BIOINFORMATICS ORIGINAL PAPER

Vol. 30 no. 14 2014, pages 1958–1964
doi:10.1093/bioinformatics/btu160

Sequence analysis

Advance Access publication March 22, 2014

Improved transcript isoform discovery using ORF graphs

William H. Majoros[1,2,*], Niel Lebeck[3], Uwe Ohler[2,4,5,6] and Song Li[2]

[1]Program in Computational Biology and Bioinformatics, Duke University, Durham, NC 27708, USA, [2]Institute for Genome Sciences and Policy, Duke University, Durham, NC 27705, USA, [3]Department of Computer Science, Duke University, Durham, NC 27708, USA, [4]Department of Biostatistics and Bioinformatics, Duke University, Durham, NC 27710, USA, [5]Berlin Institute for Medical Systems Biology, Max Delbruck Center for Molecular Medicine, Berlin 13125, Germany and [6]Department of Biology, Humboldt University of Berlin, Berlin 10115, Germany

Associate Editor: Michael Brudno

ABSTRACT

Motivation: High-throughput sequencing of RNA *in vivo* facilitates many applications, not the least of which is the cataloging of variant splice isoforms of protein-coding messenger RNAs. Although many solutions have been proposed for reconstructing putative isoforms from deep sequencing data, these generally take as their substrate the collective alignment structure of RNA-seq reads and ignore the biological signals present in the actual nucleotide sequence. The majority of these solutions are graph-theoretic, relying on a *splice graph* representing the splicing patterns and exon expression levels indicated by the spliced-alignment process.

Results: We show how to augment splice graphs with additional information reflecting the biology of transcription, splicing and translation, to produce what we call an *ORF* (open reading frame) *graph*. We then show how ORF graphs can be used to produce isoform predictions with higher accuracy than current state-of-the-art approaches.

Availability and implementation: RSVP is available as C++ source code under an open-source licence: http://ohlerlab.mdc-berlin.de/software/RSVP/.

Contact: bmajoros@duke.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on September 2, 2013; revised on February 25, 2014; accepted on March 17, 2014

1 INTRODUCTION

High-throughput sequencing of RNA transcripts facilitates many applications, including gene expression analysis, single nucleotide polymorphism (SNP) detection, discovery of unannotated genes and cataloging of different splice isoforms for individual loci. As the cost of sequencing continues to decline, RNA transcript assays will become increasingly common, and efficient computational methods for extracting biological knowledge from the resulting data will become increasingly crucial.

The cataloging of splice isoform variants of protein-coding genes remains an especially important goal, as the range of possible downstream effects following the transcription of a gene is constrained by the variety of transcripts that can be produced from that locus. Various other applications using transcriptomic data rely on knowing the set of transcripts that can be produced by each gene. Even if not every transcript from a protein-coding locus results in stable protein production, identification of those transcripts that do likely produce proteins and an accurate characterization of the protein isoforms produced remain important steps in the process of understanding individual transcription units.

Unfortunately, few genomes currently have complete annotations of all splice variants of all genes. Although a full-length sequencing of whole transcripts promises to become technologically feasible in the future, at present the inference of the whole transcript structure from short RNA sequencing reads in eukaryotic organisms depends on computational methods for assembling exons into complete structures. In the case of functional protein-coding transcripts, these structures should contain a valid *open reading frame* (ORF) capable of being translated into a functional protein, i.e. beginning with a valid *start codon* (AUG) and ending with a valid *stop codon* (UGA, UAA or UAG) in the same period-3 phase.

Current methods for discovering splice isoforms from short RNA sequencing reads fall into roughly two categories: those that explicitly perform (spliced) alignment of reads to a reference genome, and those that instead perform *de novo* assembly of reads without an explicit reference. Given the increasing availability of near-complete genomic sequences, we have focused on the first category.

Reference-based methods typically use the mapping of spliced reads to induce a structure, called *splice graph*, in which each vertex represents an exon and each edge represents an intron. Paths through the graph represent individual combinations of exons comprising putative transcript isoforms. *De novo* methods also use a graph, called a *de Bruijn* graph, in which paths again represent transcripts (though vertices now represent *k*-mers). Thus, graph-theoretic algorithms based on path enumeration are common to most methods of transcript assembly from RNA sequencing data.

