BIOINFORMATICS APPLICATIONS NOTE Vol. 23 no. 11 2007, pages 1427–1428
doi:10.1093/bioinformatics/btm095

Sequence analysis

Ngila: global pairwise alignments with logarithmic and affine gap costs

Reed A. Cartwright*

Department of Genetics, University of Georgia, Athens, GA 30602-7223, USA
Present address: Bioinformatics Research Center, North Carolina State University, Campus Box 7566, Raleigh, NC 27695-7566, USA

Received on October 31, 2006; revised on February 12, 2007; accepted on March 7, 2007

Advance Access publication March 25, 2007

Associate Editor: John Quackenbush

ABSTRACT

Summary: Ngila is an application that will find the best alignment of a pair of sequences using log-affine gap costs, which are the most biologically realistic gap costs.

Availability: Portable source code for Ngila can be downloaded from its development website, http://scit.us/projects/ngila/. It compiles on most operating systems.

Contact: racartwr@ncsu.edu or reed@scit.us

Supplementary information: Appendices

1 INTRODUCTION

Over the last two decades, several molecular studies have demonstrated that indel lengths obey a power law, i.e. the logarithms of indel lengths are linearly related to the logarithms of their frequency (Benner *et al.*, 1993; Chang and Benner, 2004; Gonnet *et al.*, 1992; Gu and Li, 1995; Zhang and Gerstein, 2003). Taking this observation into account and using biologically accurate models of indel lengths can improve alignment accuracy. Under a Zipfian power law distribution, the probability that an observation is $x = \{1, 2, \ldots\}$ is $f(x|z) = x^{-z}/\zeta(z)$, where $z > 1$ is a parameter and $\zeta(z) = \sum_{n=1}^{\infty} n^{-z}$ is Riemann's Zeta Function. The mean and variance of this distribution are undefined (i.e. infinite) for $z \leq 2$ and $z \leq 3$, respectively.

Based on this power law, some researchers have argued that logarithmic gap costs, $G(k) = a + c \ln k$, may be appropriate for sequence alignments (Gonnet *et al.*, 1992; Gu and Li, 1995; Waterman, 1984). However, Cartwright (2006) has demonstrated that log-affine gap costs, $G(k) = a + bk + c \ln k$, are more accurate than both affine and logarithmic gap costs when indels follow a power law. Even though it is straight forward to implement monotonic gap costs using Miller and Myers (1988) and while Mott (1999) provided the application MONOTONE for finding local alignments using logarithmic costs, there are no applications available that can align sequences globally using logarithmic or log-affine gap costs. Here we present Ngila, a portable implementation of Miller and Myers (1988) that can globally align pairs of sequences using logarithmic and affine gap costs.

2 IMPLEMENTATION

Needleman and Wunsch (1970) provided a method of sequence alignment where the maximum similarity of a pair of sequences was calculated using dynamic programming. Sellers (1974) provided an alternative approach where the minimum distance of the sequence pair was calculated instead. These algorithms took $O(mn)$ time and $O(mn)$ space where m and n are the lengths of the two sequences being aligned. Waterman *et al.* (1976) generalized Sellers's metric to arbitrary gap weights, producing an $O(mn(m + n))$ algorithm, referred to as WSB. Gotoh (1982) demonstrated that with affine gap penalties the WSB algorithm can be run in $O(mn)$ time.

Waterman (1984) subsequently modified the WSB algorithm into the candidate-list method for concave gap weights, which could simplify the search process. Waterman's candidate list method was further refined by Miller and Myers (1988) into an $O(mn)$ algorithm. In addition, Miller and Myers (1988) pointed out how their algorithm could be modified using Hirschberg's (1975) divide-and-conquer method to produce alignments in $O(mn \lg(m + n))$ time but with $O(m)$ space.

