BIOINFORMATICS ORIGINAL PAPER

Vol. 23 no. 6 2007, pages 673–679
doi:10.1093/bioinformatics/btm009

Genome analysis

Identifying bacterial genes and endosymbiont DNA with Glimmer

Arthur L. Delcher[1,*], Kirsten A. Bratke[2], Edwin C. Powers[3] and Steven L. Salzberg[1]

[1]Center for Bioinformatics & Computational Biology, University of Maryland, College Park, MD 20742, USA
[2]Smurfit Institute of Genetics, University of Dublin, Trinity College Dublin, Dublin 2, Ireland
[3]Department of Chemical & Biomolecular Engineering, Johns Hopkins University, Baltimore, MD 21218, USA

Received on August 3, 2006; revised on December 15, 2006; accepted on January 14, 2007

Advance Access publication January 19, 2007

Associate Editor: Alfonso Valencia

ABSTRACT

Motivation: The Glimmer gene-finding software has been successfully used for finding genes in bacteria, archæa and viruses representing hundreds of species. We describe several major changes to the Glimmer system, including improved methods for identifying both coding regions and start codons. We also describe a new module of Glimmer that can distinguish host and endosymbiont DNA. This module was developed in response to the discovery that eukaryotic genome sequencing projects sometimes inadvertently capture the DNA of intracellular bacteria living in the host.

Results: The new methods dramatically reduce the rate of false-positive predictions, while maintaining Glimmer's 99% sensitivity rate at detecting genes in most species, and they find substantially more correct start sites, as measured by comparisons to known and well-curated genes. We show that our interpolated Markov model (IMM) DNA discriminator correctly separated 99% of the sequences in a recent genome project that produced a mixture of sequences from the bacterium *Prochloron didemni* and its sea squirt host, *Lissoclinum patella*.

Availability: Glimmer is OSI Certified Open Source and available at http://cbcb.umd.edu/software/glimmer

Contact: adelcher@umiacs.umd.edu

1 INTRODUCTION

The genomes of bacteria, archæa and viruses are very gene-dense, with protein-coding regions typically comprising 90% or more of the DNA sequence. As a consequence, the accuracy of prokaryotic gene-finding programs depends primarily on identifying which of the six possible reading frames contains the true gene (Besemer and Borodovsky, 1999; Borodovsky and McIninch, 1993; Guo *et al*., 2003; Ouyang *et al*., 2004). The accuracy of gene finding systems in these species is very high as compared to eukaryotic gene finders; previous versions of the Glimmer system had a sensitivity of 99% or higher (Delcher *et al*., 1999; Salzberg *et al*., 1998).

*To whom correspondence should be addressed.

However, there is still some room for improvement. First, the measurement of sensitivity relies on comparisons to well-annotated bacterial genomes, where the best we can do is to count how many 'known' genes are found by a gene finder. Genes are considered known if they have clear homology, as measured by amino-acid similarity, to genes in other species. This similarity often breaks down near the 5′ end of the transcript, which also tends to be the region where gene finders disagree on the precise position of the start codon. Thus, one area where gene finders might still improve is in prediction of start sites, as has been pointed out in previous studies (Besemer *et al*., 2001).

A second issue is false positives, i.e. gene predictions that do not correspond to genuine protein-coding genes. Because bacteria are so gene-dense, it is very difficult to say with confidence that any gene predicted to lie in an otherwise intergenic region is false. Fortunately, the growing number of sequenced genomes from closely related species does provide some help with this question, and indeed some microbial gene-finding systems rely on database searches to identify genes (Badger and Olsen, 1999; Frishman *et al*., 1998; Larsen and Krogh, 2003; Nielsen and Krogh, 2005). If a predicted protein is not conserved between closely related species, then evolutionary arguments can be made that the prediction is false. A greater source of false positives in earlier releases of Glimmer, though, came from predicting too many overlapping genes. Because truly overlapping genes are quite rare in bacterial genomes, the system should generally avoid such predictions. Here too, homology to other species can resolve the question of which gene is correct. Our challenge was to reduce the false positive rate of Glimmer without sacrificing its high sensitivity (true positive) rate.

The new Glimmer, release 3.0, achieves a dramatically lower false-positive rate, predicts many more start sites correctly, and maintains its high true positive rate. It does this through a new algorithm for scanning coding regions, a new start site detection module, and an overall architecture that for the first time integrates all gene predictions across an entire genome. In addition, a new automated training program produces substantially improved training sets, particularly on genomes with high GC-content.