Although splice graphs are highly informative to the transcript assembly process because of their inclusion of information about both splicing patterns and relative exon inclusion levels, they lack any information about underlying nucleotide patterns. In particular, they provide no immediate means of determining whether a path through the graph corresponds to a transcript with a valid ORF. For protein-coding genes, this is one of the most important constraints in determining whether a transcript can encode a viable protein, as translation occurs only in ORFs.

[*]To whom correspondence should be addressed.

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

© The Author 2014. Published by Oxford University Press. All rights reserved. For Permissions, please e-mail: journals.permissions@oup.com

In this article, we show how to augment splice graphs with additional information reflecting the biology of transcription, splicing and translation, to produce what we call an *ORF graph*. ORF graphs contain explicit *phase* information, allowing efficient enumeration of isoforms containing a valid ORF, and they permit phase-specific scoring of individual transcript elements, so that isoforms can be efficiently ordered by their likelihood under some joint sequence model. In this way, ORF graphs provide more than simply an indication of whether a putative transcript contains an ORF: via the vertex and edge scores, they provide information regarding coding potential and splice site strength, and via the phase information, they provide an efficient means of ranking exon combinations without invoking the combinatorial explosion of exhaustive enumeration.

Finally, we present an efficient software implementation, called *RSVP* (*RNA-seq Variant Prediction*), which achieves higher prediction accuracy than current state-of-the-art approaches for identifying transcript isoforms. Our implementation has the advantage that it does not depend on particular library preparation protocols (e.g. strand-specific libraries, paired-end reads), although the ORF graph formalism is flexible enough that many such considerations could be accommodated. Our implementation is made available under an open-source software license.

2 METHODS

2.1 Overview of RSVP

Our pipeline begins with the generation of an initial ORF graph annotated with scores derived from DNA sequence features. Each vertex in the graph represents a *signal* such as a splice site or start/stop codon, and each edge represents a *content interval* such as an exon, intron or intergenic region. This initial ORF graph captures the most likely ORFs for the current locus, conditional only on the DNA sequence. The size of the graph is linear in the sequence length (see below). The graph is pruned to eliminate vertices not reachable from both left and right termini; it may also (optionally) be pruned to eliminate vertices having low likelihoods.

The graph is then reweighted by incorporating RNA-seq evidence. An external spliced alignment program produces 'pileup' (read depth) information for each genomic position, as well as paired splice junctions resulting from spliced read alignments. This information is used to compute a new score for each vertex and edge in the ORF graph; these new scores are combined with the initial DNA-based scores via a simple mixture.

Finally, an *N*-best algorithm is used to extract the *N* highest-scoring paths from the graph, corresponding to *N* isoform predictions; note that isoforms may differ not only by splice pattern but also by the reading frame or the length of coding segment (CDS) (position of start codon). This algorithm traverses the graph while observing phase constraints encoded by the graph to ensure that the predicted CDS will be a valid ORF. This step takes time $O(N\,|V|)$ for number of vertices $|V|$. Isoform predictions are emitted in the GFF format for convenient downstream manipulation by other programs.

These steps are described in more detail below (see Fig. 1).

2.2 Properties of ORF graphs

An ORF graph $G=(V,E)$ consists of a set of vertices V and edges E in which each vertex and each edge has associated with it three scores, denoted by $s_{v,i}$ and $s_{e,i}$ for vertex v and edge e, respectively, $0 \le i < 3$. Each vertex has an associated type *type*(v) drawn from the set {*START_CODON*, *STOP_CODON*, *DONOR_SITE*, *ACCEPTOR_SITE*, *TRANSLATION_START*, *TRANSLATION_STOP*, *LEFT_TERMINUS*, *RIGHT_TERMINUS*}, and each edge has a type *type*(e) ∈ {*EXON*, *INTRON*, *INTERGENIC*}. Each vertex and edge also has associated with it a strand {+, −}, except for intergenic edges, which are said to be *unstranded* (all others are *stranded*).