Ngila implements the Miller and Myers (1988) algorithm in order to find a least costly global alignment of two sequences given homology costs and a gap cost of $w_k = a + bk + c \ln k$. Two versions of the algorithm are included: holistic and divide-and-conquer. The former is faster but the latter utilizes less memory. Ngila starts with the divide-and-conquer method but switches to the holistic method for subsequences smaller than a user-established threshold. This improves its speed without substantially increasing memory requirements. Ngila also allows users to assign costs to end gaps that are smaller than costs for internal gaps. This is important for aligning using the free-end-gap method.

3 STATISTICAL ALIGNMENT

The algorithm in Ngila finds the minimum cost of aligning two sequences (Sellers, 1974) instead of the maximum similarity of aligning two sequences (Needleman and Wunch, 1970).

*To whom correspondence should be addressed.

An extension of the method of Smith *et al.* (1981) can be used to convert a maximum search to a minimum search (Holmes and Durbin, 1998). This is important when using Ngila to find maximum likelihood alignments from a statistical model. One advantage of statistical models is the ability to estimate parameters without circularity (Thorne *et al.*, 1991).

Based on a statistical model, the scores of 'matches' of type i, α_i, and the penalties of gaps of length k, w_k, can be used to calculate the alignment with maximum log-likelihood: $l = \max\left\{\sum \alpha_i \eta_i - \sum w_k \Delta_k\right\}$, where η_i is the number of residue matches of type i and Δ_k is the number of gaps of length k. A minimum cost analog of this equation is $d = \min\left\{\sum \beta_i \eta_i + \sum G(k)\Delta_k\right\}$, where $\beta_i = (x - \alpha_i)/y$ is the cost of a match of type i, $G(k) = (xk/2 + w_k)/y$ is the cost of a gap of length k, and x and y are arbitrary constants (Cartwright, 2006). The above result allows minimum cost algorithms like Ngila to be used to calculate maximum likelihood alignments. Note that such conversion will also change the cost of free end gaps of length k to $xk/2y$.

Applying the above conversion to a simple statistical model developed in Cartwright (2006) suggests that the following costs are biologically appropriate: $\beta_{\text{match}} = 0$, $\beta_{\text{mismatch}} = 1$, and

$$w_k = \frac{\ln \zeta(z) - \ln\left(e^{\lambda\theta} - 1\right) + 0.5\ln\left(1 + 3e^{-4\theta/3}\right)k + z\ln k}{\ln(1 + 3e^{-4\theta/3}) - \ln(1 - e^{-4\theta/3})} \qquad (1)$$

Here, θ is the expected number of substitutions per residue between a pair of sequences, λ is the expected number of indels per substitution and z is the slope parameter of the power law of indel lengths. We are currently developing an expectation-maximization (EM) algorithm to robustly estimate these parameters from comparative genomic sequences.

4 PERFORMANCE

Using human ADH1C intron 7 (NC_000004.10: 100488859–100489717 complement) and mouse ADH1 intron 7 (NC_000069.4: 1138226273–138227189), an EM algorithm summed across all possible alignments and estimated $\theta = 0.47822$, $\lambda = 0.213542$ and $z = 1.80675$ (Cartwright, unpublished data). Using these values, the sequence simulation program, Dawg (Cartwright, 2005), created a dataset of 1000 pairwise alignments. The sequences in these alignments were truncated to the lengths shown in Table 1. These were then aligned using a log-affine gap cost calculated via Equation (1) and an affine cost derived from the log-affine gap cost by least squares. Speed and alignment quality were calculated and are summarized in Table 1. As expected, the time complexity of Ngila is roughly $O(mn)$, and log-affine costs are slower but produce better alignments.