We also introduce a new use for the interpolated Markov model (IMM) that is at the core of Glimmer. Recent large-scale

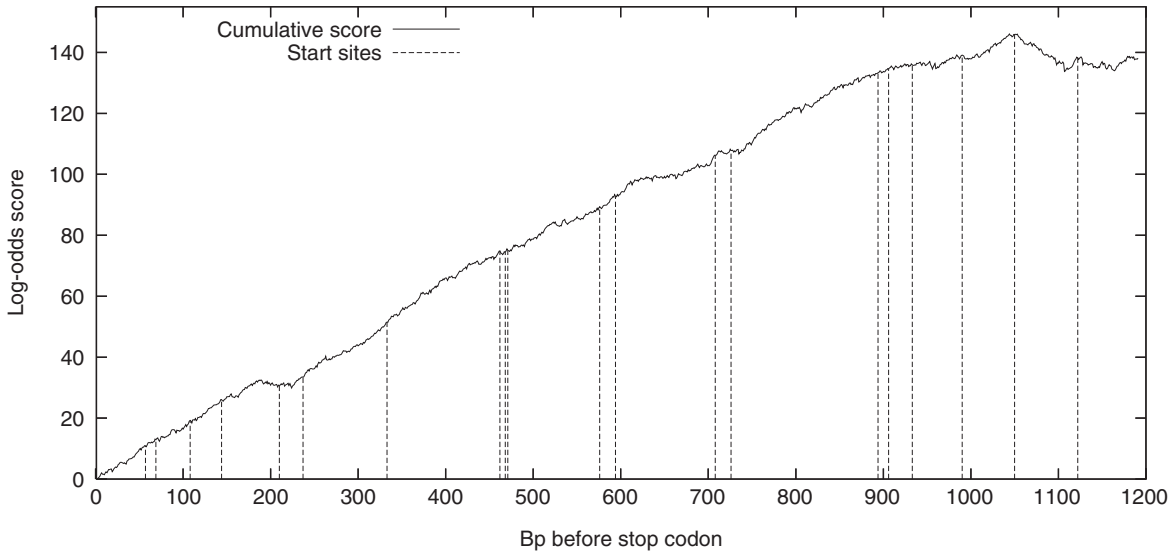

Fig. 1. Scoring an open reading frame from the stop codon backwards. The stop codon is at position 0 on the X-axis and the cumulative log-odds score is plotted as the solid line. Positions of possible start codons are indicated by vertical dashed lines. This ORF contains the fructose bis-P aldolase gene in *Escherichia coli* (EG14062) and the current Ecogene verified start site is at position 1050, near the peak score. This position is an update of the originally annotated start at position 1122.

sequencing projects of eukaryotic species have inadvertently captured the genomes of bacterial endosymbionts as a side effect of the overall project (Salzberg *et al*., 2005). When a eukaryotic species has an intracellular endosymbiont, as is true for many invertebrates including fruit flies, mosquitoes and nematodes, then a whole-genome shotgun sequencing project cannot avoid capturing some of the symbiont DNA. Consequently there is a need to identify and separate the DNA from the host and the symbiont, in order to assemble the two genomes separately (and correctly). Besides these eukaryotic genome projects, a growing number of bacterial sequencing projects are targeting endosymbionts that can only be grown inside their hosts, including *Wolbachia pipientis* (Wu *et al*., 2004) and *Prochloron didemni* (J. Ravel, personal communication). In these projects, despite investigators' best efforts to isolate bacterial DNA, a considerable amount of eukaryotic host DNA remained in the sample and needed to be removed. Motivated by these problems, we developed a new algorithm in which the IMM within Glimmer is trained separately on host and endosymbiont DNA, and then turned into a classifier to separate the raw sequences. We report here on this new module and its successful use in two recent genome projects.

The Glimmer 3.0 package is distributed as OSI Certified Open Source software and is freely available at *http://cbcb.umd.edu/software/glimmer*

2 METHODS AND RESULTS

2.1 Reverse scoring

The IMM scoring algorithm in Glimmer computes the log-likelihood that a given interval on a DNA sequence was generated by a model of coding versus noncoding DNA. This model represents the probability of a nucleotide given a subset of positions in a window (called the context) adjacent to the nucleotide—for details, see (Delcher *et al*., 1999; Salzberg *et al*., 1998). Glimmer 3.0 takes advantage of the flexibility of this algorithm by scoring all open reading frames (ORFs) in reverse, from the stop codon back toward the start codon, with the probability of each base conditioned on a context window on its 3′ side and the score of the ORF being the log-likelihood sum of the bases contained in the ORF. The score is computed incrementally as a cumulative sum at every codon position in a given ORF. In many cases, these scores show a marked peak in value, and furthermore this peak typically occurs near the correct start site (see Fig. 1 for an example). The advantage of scanning ORFs in reverse is that for nucleotides near the start site, the context window of the IMM is contained within the coding portion of the gene, which is the type of data on which it was trained. This results in a more precise cumulative score at nucleotides very close to the start site, compared to a context window on the opposite side which would intersect a non-coding region.