A properly formed ORF graph obeys a number of syntax constraints. Only unstranded edges may be incident on vertices of differing strand. Edges are directed, and cycles are not permitted in the graph. Each vertex has a coordinate indicating the genomic position of its corresponding sequence element (e.g. start codon, acceptor splice site). Every ORF graph begins at left with a LEFT_TERMINUS vertex and ends at right with a RIGHT_TERMINUS vertex. We denote by $A \rightarrow B$ an edge from a vertex of type A to a vertex of type B. For example, an edge DONOR_SITE→ACCEPTOR_SITE denotes an intron, whereas START_CODON→DONOR_SITE denotes a 5′ (initial) exon (see Supplementary Material for full syntax rules). Intronic and intergenic edges are said to be *non-coding* edges; exonic edges are termed *coding* edges and denote only the coding-segment portions of exons.

To trace a valid ORF across an ORF graph, codon phase must be mapped across each succeeding edge as the edge is traversed. Thus, for a path $p=\{v_0, v_1,.., v_n\}$, each vertex in the path is assigned a phase $phase(v_i)$. START_CODON and STOP_CODON vertices are constrained to always occur in phase 0, as are both terminal vertices. For a forward-strand coding edge denoting an underlying sequence interval of length L, the phase of the left vertex v_i can be mapped across the edge via $r=(phase(v_i)+L) \bmod 3$; in a valid path, r must equal $phase(v_{i+1})$ to ensure that the ORF is read in the proper phase in the next exon. (See Supplementary Material for mapping rules for other edges.) The score of a path can be obtained by summing the scores of vertices and edges in the path; because an edge may occur in multiple paths and may have a different phase in each of those paths, an edge's contribution to a path score may differ between paths. An edge score of $-\infty$ in some phase indicates that the edge cannot occur in that phase (because of an in-frame stop codon); thus, a path with score $-\infty$ is an invalid path and does not represent a valid ORF.

ORF graphs can be maintained in a memory-efficient manner. In most genomes, the random accumulation of stop codons in non-coding sequence renders long non-transcribed ORFs increasingly unlikely, so that coding edges (both those denoting transcribed and non-transcribed reading frames) are naturally bounded in both average length and expected number. Non-coding edges can be artificially bounded in number by applying standard assumptions from the gene-finding literature (Burge, 1997; Majoros, 2007), or they can be represented implicitly via prefix-sum arrays (e.g. in linear space). In this way, we avoid a quadratic growth in the graph size as sequence lengths increase.

2.3 Extracting DNA sequence features

The gene-finding literature is rich with methods for extracting statistical features from nucleotide sequence—a process called *sensing* (Majoros, 2007). We distinguish *signal sensors* (those used for scoring fixed-length sequence elements such as start/stop codons and splice sites) from *content sensors* (for scoring variable-length elements such as exons and introns). Signal sensors can be used to score the vertices in an ORF graph, and content sensors can be used to score edges, as described below.

Common signal-sensing models include probabilistic weight matrices, with or without between-position dependencies. Signal sensors typically model not only the consensus sequence (e.g. ATG for start codons) but also informative context positions in the immediate vicinity, e.g. the Kozak sequence for eukaryotic start codons, branch points and polypyrimidine tracts of acceptor sites. For the variable-length features, content sensors are typically constructed from Markov chains. Higher-order Markov chains (with order typically 5–8) capture dependencies between nearby nucleotides and in coding regions effectively capture codon statistics. Three-periodic Markov chains explicitly model the three codon

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

phases. Higher-order Markov chains for initial exons may capture common *k*-mers associated with signal peptides. Interpolated Markov chains allow the use of variable dependency order to mitigate the effects of sample size for rarer *k*-mers. See Supplementary Material for additional details.

