A NEW GENERATION OF HOMOLOGY SEARCH TOOLS BASED ON PROBABILISTIC INFERENCE

SEAN R. EDDY[1]
eddys@janelia.hhmi.org

[1] *Janelia Farm Research Campus, Howard Hughes Medical Institute, 19700 Helix Drive, Ashburn VA 20147, USA*

Many theoretical advances have been made in applying probabilistic inference methods to improve the power of sequence homology searches, yet the BLAST suite of programs is still the workhorse for most of the field. The main reason for this is practical: BLAST's programs are about 100-fold faster than the fastest competing implementations of probabilistic inference methods. I describe recent work on the HMMER software suite for protein sequence analysis, which implements probabilistic inference using profile hidden Markov models. Our aim in HMMER3 is to achieve BLAST's speed while further improving the power of probabilistic inference based methods. HMMER3 implements a new probabilistic model of local sequence alignment and a new heuristic acceleration algorithm. Combined with efficient vector-parallel implementations on modern processors, these improvements synergize. HMMER3 uses more powerful log-odds likelihood scores (scores summed over alignment uncertainty, rather than scoring a single optimal alignment); it calculates accurate expectation values (E-values) for those scores without simulation using a generalization of Karlin/Altschul theory; it computes posterior distributions over the ensemble of possible alignments and returns posterior probabilities (confidences) in each aligned residue; and it does all this at an overall speed comparable to BLAST. The HMMER project aims to usher in a new generation of more powerful homology search tools based on probabilistic inference methods.

Keywords: sequence alignment; homology search; hidden Markov models

1. Introduction

Sequence homology searches are one of the most important application areas in computational molecular biology. Algorithms and software tools for detecting distantly related sequences are as important in molecular biology as telescopes are in astronomy, allowing us to look back deeply in time at sequence evolution. The BLAST suite of programs has been the workhorse of homology searches since the early 1990's [2, 3]. Since the advent of BLAST, the field has made a number of theoretical advances, particularly in the use of probabilistic models (such as hidden Markov models, HMMs) to parameterize complex position-specific models, to integrate multiple sources of information consistently, and to frame the homology detection problem more formally and powerfully as a statistical inference problem with log-likelihood statistics [6, 14]. However, implementations of these HMM-based methods have suffered from the fact that they are about two orders of magnitude slower than BLAST, because they implement full dynamic programming methods (akin to full Smith/Waterman alignment [18]) without the heuristic accelerations that make BLAST fast.

The goal of the HMMER3 project is to achieve a widely available practical implementation competitive with BLAST's speed, while further extending the power of probabilistically based methods.

2. Sequence Homology Search as a Statistical Inference Problem

From a statistical inference perspective, we can frame the homology search problem as a test of two hypotheses: is a target sequence $\mathbf{x}$ more likely to be a homolog of our query sequence (or query alignment) $\mathbf{y}$ (call this hypothesis $H_{\mathbf{y}}$) or is $\mathbf{x}$ more likely to be a nonhomologous "random" sequence (call this hypothesis R, our null hypothesis)? Theory says if this is the problem, we should aim to calculate a log-odds likelihood score [6, 7, 16]:

$$S(\mathbf{x} \mid \mathbf{y}) = \log \frac{P(\mathbf{x} \mid H_{\mathbf{y}})}{P(\mathbf{x} \mid R)}.$$

Traditionally, though, we calculate an *alignment score*, where each individual residue of $\mathbf{x}$ is scored either as a homolog of an individual residue in $\mathbf{y}$ (by assigning a score from a substitution matrix $\sigma(a,b)$ for an aligned homologous residue pair a,b), or as an insertion relative to $\mathbf{y}$. Insertions (and deletions in $\mathbf{y}$ relative to $\mathbf{x}$) are scored using arbitrary gap penalties, usually consisting of a gap-open penalty for starting an insertion/deletion and a gap-extend penalty for each residue in the insertion/deletion.

An alignment score depends on specifying a particular alignment. Thus even if we had a probabilistic model instead of a model with arbitrary gap penalties, an alignment-dependent homology model would at best be specifying a score that involves a joint probability $P(\mathbf{x}, \pi \mid H_{\mathbf{y}})$ for a particular alignment π, rather than the probability $P(\mathbf{x} \mid H_{\mathbf{y}})$ we are really interested in. The alignment π is a so-called *nuisance variable* in the inference problem, and theory says we should sum over (marginalize) alignments to obtain the likelihood for $\mathbf{x}$:

$$P(\mathbf{x} \mid H_{\mathbf{y}}) = \sum_{\pi} P(\mathbf{x}, \pi \mid H_{\mathbf{y}}).$$