As the figure shows, the cumulative IMM score steadily increases as we move away from the stop codon at the left until it reaches a peak and then begins to decline. The decrease occurs because the bases upstream of the gene start codon are non-coding and produce negative IMM log-odds scores. The figure also shows, as vertical dashed lines, the positions of all possible start codons in the ORF. We hypothesized that if we used the highest-scoring start codon in these plots, then Glimmer would find a higher percentage of true start sites. This is borne out in our experiments, described subsequently.

An important difference between this algorithm and earlier versions of Glimmer is that, unlike those versions, which had a strong bias in favor of longer ORFs, this algorithm chooses start sites based on their relative scores.

2.2 Ribosome binding sites

In previous versions of Glimmer, the ribosome binding site (RBS) was essentially ignored, even though it provides a strong signal for the position of the true start site. We addressed this problem with a standalone program, RBSfinder, that can be run as a post-processor on the results of Glimmer's analysis. RBSfinder is quite effective at finding

ribosome binding sites and adjusting Glimmer's predictions (Suzek *et al*., 2001), but we nonetheless felt that a better design would integrate RBS evidence directly into the gene-finding algorithm. Glimmer3 now contains this long-awaited integration.

After experimenting with several alternative algorithms, we found that the ELPH software (*http://cbcb.umd.edu/software/ELPH*) was highly effective at identifying the likely RBS in most bacterial genomes. Input to ELPH is a specified motif length and any set of sequences, in which it identifies likely shared motifs using a Gibbs sampling algorithm. ELPH produces a position weight matrix (PWM) that Glimmer3 then uses to score any potential RBS. If a substantial set of training genes is available, the regions upstream from their starts can be given to ELPH to produce a PWM. Otherwise, Glimmer3 can bootstrap itself by first running without a PWM, generating a set of gene predictions, and then extracting regions upstream of those predictions as input to ELPH. Glimmer3 can then be re-run with the PWM to produce a more accurate set of start-site predictions. The entire process can be iterated as desired until a consistent PWM and set of gene predictions result. This strategy of using a Gibbs-sampler to find RBS motifs in an iterative fashion was introduced in the GeneMarkS gene-finding system (Besemer *et al*., 2001).

2.3 Reduced overlapping predictions

Both Glimmer2 and Glimmer3 start by identifying open reading frames (ORFs) with sufficiently high IMM scores to be processed further. In many cases, these ORFs overlap by more than the (user-specified) maximum allowed distance, indicating that only one of them is a true gene. Glimmer2 uses a series of rules based on ORF lengths, ORF scores and the IMM score of the overlapping region to attempt to resolve these overlap cases. When the rules do not produce a clear conclusion, however, Glimmer2 outputs both ORFs with an annotation indicating the overlap. As a result, Glimmer2 can have a high false-positive rate, particularly for high-GC genomes, which have large numbers of overlapping ORFs.

In contrast, Glimmer3 begins by assigning a score to each valid start position within an ORF. This score is the sum of RBS score plus the IMM coding potential score plus a score for the start codon (determined by the relative frequency of each possible start codon in the same training set used to determine the RBS). Each possible start codon is linked to the stop codon that terminates its ORF.

A global dynamic-programming algorithm is then used to select the set of ORFs and start sites with maximum total score across the entire input sequence, subject to the constraint that no overlaps greater than a specified maximum are allowed. Specifically, the set of potential start sites and stop positions is scanned in sorted order by location on the input sequence. At each start or stop feature f, the score of the maximum-scoring set of genes up to and including f is computed as the maximum compatible prior score in any of the six reading frames plus the score of f. Because overlaps are allowed, the value for a feature f may, in fact, change as the result of a feature encountered after f. To accommodate this case, our algorithm backtracks to update scores within the maximum allowed overlap distance and adjusts the scores to avoid double counting the score of the overlap region. Because the maximum overlap distance is typically small compared to the average gene length, the additional cost is usually insignificant.

In many respects this algorithm functions like the hidden Markov models (HMMs) used in other gene-finding programs such as GeneMark.hmm (Lukashin and Borodovsky, 1998) and EasyGene (Larsen and Krogh, 2003). The principal differences are that small overlaps between genes are allowed (without resort to a complicated set of overlap states in an HMM) and that potential coding regions are pre-scored by the IMM in the stop-to-start direction so that the scanning direction of the algorithm effectively alternates on different segments of the sequence. The result is that the final set of Glimmer3 predictions contains no overlaps greater than the specified maximum, and the total number of Glimmer3 predictions is almost always less than the corresponding number of Glimmer2 predictions.

2.4 Improved training with `long-orfs`

One of Glimmer's strengths has always been the ease with which any user can automatically train it on a new genome. The `long-orfs` program in the Glimmer system is used to create a training set of genes from a genome by selecting ORFs above a threshold length that do not overlap other ORFs above that threshold length. The threshold length is computed by the program to be the value that maximizes the number of non-overlapping ORFs produced, thus maximizing the amount of data in the training set. For most genomes this approach is quite effective, typically producing a training set containing nearly half of all genes with relatively few ORFs that are not genes. In the case of high-GC genomes (>60% GC), however, the scarcity of stop codons results in an abundance of long ORFs that are not genes. For such genomes, the version of `long-orfs` in Glimmer2 produces very small output sets, with many incorrect genes.