Content regions, as denoted by edges in the ORF graph, can also be scored as to their length. Introns and intergenic regions are typically modeled in gene-finders as having geometric length distributions, whereas exons are typically modeled via discrete histograms. Efficient evaluation of signal and content sensors and length distributions can be achieved via a left-to-right pass over a genomic sequence, as demonstrated by the numerous generalized hidden Markov model (GHMM)-based gene finders currently available (Majoros *et al*., 2005). When probabilistic scoring functions are used, log scores can simply be summed along each path to get the joint (log) likelihood of the sequence and the parse, which, for a fixed sequence, is proportional to the posterior probability of the parse (in log space).

2.4 Constructing the ORF graph

Most modern gene-finding programs, particularly those based on GHMMs and similarly structured models such as *generalized conditional random fields* (GCRFs), implicitly outline an ORF graph during their operation, even if they do not explicitly build and emit such a graph. GHMM decoding algorithms maintain phase information while applying signal and content sensors to score the genomic sequence and construct a *trellis* (a subgraph of an ORF graph) efficiently using dynamic programming techniques; at the end, they perform a traceback operation across the trellis to reconstruct the highest-scoring path (Burge, 1997; Majoros *et al*., 2005). It is thus relatively straightforward to modify a GHMM-based or GCRF-based gene finder to explicitly construct and emit an ORF graph, which can then be used for other downstream analyses, such as enumeration of possible splice isoforms.

The GHMM-based gene finder *GeneZilla* (formerly, *TIGRscan*—Majoros *et al*., 2004) builds an explicit ORF graph and emits it in the GFF format. Each vertex is assigned a score by the corresponding signal sensor, and each edge is assigned a score by the appropriate content sensor. Standard practices used in GHMM-based gene finding ensure that the resulting graph grows only linearly in the size of the reference sequence. GeneZilla is among the most space- and time-efficient GHMM-based gene finders available. It operates by maintaining a set of signal sensors that emit vertices for putative signals (e.g. start/stop codons, splice sites) encountered during a left-to-right scan over the reference sequence. Vertices are stored in type-specific priority queues and linked back to predecessor vertices from other queues while observing syntax and phase constraints. Stop codons encountered in the sequence cause the current reading frame to be terminated, resulting in a score of $-\infty$ in downstream signals in the appropriate phase; all scores are log likelihoods. See Supplementary Material for additional details.

Because the graph construction process runs in time linear in the length of the reference sequence, and because the signal queues are effectively bounded by a constant size on average, the overall construction time remains linear, and in practice is fast.

2.5 Reweighting the ORF graph

Given an ORF graph annotated with scores s_{DNA} derived from DNA features, a new graph combining information from both DNA and RNA evidence can be derived by simply reweighting the edges and vertices of the original graph via linear combination of DNA and RNA features: $s_{\text{new}} = \lambda_{\text{DNA}} s_{\text{DNA}} + \lambda_{\text{RNA}} s_{\text{RNA}}$. For RSVP, we use fixed weights of $\lambda_{\text{DNA}} = 0.99$ and $\lambda_{\text{RNA}} = 0.01$, as we found these to work well in initial runs on the training data. To simplify later traversal of the graph, we also subsume all vertex scores into incident coding edge scores, so that summing the edge scores along a path gives the complete score of the path.

RNA scores are computed as follows; note that these scores are used only to improve identification of isoform structures, as quantification of transcript levels is not a goal of RSVP. For each genomic position *i* in a locus, define d_i to be the read depth at that position, and define d_{max} to be the maximum read depth over all positions in the locus. For exon edges in the ORF graph, $s_{RNA} = \prod_i d_i/d_{max}$ over all positions *i* in the genomic interval corresponding to the edge. For intronic edges, $s_{RNA} = (n_{splice}/n_{all})^L$ is the number of spliced reads n_{splice} that are spliced exactly as the putative intron, divided by the maximum depth n_{all} of spliced reads for the locus, raised to intron length *L*. For intergenic edges, s_{RNA} is $[(1-c)(1-t)]^L$ for *c*, the per-position RNA score that would be applied if the edge had been coding, and *t*, the per-position RNA score that would be applied if the edge had been intronic. A pseudocount is added to all RNA scores. Read depths are log-transformed to mitigate the effects of sequence bias. During reweighting, we also mark edges having any RNA support (e.g. non-zero read count for exon edges, non-zero spliced read count for intron edges). The default behavior of RSVP is to delete unsupported intron edges; this step may be disabled.