Traditional alignment scoring methods cannot do this summation, because it only makes sense if the terms $P(\mathbf{x}, \pi \mid H_{\mathbf{y}})$ are probabilities that can be meaningfully summed. The use of arbitrary gap-open and gap-extend penalties, among other things, prevents meaningful summation. Traditional methods instead calculate the score of an optimal alignment $\mathring{\pi}$, thus implicitly making the assumption that

$$P(\mathbf{x} \mid H_{\mathbf{y}}) \approx P(\mathbf{x}, \mathring{\pi} \mid H_{\mathbf{y}}),$$

or in other words, that all the probability mass is concentrated in this single optimal alignment – that we have no uncertainty about the correctness of the optimal alignment.

An assumption of no alignment uncertainty might be reasonable for closely related sequences, but it breaks down on precisely the sequences we are most concerned with detecting. The most remote homologs are the most difficult to detect and the most difficult to align with certainty. On theoretical grounds, we should expect that the optimal-alignment-dependent assumptions of traditional methods compromise our ability to detect remote homologs.

Using hidden Markov models, one can express $P(\mathbf{x} \mid H_{\mathbf{y}})$ making the same independence assumptions that are made by traditional sequence alignment scoring, but in a fully probabilistic model where the necessary summation over alignments is meaningful. The HMM dynamic programming algorithm to calculate this sum is called the "Forward" algorithm, as opposed to the HMM "Viterbi" algorithm which is the close analog of traditional optimal alignment algorithms. Using Forward, we can calculate a log-odds likelihood score, summed over alignment uncertainty.

The advantages of using probabilistic alignment models and summing over alignments have long been recognized. Two popular profile HMM packages are in widespread use, HMMER and SAM [12], and the SAM package has long used Forward scores by default. "Pair hidden Markov models" (pair-HMMs) have been used to express a probabilistic model of local sequence alignment [6, 9]. It is also possible to treat traditional alignment scores as unnormalized log probabilities (akin to free energies) and implement "probabilistic" local alignment using partition function calculations for renormalization [4, 15]. However, it is nontrivial to implement a probabilistic model that does not make undesirable assumptions about local alignment distributions. For example, pair-HMMs are usually made with a single "match" state that emits aligned residue pairs, with a self-transition probability to emit subsequent aligned pairs; this implies that ungapped local alignment segment lengths are geometrically distributed, with length 1 being most probable. In contrast, Smith/Waterman local alignment [18] scores the start and end of any local alignment as zero cost regardless of local alignment segment length, corresponding to an unnormalized uniform distribution, and the uninformative uniform distribution is arguably the better assumption.

The version 3.0 implementation of HMMER (HMMER3) is based on an improved probabilistic local alignment model, with parameters and assumptions that can be mapped essentially one-to-one onto the parameters and assumptions of local Smith/Waterman sequence alignment, but where all the parameters are probabilistic so that the Forward algorithm can be implemented and full log-odds likelihood scores can be calculated. Perhaps most importantly, the HMMER3 local alignment model appears to simplify expectation value (E-value) calculations, as described in the following section.

3. E-value Statistics of Log-Odds Likelihood Scores

Scores are not enough. It is also essential to be able to calculate the statistical significance of a score: the E-value (expectation value), the expected number of times we would see a score this high by chance if the database we searched were entirely composed of nonhomologous "random" sequences.

For *ungapped* local alignment scores, Karlin/Altschul theory [10, 11] (one of the foundations of BLAST's power and success) specifies an analytical equation for the E-value of a score in terms of a Gumbel (extreme value) distribution controlled by two parameters K and λ, where λ is the base of the log of the log-odds substitution matrix scoring system ($\lambda = \log 2 = 0.693$, if the scoring matrix is in units of bits), and K can be calculated by a quick recursion.

For *gapped* local alignment scores using arbitrary insertions and deletion penalties, empirical results show that Karlin/Altschul statistics still approximately hold [1, 3], so long as the gap penalties are not too permissive. However, the λ parameter is no longer a known constant; instead it must be fitted by simulation to the distribution of scores of many random sequences. So long as only a few scoring systems are in use (such as the default use of the BLOSUM62 score matrix with particular choices of gap-open and gap-extend penalties in BLAST), a λ parameter can be precalculated for each scoring system. Even for profile position-specific scoring systems, so long as the arbitrary gap penalties are constant and position-independent (as in PSI-BLAST), an appropriate λ may be precalculated. However, once gap penalties are made position-specific, it has appeared that a costly simulation is needed to determine λ. HMMER2, for example, required a costly *hmmcalibrate* run to "calibrate" a λ parameter for every new profile HMM that might be used as a query.