To overcome this problem, the `long-orfs` program in Glimmer3 incorporates a new routine to filter the initial set of ORFs based on amino-acid composition. Here we wish to take advantage of the fact that the genes in widely disparate bacterial genomes tend to use a common, universal amino acid distribution (Luscombe *et al*., 2001; Pascal *et al*., 2005). By comparing the ORFs found by `long-orfs` to a universal distribution, we should be able to eliminate many ORFs that are highly unlikely to be protein-coding genes. Specifically, we compute the distribution of each ORF's amino acids and compare it to both a positive model derived from a large sample of microbial genomes, and to a negative model created from alternative reading frames of those genes. We then compute the ratio of the distance of the candidate ORF's distribution to the positive and negative models, and only those ORFs whose ratio is below a user-specified threshold are passed on to the length- and overlap-calculation stage. These distance calculations are actually computed using the entropies of amino-acid distributions as described in (Ouyang *et al*., 2004).

Table 1 compares the output of the Glimmer2 and Glimmer3 versions of `long-orfs` to the set of all annotated genes for a sample of 13 bacterial and archæal genomes obtained from NCBI (Wheeler *et al*., 2006). Note how for the high-GC (67%) *Ralstonia solanacearum* genome, Glimmer2's `long-orfs` outputs only 288 ORFs, of which a mere 55% match annotated genes. In contrast, Glimmer3's `long-orfs` identifies 1175 ORFs, 96% of which match annotated genes.

2.5 Gene prediction accuracy

In order to test the effect of the improvements in Glimmer3, we compared it to several different sets of data, shown in the tables. First, we compared its predictions on a sample of complete genomes to the 'known' genes from those genomes. We used NCBI annotation to determine when a gene was known, by simply removing all genes annotated as 'hypothetical' from the set of known genes. Genes assigned a function are in most cases closely homologous to genes from other genomes, which provides independent evolutionary evidence that these genes are real. We are aware that this method has its shortcomings, but no other method yields nearly as many genes for testing. This method also does not guarantee that the start codon is correctly predicted, because homology does not need to extend for the full length of a predicted protein in order to be considered adequate evidence for assigning function.

Another important test of Glimmer 3.0 was its comparison in relation to Glimmer 2.13, and we, therefore, ran both algorithms on the same set of genomes. In Table 2 we show the accuracy results of both Glimmer2 and Glimmer3 on our benchmark genome sample. Both algorithms

Table 1. Glimmer3 and Glimmer2 `long-orfs` output comparison

Genome			Glimmer3 `long-orfs`			Glimmer2 `long-orfs`			G3 versus G2		
Organism	GC%	# Genes	3′ Matches		Extra	3′ Matches		Extra	3′ Matches		Extra
A.fulgidus	49	2398	1083	45%	26	706	29%	18	+377	+16%	+8
B.anthracis	35	5308	3494	66%	194	2934	55%	160	+560	+11%	+34
B.subtilis	44	4095	2647	65%	21	2062	50%	21	+585	+14%	0
C.tepidum	57	2252	943	42%	37	438	19%	30	+505	+22%	+7
C.perfringens	29	2660	2111	79%	16	1885	71%	16	+226	+8%	0
E.coli	51	4231	2754	65%	39	1815	43%	17	+939	+22%	+22
G.sulfurreducens	61	3438	1432	42%	59	553	16%	61	+879	+26%	−2
H.pylori	39	1556	1141	73%	20	831	53%	10	+310	+20%	+10
P.fluorescens	63	6134	2873	47%	71	579	9%	129	+2294	+37%	−58
R.solanacearum	67	3435	1133	33%	42	157	5%	131	+976	+28%	−89
S.epidermidis	32	2487	1797	72%	40	1480	60%	27	+317	+13%	+13
T.pallidum	53	1034	507	49%	7	379	37%	6	+128	+12%	+1
U.parvum	26	614	400	65%	0	338	55%	9	+62	+10%	−9
Averages:				57%			39%		+628	+18%	−5

For each genome, '#Genes' counts all genes longer than 90bp in the NCBI annotation after removing genes with frame shifts and internal stop codons. A prediction is a match iff it has the same reading frame and stop codon as a gene. Extra predictions are those that are not matches. The 'G3 versus G2' column is the Glimmer3 value minus the Glimmer2 value.