2.6 Extracting the *N*-best paths

To generate *N* distinct isoform predictions for a locus, we identified the *N* highest-scoring paths spanning the ORF graph from the left terminus to the right terminus. This algorithm involves a simple modification of the standard Viterbi algorithm (Viterbi, 1967) for extracting the single best state path for an HMM. Briefly, Viterbi decoding involves a left-to-right pass, which builds a trellis in which each vertex is linked back to the optimal predecessor vertex, followed by a right-to-left pass that reconstructs the optimal path. For gene-finding HMMs and GHMMs, Viterbi additionally considers the three codon phases separately and respects phase constraints when reconstructing the optimal path (Majoros, 2007). The *N*-best algorithm generalizes this to maintain the *N* optimal predecessor *link pairs* (predecessor and predecessor's link index) in each phase at each vertex during trellis construction and then constructs *N* paths at the end by tracing backward separately from the *N* queue elements at the right terminus (see Supplementary Material for additional details).

3 RESULTS

We assessed the accuracy of RSVP by applying it to library-normalized RNA-seq data collected from *Arabidopsis thaliana* seedlings and flowers (Marquez *et al*., 2012; see Supplementary Material for comparisons on human data.). This model organism is highly suitable for comparative testing of transcript-based approaches to isoform reconstruction, because of its relatively small introns, small numbers of exons per gene and large proportion of single-exon genes. TAIR10 (Lamesch *et al*., 2012) human-curated annotations were partitioned into a training set consisting of all genes on chromosome 1, and a test set consisting of a subset of the remaining TAIR10 genes filtered to be non-redundant with the training set by ensuring that no test gene and training gene were >80% similar over 80% of their length according to BLASTN (Basic Local Alignment Search Tool / Nucleotides) (Altschul *et al*., 1997). This resulted in 7078 training genes and 17 617 test genes. Because we wished to test the ability of RSVP to identify alternative splice isoforms of known genes, we reduced the test set to 4230 genes each having multiple splice isoforms annotated in TAIR10, and we applied RSVP to a genomic interval extending from 50 bp upstream of the most 5′ annotated start codon for the gene to 50 bp downstream of the most 5′ annotated stop codon. Alignment of reads to the genome

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

was performed by Bowtie 2 (Langmead and Salzberg, 2012) and Tophat 2 (Kim *et al*., 2013), and these were processed into pileup files by SAMtools (Li *et al*., 2009). We further separated the test set into two subsets, a 'high-coverage' set for which FPKM (fragments per kilobase of sequence per million reads mapped) was at least 1 (3888 genes) and a 'low-coverage' set for which $0.1 \leq \text{FPKM} < 1$ (207 genes). We ran the *N*-best algorithm for values of *N* ranging from 1 to 20.

For comparison, we also applied three previously published tools: Cufflinks 2 (Trapnell *et al*., 2010), Scripture (Guttman *et al*., 2010) and iReckon (Mezlini *et al*., 2012). Because these programs predict transcripts without indicating the reading frame and the translation start and stop sites, we examined all possible ORFs and took the longest to be their intended prediction. Splice sites were also filtered to include only the most common splice consensuses in TAIR10 (donor sites: GT/GC/AT; acceptor sites: AG/AC). We also applied the *ab initio* gene finder GeneZilla to serve as a baseline. To assess the utility of ORF graphs for discriminating coding from non-coding loci, we applied RSVP to a set of 2707 putative lincRNAs (long interspersed non-coding loci) obtained from Liu *et al.* (2012) and additionally to a set of 807 transcribed loci of length at least 200 nt identified by Cufflinks as having $\text{FPKM} \geq 1$ but not falling within 50 nt of any annotated TAIR10 protein-coding gene. For this experiment, we disabled RNA scoring ($\lambda_{\text{RNA}} = 0$) in case of possible confounding biases in expression levels.

