$	93.7%	83.9%	83.9%
Viterbi, $P_j = 10^{-9}$ or $P_j = 10^{-5}$	100.0%	100.0%	100.0%

Conclusion

- Many options for decoding HMMs
- Choice application dependent (gain function)
- Some choices lead to NP-hard problems
- Our gain function appropriate for recombination detection, polynomial-time algorithm
- Problem: how to choose parameters, e.g. W and γ
- Possible extensions:
  – closer buddies score more
  – combine with reward for correct bases
- **http://compbio.fmph.uniba.sk/herd/**