Table 2. Glimmer3 prediction accuracy when trained on confirmed genes

Genome			Glimmer3 Predictions					versus Glimmer2.13		
Organism	GC%	# Genes	3′ Matches		5′ & 3′ Matches		Extra	3′ Match	5′ & 3′	Extra
A.fulgidus	49	1165	1162	99.7%	841	72.2%	1308	−2	−67	−59
B.anthracis	35	3132	3119	99.6%	2717	86.7%	2345	+6	+726	−77
B.subtilis	44	1576	1559	98.9%	1379	87.5%	2886	+11	+413	−539
C.tepidum	57	1292	1284	99.4%	867	67.1%	778	+2	−33	−190
C.perfringens	29	1504	1501	99.8%	1360	90.4%	1177	−1	+244	−28
E.coli	51	3603	3525	97.8%	3014	83.7%	942	+16	+693	−632
G.sulfurreducens	61	2351	2320	98.7%	1883	80.1%	1107	+15	+541	−380
H.pylori	39	915	908	99.2%	785	85.8%	774	+1	+46	−94
P.fluorescens	63	4535	4484	98.9%	3412	75.2%	1896	+14	+731	−704
R.solanacearum	67	2512	2468	98.2%	1922	76.5%	1091	+72	+646	−326
S.epidermidis	32	1650	1646	99.8%	1496	90.7%	767	+3	+338	−66
T.pallidum	53	575	569	99.0%	397	69.0%	568	+3	+55	−296
U.parvum	26	327	325	99.4%	292	89.3%	297	0	+19	−17
Averages:				99.1%		81.1%		+11	+335	−262

For each genome, '#Genes' counts genes in the NCBI annotation that are at least 90bp long, do not have frame shifts or internal stop codons, and whose function description does not contain the string 'hypothetical'. Both Glimmer3 and Glimmer2 were run with the same options and training/test sets in an 8-way cross-validation experiment on this set of genes. A prediction is a 3′ match iff it has the same reading frame and stop codon as a gene. 5′ & 3′ matches are predictions with the same start and stop codon as the annotation. Extra predictions are those that are not matches. The 'versus Glimmer2' column is the Glimmer3 value minus the corresponding Glimmer2 value.

were trained and tested on the same data set of non-hypothetical genes in an eight-way cross-validation experiment. (For each genome, the genes were divided into eight approximately equal-size subsets. One subset was held out for testing and Glimmer was trained on the remaining subsets. This was repeated eight times so that each gene was part of one test set.) As shown in the table, Glimmer3 nearly always achieves equal or higher sensitivity than Glimmer2, but with far fewer additional predictions, indicating much greater specificity. Glimmer3 also has far greater agreement between its start codon predictions and those in the benchmark genomes: in 11 of the 13 genomes, it has greater agreement than Glimmer2. For example, in *Bacillus anthracis* Glimmer3 predicts 726 start sites that agree with the NCBI annotation but disagree with Glimmer2. Note too that we cannot guarantee that the start sites are correctly annotated for any of these genomes, but without further laboratory evidence these are the best data available. Table 3 compares the results of running both Glimmer2 and Glimmer3 using the output of their respective versions of the `long-orfs` program to produce a training set. As before, the predictions are compared to non-hypothetical genes in the annotation. In this case, Glimmer2's performance is substantially worse because of errors in its

Table 3. Glimmer3 prediction accuracy with `long-orfs` training

Genome			Glimmer3 Predictions					versus Glimmer2.13		
Organism	GC%	# Genes	3′ Matches		5′ & 3′ Matches		Extra	3′ Match	5′ & 3′	Extra
A.fulgidus	49	1165	1161	99.7%	873	74.9%	1332	−2	−34	−64
B.anthracis	35	3132	3125	99.8%	2751	87.8%	2419	−1	+752	−144
B.subtilis	44	1576	1562	99.1%	1391	88.3%	3020	+3	+421	−724
C.tepidum	57	1292	1289	99.8%	934	72.3%	835	+3	+26	−400
C.perfringens	29	1504	1501	99.8%	1383	92.0%	1192	−1	+267	−20
E.coli	51	3603	3534	98.1%	3112	86.4%	1002	+11	+784	−843
G.sulfurreducens	61	2351	2337	99.4%	1933	82.2%	1165	+7	+575	−734
H.pylori	39	915	910	99.5%	795	86.9%	788	+2	+57	−103
P.fluorescens	63	4535	4510	99.4%	3598	79.3%	1953	+35	+895	−2359
R.solanacearum	67	2512	2485	98.9%	2028	80.7%	1183	+341	+1044	−2184
S.epidermidis	32	1650	1646	99.8%	1514	91.8%	791	+8	+358	−32
T.pallidum	53	575	567	98.6%	391	68.0%	567	−2	+50	−281
U.parvum	26	327	324	99.1%	295	90.2%	297	−1	+21	−11
Averages:				99.3%		83.1%		+31	+401	−608

Genomes and columns are as in the preceding table. Glimmer3 was run by using the output of its `long-orfs` program to train an IMM. The output of an initial run of Glimmer3 was used to set start codon frequencies and to find a ribosome-binding-site motif. A second run of Glimmer3 using those values generated the above predictions. Glimmer2 was trained on the output of its version of the `long-orfs` program.