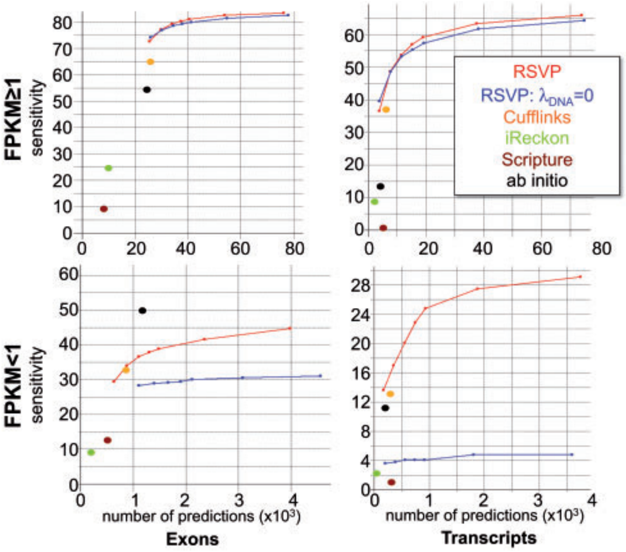

Fig. 2. Exon and transcript sensitivity on the held-out test set as a function of numbers of exons or transcripts predicted (in thousands). Left column: exons. Right column: transcripts. Top row: genes with $\text{FPKM} \geq 1$. Bottom row: genes with $\text{FPKM} < 1$. RSVP: using the full mixture model combining DNA and RNA evidence. RSVP:RNA: using only the RNA mixture component

As illustrated in Figure 2, RSVP achieves higher exon and transcript sensitivity than all three competing programs when making roughly the same numbers of predictions on annotated coding genes. Exon sensitivity is defined as the proportion of TAIR10 exons in the held-out test set that were included in the set of predicted transcripts, where an exon's begin and end coordinates must be predicted exactly correctly to be counted as a found exon. Similarly, a TAIR10 transcript is considered to be found by a predictor if one of the predicted transcripts exactly matches (for all exon coordinates, including translation start/stop codon coordinates) a TAIR10 transcript. We do not evaluate specificity, as the premise for applying tools of this type is that the set of annotations is incomplete, and therefore the false-positive rate cannot be reliably estimated based on existing annotations alone. In lieu of specificity, we plot the number of predictions (*x*-axis), which is expected to scale inversely with specificity.

Figure 2 additionally shows that when using both DNA and RNA evidence, RSVP produces higher exon and transcript sensitivity at roughly equal prediction counts, for both low-FPKM and high-FPKM genes, than when DNA scores are omitted. As expected, the effect was greater for low-FPKM genes, where the use of DNA evidence may compensate for sampling error and gaps in coverage by low RNA read counts. This underscores the importance of DNA evidence for lowly expressed genes and/or lower-quality libraries. In Figure 3, we show that increasing

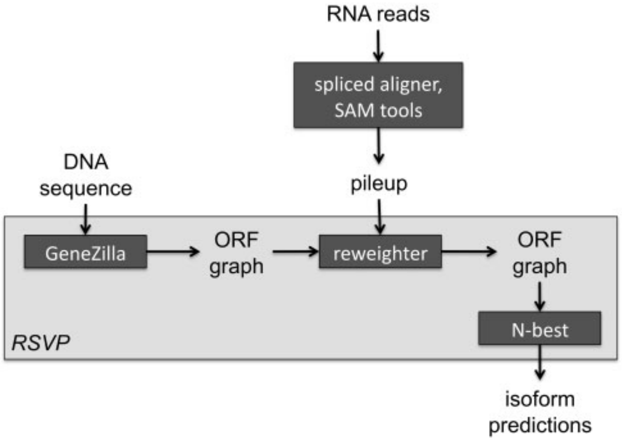

Fig. 1. The RSVP pipeline. The gene-finder GeneZilla is used to construct an initial ORF graph, which is then reweighted and further pruned based on RNA-seq data. An *N*-best algorithm extracts isoform predictions from the reweighted graph