The failure to be able to determine λ efficiently for optimal alignment scores is bad enough, but worse, Karlin/Altschul theory *only* applies to optimal alignment scores, not to the log-odds likelihood Forward scores we should prefer to use. The distribution of Forward scores is not expected to be a Gumbel extreme value distribution (which arises when one takes the maximum over a large number of possibilities drawn from some underlying distribution, as is the case for optimal local alignment score drawn from a Poisson distribution over possible alignment scores); and indeed, empirically, Forward scores are clearly not Gumbel distributed [7, 13].

Recent results – particularly from Terry Hwa, Ralf Bundschuh, and their collaborators [1, 5, 17, 19] – suggested a way forward. Based on their work, last year I put forward conjectures about the statistical distributions of Viterbi and Forward scores for HMMER3's probabilistic local alignment model [7]: first, that λ would be $\log 2$ just as in ungapped alignment (essentially because $\lambda = \log 2$ is conjectured to be a property of any fully probabilistic model's Viterbi scores, including the fully probabilistic ungapped alignment model studied by Karlin and Altschul as well as a much more general class of alignment models including HMMER3's); and second, that Forward scores should asymptotically converge to an exponential tail of the same slope $\lambda = \log 2$. These conjectures were shown by numerical simulation (though not by formal proof) to hold for a wide range of different query models and target sequences [7].

HMMER3 is therefore able to calculate E-values for Forward log-odds likelihood scores, without any need for computationally expensive simulations.

4. Heuristic Acceleration of Profile HMM Algorithms

The main drawback of profile HMM implementations has been that they are about 100x slower than BLAST. Unfortunately, with respect to speed, scoring with the Forward algorithm is a step in the wrong direction. A good implementation of the Forward algorithm is typically about three-fold slower than an implementation of Viterbi optimal alignment. To use Forward practically – indeed, to be competitive with BLAST at all with respect to speed – profile HMM implementations need dramatic speed improvement.

However, it is not the case that HMM-based models are necessarily slow relative to traditional scoring methods. The main difference between HMMs and traditional scoring is in model parameterization, not so much in the algorithms used. All local alignment dynamic programming algorithms are slow. The Viterbi algorithm with a local alignment HMM is essentially identical to the Smith/Waterman dynamic programming algorithm [18]. In the case of traditional scoring, heuristic acceleration methods like BLAST and FASTA were implemented to approximate Smith/Waterman scores. In the case of profile HMM methods, there have yet been no widely used implementations of heuristic accelerations.

HMMER3 implements a new heuristic acceleration algorithm called MSV (Multiple ungapped Segment Viterbi). It creates a simplified version of its local alignment model, implicitly removing insertion and deletion states and setting match-match transition probabilities to 1.0. This model generates (aligns to) multiple ungapped local alignment segments. The MSV score essentially corresponds to a BLAST sum score of multiple ungapped local alignments, but BLAST's word hit and hit extension steps are bypassed because HMMER calculates its MSV score directly by dynamic programming. Because these two main BLAST heuristics are obviated, the HMMER MSV heuristic should be more sensitive than BLAST's overall set of heuristics, and empirically this appears to be the case. A key to HMMER3's speed is that the MSV alignment scoring algorithm can be implemented very efficiently using vector-parallel instructions available on modern processors (SSE instructions on Intel-compatible platforms; Altivec instructions on PowerPC platforms) in greatly reduced precision (single unsigned bytes).

The MSV algorithm calculates only an approximate local alignment score, so HMMER3 uses it only as a heuristic filter. Because the MSV score results from a fully probabilistic (albeit simplified) model, the same statistical conjectures described above for optimal local alignment score distributions hold, and a P-value for an MSV score can be calculated easily. This allows setting a well-principled filtering threshold, where the default is to allow sequences with $P < 0.02$ (expected 2% of random sequences) through the filter. These high-scoring (low P-value) sequences are passed on to more powerful filters, and ultimately to a full Forward/Backward HMM calculation that yields the desired log-odds likelihood score, an inferred domain structure of the target protein sequence, an "optimal expected accuracy" alignment, and posterior probabilities (confidences) for each aligned residue.

Because the MSV filter only reduces the sequence search space by 50-fold at the default P-value threshold of 0.02, but the full Forward/Backward algorithm is about 1000-fold slower than MSV, it was also necessary to implement vectorized versions of all steps of the HMMER3 sequence processing pipeline, including the Forward/Backward algorithm itself, where we obtained about a 20x speedup using SSE2 vector instructions and a "sparse rescaling" technique for performing calculations in probabilities rather than in log probabilities, as is usually done for numerical floating point overflow/underflow reasons.