Table 1. Comparison of log-affine and affine gap costs

Lengths	Log-affine			Affine		
	Speed	Scaled	Quality	Speed	Scaled	Quality
1600 × 1600	570	100	72%	211	37	58%
1600 × 800	285	50	78%	110	19	68%
800 × 800	147	26	69%	58	10	57%
800 × 400	76	13	76%	32	5.6	67%
400 × 400	41	7.1	64%	18	3.2	56%

Sets of 1000 simulated sequence pairs were aligned holistically using a match cost of 0, a mismatch cost of 1 and a gap cost of either $w_k = 1.67884 + 0.279091\,\text{k} + 1.06161\,\ln\,\text{k}$ or $w_k = 6.86637 + 0.281073\,k$. Speed is the number of seconds Ngila spent aligning the sets, averaged across four runs. Quality is calculated by the average identity of Ngila's alignments to the simulated alignments (Cartwright, 2006). Tests were done on a 3GHZ Xeon workstation.

5 CONCLUSION

Ngila is an application that will calculate the best alignment between a pair of sequences using log-affine gap costs, which are very accurate and biologically realistic. It has the potential to advance research into alignment algorithms and provide more biologically accurate alignments.

ACKNOWLEDGEMENTS

This research was funded by an NSF Predoctoral Fellowship and N.I.H. grant GM070806. The author would like to thank Ben Redelings, Jeff Thorne and two anonymous reviewers for their helpful suggestions.

Conflict of Interest: none declared.

REFERENCES

Benner,S.A. *et al.* (1993) Empirical and structural models for insertions and deletions in the divergent evolution of proteins. *J. Mol. Biol.*, **229**, 1065–1082.

Cartwright,R.A. (2005) DNA assembly with gaps (Dawg): Simulating sequence evolution. *Bioinformatics*, **22** (Suppl. 3), iii31–iii38.

Cartwright,R.A. (2006) Logarithmic gap costs decrease alignment accuracy. *BMC Bioinformatics*, **7**, 527.

Chang,M.S.S. and Benner.S.A. (2004) Empirical analysis of protein insertions and deletions determining parameters for the correct placement of gaps in protein sequence alignments. *J. Mol. Biol.*, **341**, 617–631.

Gonnet,G.H. *et al.* (1992) Exhaustive matching of the entire protein sequence database. *Science*, **256**, 1443–1445.

Gotoh,O. (1982) An improved algorithm for matching biological sequences. *J. Mol. Biol.*, **162**, 705–708.

Gu,X. and Li,W.H. (1995) The size distribution of insertions and deletions in human and rodent pseudogenes suggests the logarithmic gap penalty for sequence alignment. *J. Mol. Evol.*, **40**, 464–473.

Hirschberg,D.S. (1975) A linear space algorithm for computing maximal common subsequences. *Commun. ACM*, **18**, 341–343.

Holmes,I. and Durbin,R. (1998) Dynamic programming alignment accuracy. *J. Comput. Biol.*, **5**, 493–504.

Miller,W. and Myers,E.W. (1988) Sequence comparison with concave weighting functions. *Bull. Math. Biol.*, **50**, 97–120.

Mott,R. (1999) Local sequence alignments with monotonic gap penalties. *Bioinformatics*, **15** 455–462.

Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, **48**, 443–453.

Sellers,P.H. (1974) On the theory and computation of evolutionary distances. *SIAM J. Appl. Math.*, **26**, 787–793.

Smith,T.F. *et al.* (1981) Comparative biosequence metrics. *J. Mol. Evol.*, **18**, 38–46.

Thorne,J.L. *et al.* (1991) An evolutionary model for maximum-likelihood alignment of DNA-sequences. *J. Mol. Evol.*, **33**, 114–124.

Waterman,M.S. (1984) Efficient sequence alignment algorithms. *J. Theor. Biol.*, **108**, 333–337.

Waterman,M.S. *et al.* (1976) Some biological sequence metrics. *Adv. Math.*, **20**, 367–387.

Zhang,Z. and Gerstein,M. (2003) Patterns of nucleotide substitution, insertion and deletion in the human genome inferred from pseudogenes. *Nucleic Acids Res.*, **31**, 5338–5348.