Table 4. Glimmer3 prediction accuracy compared to other gene-finding systems

Genome		versus GeneMark.hmm			versus EasyGene 1.2			versus GeneMarkS		
Organism	# Genes	3′ Match	5′ & 3′	Extra	3′ Match	5′ & 3′	Extra	3′ Match	5′ & 3′	Extra
A.fulgidus	1165	+4	−20	−86	+5	−25	+119	0	+2	−71
B.anthracis	3132	−2	-48	−134	+13	-63	+175	+1	+412	−142
B.subtilis	1576	+2	+280	+87	+15	−10	+536	−5	−39	+193
C.tepidum	1292	+1	+21	+19	+10	+9	+182	+1	−14	+29
C.perfringens	1504	−2	+177	−120	−2	−8	−21	−3	−14	−139
E.coli	3603	−25	+18	+188	+60	+44	+407	−25	−29	+190
G.sulfurreducens	2351	+13	+215	+34	+5	−1	+60	+14	+41	+66
H.pylori	915	−1	−3	−55	+4	−6	+148	−1	−8	−41
P.fluorescens	4535	+17	+288	+59	NA	NA	NA	+17	+479	+46
R.solanacearum	2512	+7	+183	+225	+11	+48	+193	−3	+160	+190
S.epidermidis	1650	+3	−32	−40	NA	NA	NA	+6	+204	−64
T.pallidum	575	+2	−8	+94	+8	−8	+176	−2	−18	+90
Averages:		+2	+89	+23	+13	−2	+198	+2	+98	+29

Glimmer3 predictions are as in the preceding table and each entry is the Glimmer3 value minus the corresponding value for the other gene-finder. GeneMark.hmm results were taken from the GeneMarkHMM files downloaded from NCBI. EasyGene 1.2 results were downloaded from *http://servers.binf.ku.dk/cgi-bin/easygene.search* GeneMarkS results were obtained from the server at *http://exon.gatech.edu/GeneMark/genemarks.cgi* None of these systems had results for *Ureaplasma parvum*, which uses a non-standard translation code. NA entries indicate strains that were not available for download.

`long-orfs` output. Glimmer3's performance, however, is substantially the same as when it is trained with annotated data, indicating that it is likely to do well even in the absence of a pre-computed set of 'known' genes for training.

Table 4 compares Glimmer3 predictions to those of three other gene-finders, GeneMark.hmm, EasyGene (Larsen and Krogh, 2003) and GeneMarkS (Besemer *et al.*, 2001), and shows that Glimmer obtains comparable results. These other systems all run through web servers or by downloading precomputed predictions, and EasyGene uses homology-search results to help determine its parameters. In contrast, Glimmer runs locally and offers many options for choosing parameters and training sets, and can be run on collections of contigs from unfinished assemblies.

Further evidence of Glimmer3's improved accuracy is provided by comparing its predictions to the results of recent laboratory experiments to identify unannotated genes. One such experiment was conducted on the hyperthermophilic archæon *Pyrococcus furiosos* by Poole *et al.* (Poole *et al.*, 2005), who used microarray expression evidence and recombinant protein tests to examine 127 ORFs not in the NCBI annotation. Of the 17 proteins that the Poole group were able to confirm, Glimmer3 predicted 16, of which 14 also agreed on the start sites.

Table 5. Comparison of start-site prediction accuracy

Test Set	Number of Genes	Description	MED-Start on Glimmer2 Orfs		MED 2.0		Glimmer3	
			3′ Matches	5′ and 3′ Matches	3′ Matches	5′ and 3′ Matches	3′ Matches	5′ and 3′ Matches
EcoGene2006	878	EcoGene proteins (EcoData070306) annotated as "Verified" but not annotated as "EXCEP" or "MUTANT"					99.5%	92.9%
EcoGene2004	854	Ecogene proteins used in (Zhu *et al.*, 2004)	99.3%	92.0%	99.1%	92.0%	99.5%	92.7%
Link	195	Subset of EcoGene with single-amino-acid or no leader sequence (Link *et al.*, 1997)	100.0%	95.4%	99.0%	93.3%	100.0%	95.8%
Bsub58	58	*B.subtilis* genes confirmed by comparison to *B.halodurans*	98.3%	94.8%	100.0%	96.6%	98.3%	94.8%

The last three datasets are the same sets used by Zhu *et al.* to assess the accuracy of MED-Start (Zhu *et al.*, 2004), from which the first MED values are taken. The second come from the MED 2.0 web site, *http://ctb.pku.edu.cn/main/SheGroup/MED2.htm*

2.6 Start-site prediction accuracy

Assessing the accuracy of start-site predictions is very difficult due to the scarcity of reliable data about start sites. For *E.coli*, a substantial number of proteins have been verified through N-terminal sequencing, providing a highly accurate (although limited to just one species) set of data for measuring the accuracy of start site predictions. The curators of the EcoGene database (Rudd, 2000) have collected and annotated 878 genes (as of July 2006) from *E.coli* with confirmed start sites, and we used these to measure Glimmer3's accuracy. For comparison purposes, we also tested Glimmer3 on three of the same data sets used in (Zhu *et al.*, 2004) to assess the accuracy of the program MED-Start. All these data sets are described in Table 5.