Overall, HMMER3 searches typically run at about BLAST speed; usually slightly faster than WU-BLAST and somewhat slower (about 3-fold) than NCBI BLAST.

5. Conclusion

As of this writing, HMMER3 is freely available in a public beta test release (3.0b2; `ftp://selab.janelia.org/pub/software/hmmer3/hmmer-3.0b2.tar.gz`), and it is discussed on a web site (`hmmer.org`) and a blog (`cryptogenomicon.org`).

The theory encompasses not only profile searches (position-specific multiple alignment models) but also single sequence queries as a special case. Enabled by HMMER3's new speed, HMMER3 now includes applications not just in the usual profile HMM niche for searching profile databases such as Pfam [8], but also directly competitive in BLAST's traditional niches of single sequence search (HMMER3's phmmer is analogous to BLAST's blastp) and iterative search (HMMER3's jackhmmer is analogous to PSI-BLAST).

Our internal benchmarking shows HMMER3's search programs to be a significant advance in remote homolog detection over other sequence homology searching tools, while having essentially the same speed as BLAST applications. However, we are cautious about the difficulty of doing unbiased internal benchmarking, and it will be interesting to see the results of independent benchmarking.

The HMMER project aims to usher in a new generation of practical and powerful homology search tools. Our future plans include applications for DNA comparisons and genome analysis, freely available high-performance implementations for a variety of modern processors including general-purpose graphics processing units (GP-GPUs), and the establishment of freely available public servers for routine database searches with these new tools.

References

[1] S. F. Altschul, R. Bundschuh, R. Olsen, and T. Hwa. The estimation of statistical parameters for local alignment score distributions. *Nucl. Acids Res.*, 29:351–361, 2001.

[2] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215:403–410, 1990.

[3] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucl. Acids Res.*, 25:3389–3402, 1997.

[4] P. Bucher and K. Hofmann. A sequence similarity search algorithm based on a probabilistic interpretation of an alignment scoring system. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 4:44–51, 1996.

[5] R. Bundschuh. Rapid significance estimation in local sequence alignment with gaps. *J Comput. Biol.*, 9:243–260, 2002.

[6] R. Durbin, S. R. Eddy, A. Krogh, and G. J. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids.* Cambridge University Press, Cambridge UK, 1998.

[7] S. R. Eddy. A probabilistic model of local sequence alignment that simplifies statistical significance estimation. *PLoS Comput. Biol.*, 4:e1000069, 2008.

[8] R. D. Finn, J. Tate, J. Mistry, P. C. Coggill, S. J. Sammut, H.-R. Hotz, G. Ceric, K. Forslund, S. R. Eddy, E. L. L. Sonnhammer, and A. Bateman. The Pfam protein families database. *Nucl. Acids Res.*, 36:D281–D288, 2008.

[9] I. Holmes. *Studies in Probabilistic Sequence Alignment and Evolution.* PhD thesis, University of Cambridge, 1998.

[10] S. Karlin and S. F. Altschul. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl. Acad. Sci. USA*, 87:2264–2268, 1990.

[11] S. Karlin and S. F. Altschul. Applications and statistics for multiple high-scoring segments in molecular sequences. *Proc. Natl. Acad. Sci. USA*, 90:5873–5877, 1993.

[12] K. Karplus, C. Barrett, and R. Hughey. Hidden Markov models for detecting remote protein homologies. *Bioinformatics*, 14:846–856, 1998.

[13] K. Karplus, R. Karchin, G. Shackelford, and R. Hughey. Calibrating E-values for hidden Markov models using reverse-sequence null models. *Bioinformatics*, 21:4107–4115, 2005.

[14] A. Krogh, M. Brown, I. S. Mian, K. Sjölander, and D. Haussler. Hidden Markov models in computational biology: Applications to protein modeling. *J. Mol. Biol.*, 235:1501–1531, 1994.

[15] S. Miyazawa. A reliable sequence alignment method based on probabilities of residue correspondences. *Protein Eng.*, 8:999–1009, 1995.

[16] J. Neyman and E. S. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Phil. Trans. Royal Soc. London A*, 231:289–337, 1933.

[17] R. Olsen, R. Bundschuh, and T. Hwa. Rapid assessment of extremal statistics for gapped local alignment. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 7:211–222, 1999.

[18] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147:195–197, 1981.

[19] Y.-K. Yu, R. Bundschuh, and T. Hwa. Hybrid alignment: high-performance with universal statistics. *Bioinformatics*, 18:864–872, 2002.