The table shows Glimmer3 prediction results on these four data sets. On the latest EcoGene dataset we found that Glimmer3 predicted all but four genes, and matched the correct start site on 816 genes (92.9%). For the latter three data sets, we also show two sets of MED predictions: one from (Zhu *et al.*, 2004) obtained by applying MED-Start to Glimmer2 orfs and the other from the MED web site (*http://ctb.pku.edu.cn/main/SheGroup/MED2.htm*) using MED 2.0, which incorporates MED-Start within it. The results show Glimmer3 to be slightly more accurate on the *E.coli* data sets, while on the *Bacillus subtilis* data set the same as MED-Start while slightly less accurate than MED 2.0.

2.7 Separating sequences from different genomes

Although we designed Glimmer's IMM to model the 3-periodic structure of protein coding sequences, it also can be employed for more general sequence modeling. *P.didemni* is a photosynthetic microbe that lives as an endosymbiont in its host organism, the sea squirt *L.patella*. Because *P.didemni* can only be cultured in *L. patella* cells, it was not surprising that in the whole-genome shotgun sequencing project for *P.didemni*, a large number of sequences were from *L.patella*. Such a mixture of reads from two genomes, at different coverage densities, causes problems for genome assembly software, which typically assumes that its input derives from a single genome that was sampled at a uniform rate. Because no reference sequences were available (which would have allowed us to separate the sequences based on homology), we used the Glimmer IMM to classify the two types of sequences.

We began with an initial assembly of all 82 337 shotgun reads. Because the genome size of the bacterium (approximately 5 million base pairs) is much smaller than that of its eukaryote host (over 160 Mbp), the depth of coverage of the bacterium was much greater. Consequently any large scaffolds in the assembly would, with near certainty, consist of *P.didemni* sequences. Conversely, reads that failed to align with any other reads (singletons) would disproportionately be from the larger genome. Accordingly, we created training sets by classifying reads from assembly scaffolds at least 10Kbp long as being from *P.didemni* (36 920 reads), and reads where both the read and its clone-insert mate were singletons as being from *L.patella* (21 276 reads). This left 24 141 reads unclassified.

We created non-periodic Glimmer IMMs from these two training sets and classified sequences based on which of the two models gave a higher score. In a 5-way cross-validation test using the initial classification sets, the models achieved 98.9% accuracy on *P.didemni* reads and 99.9% accuracy on *L.patella* reads. The models classified 22% and 78% of the 24 141 unclassified reads as being from *P.didemni* and *L.patella*, respectively. One way to measure prediction accuracy on this test set is by considering the predictions of mate-pair reads. Because each pair comes from a single DNA template, the classification of both reads in the pair should be the same. There were 10 500 mate pairs in our unclassified set, of which only 207 (2%) were inconsistently classified. Since each inconsistent pair has one correct and one incorrect classification, this indicates 99% accuracy.

Besides the obvious benefit of removing the host sea squirt sequence from the assembly result, separating the two types of reads produced improvements in the quality of the shotgun assembly. The assembly of *P.didemni* using the mix of all reads produced 65 scaffolds 20 Kbp or longer, with total length of 5.74 Mbp. The assembly using just reads classified as *P.didemni* yielded 58 scaffolds 20 Kbp or longer, with total length of 5.84 Mbp. Both assemblies were run with the same parameter settings using the Celera Assembler program (Myers *et al.*, 2000).

3 CONCLUSION

The latest release of the Glimmer gene-finding system is significantly improved compared to its predecessor, most notably with respect to specificity and accuracy in predicting translation initiation sites. A major difficulty in developing software that

can make highly accurate predictions of coding starts is the lack of experimentally confirmed training and testing data. Despite this limitation, Glimmer3 start-site predictions have achieved a remarkably high success rate on the best-available dataset, the Ecogene verified genes in *E.coli*.

A notable advantage of the Glimmer3 system is that it is completely self-contained. This permits users to choose either the `long-orfs` program, or any set of genes that may wish to use, as training sets to build the IMM and find the RBS motif. This permits the system to be used on relatively short sequence fragments, like low-coverage draft genome assembly sequences. If necessary, a user can even use a closely related organism as a source of training data.

Another advantage of Glimmer3 is that it is distributed as source code that can run on any system with a C++ compiler. This enables any part of the program to be modified by the user and allows various modules to be used for purposes other than gene-finding, as we have demonstrated in separating target and host-contaminant data for the *P.didemni* shotgun sequencing project.

ACKNOWLEDGEMENTS

Thanks to Daniel H. Haft and William C. Nelson at TIGR for suggesting improvements and helping to test Glimmer, to Michaela Pertea for assistance with the ELPH program, and to Jacques Ravel for providing the *P. didemni* sequences under NSF grant EF-0412226. This work was supported in part by NIH grants R01-LM006845 and R01-LM007938 and HSARPA award W81XWH-05-2-0051 to SLS, and by NIAD contract. NIH-NIAID-DMID-04-34, HHSN266200400038C. Funding to pay the Open Access publication charges was provided by NIH grant R01-LM007938.

Conflict of Interest: none declared.

REFERENCES

Badger,J.H. and Olsen,G.J. (1999) CRITICA: coding region identification tool invoking comparative analysis. *Mol. Biol. Evol.*, **16**, 512–524.

Besemer,J. and Borodovsky,M. (1999) Heuristic approach to deriving models for gene finding. *Nucleic Acids Res.*, **27**, 3911–3920.

Besemer,J. *et al.* (2001) GeneMarkS: a self-training method for prediction of gene starts in microbial genomes. Implications for finding sequence motifs in regulatory regions. *Nucleic Acids Res.*, **29**, 2607–2618.

Borodovsky,M. and McIninch,J. (1993) Recognition of genes in DNA sequence with ambiguities. *Biosystems*, **30**, 161–171.

Delcher,A.L. *et al.* (1999) Improved microbial gene identification with GLIMMER. *Nucleic Acids Res.*, **27**, 4636–4641.

Frishman,D. *et al.* (1998) Combining diverse evidence for gene recognition in completely sequenced bacterial genomes. *Nucleic Acids Res.*, **26**, 2941–2947.

Guo,F.B. *et al.* (2003) ZCURVE: a new system for recognizing protein-coding genes in bacterial and archaeal genomes. *Nucleic Acids Res.*, **31**, 1780–1789.

Larsen,T.S. and Krogh,A. (2003) EasyGene—a prokaryotic gene finder that ranks ORFs by statistical significance. *BMC Bioinformatics*, **4**, 21.

Link,A.J. *et al.* (1997) Comparing the predicted and observed properties of proteins encoded in the genome of Escherichia coli K-12. *Electrophoresis*, **18**, 1259–1313.

Lukashin,A.V. and Borodovsky,M. (1998) GeneMark.hmm: new solutions for gene finding. *Nucleic Acids Res.*, **26**, 1107–1115.

Luscombe,N.M. *et al.* (2001) Amino acid-base interactions: a three-dimensional analysis of protein-DNA interactions at an atomic level. *Nucleic Acids Res.*, **29**, 2860–2874.

Myers,E.W. *et al.* (2000) A whole-genome assembly of Drosophila. *Science*, **287**, 2196–2204.

Nielsen,P. and Krogh,A. (2005) Large-scale prokaryotic gene prediction and comparison to genome annotation. *Bioinformatics*, **21**, 4322–4329.

Ouyang,Z. *et al.* (2004) Multivariate entropy distance method for prokaryotic gene identification. *J. Bioinform. Comput. Biol.*, **2**, 353–373.

Pascal,G. *et al.* (2005) Universal biases in protein composition of model prokaryotes. *Proteins*, **60**, 27–35.

Poole,F.L., 2nd *et al.* (2005) Defining genes in the genome of the hyperthermophilic archaeon Pyrococcus furiosus: implications for all microbial genomes. *J. Bacteriol.*, **187**, 7325–7332.

Rudd,K.E. (2000) EcoGene: a genome sequence database for Escherichia coli K-12. *Nucleic Acids Res.*, **28**, 60–64.

Salzberg,S.L. *et al.* (1998) Microbial gene identification using interpolated Markov models. *Nucleic Acids Res.*, **26**, 544–548.

Salzberg,S.L. *et al.* (2005) Serendipitous discovery of Wolbachia genomes in multiple Drosophila species. *Genome. Biol.*, **6**, R23.

Suzek,B.E. *et al.* (2001) A probabilistic method for identifying start codons in bacterial genomes. *Bioinformatics*, **17**, 1123–1130.

Wheeler,D.L. *et al.* (2006) Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res.*, **34**, D173–D180.

Wu,M. *et al.* (2004) Phylogenomics of the Reproductive Parasite Wolbachia pipientis wMel: A Streamlined Genome Overrun by Mobile Genetic Elements. *PLoS Biol.*, **2**, E69.

Zhu,H.Q. *et al.* (2004) Accuracy improvement for identifying translation initiation sites in microbial genomes. *Bioinformatics*, **20**, 3308–3317.