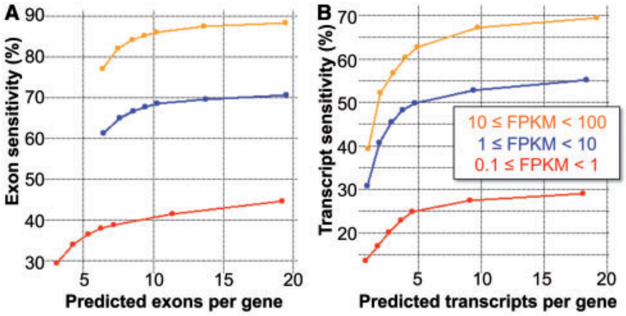

Fig. 3. Increasing FPKM increases RSVP's prediction accuracy, but with diminishing returns as FPKM approaches 100. Left: exon sensitivity as a function of number of predicted exons per gene. Right: transcript sensitivity as a function of the number of predicted transcripts per gene

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

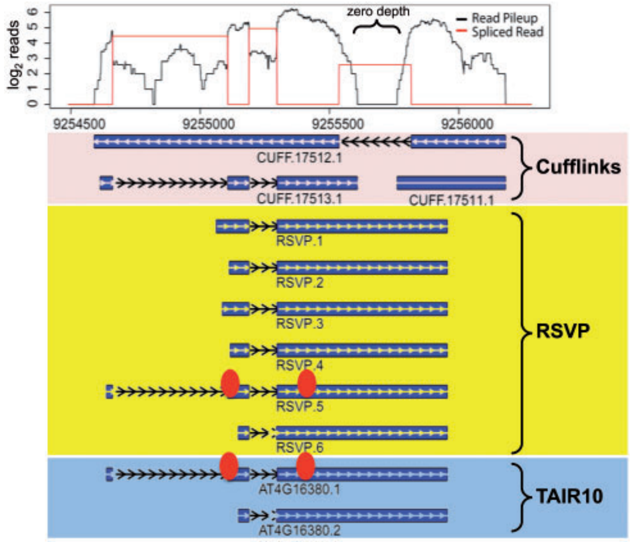

Fig. 4. Example gene (TAIR10: AT4G16380) illustrating differences between Cufflinks and RSVP predictions. None of the three Cufflinks predictions matched the TAIR10 annotations; one was on the opposite strand. RSVP predictions identified several alternative start codons. The gene is a known heavy metal transporter; a putative SCOP domain (oval) involved in metal transport is differentially included by the predicted RSVP isoforms, as well as by the TAIR10 annotations

FPKM results in higher sensitivity, although with diminishing returns as FPKM approaches 100.

The gene in Figure 4 provides some anecdotal insight into the advantages offered by the ORF graph-based approach. None of the three Cufflinks predictions for this locus matched TAIR10 annotations, and one was on the opposite strand from the annotations. RSVP predictions were on the proper strand; all contained valid ORFs; and RSVP identified several alternative start codons with non-zero read depth. In addition, a Structural Classification of Proteins (SCOP) (Murzin *et al.*, 1995) protein domain involved in heavy metal binding (55 008) occurs in one RSVP isoform but is excluded by another (in agreement with TAIR10 annotations), suggesting the possibility of differential protein function based on isoform; the gene (TAIR10: AT4G16380) is annotated as a heavy metal transporter. Note that Cufflinks was unable to extend its prediction into the zero-depth region from 9 255 608 to 9 255 761; our use of both DNA and RNA evidence allowed RSVP to reproduce the annotated isoform extending across this interval.

Table 1 provides counts of alternative isoform features for Cufflinks and RSVP on the FPKM ≥ 1 test set as compared with TAIR10 annotations. For all alternative splicing events—alternative donor or acceptor splice sites, exon skipping and intron retention—RSVP agrees more closely with TAIR10 than does Cufflinks. In contrast, although prediction of alternative start codons and stop codons by Cufflinks exceeds those in TAIR10 by roughly a factor of three, RSVP predicts substantially more of these events than Cufflinks, particularly in the case of start codons. We suspect the TAIR10 annotations are substantially underrepresenting these events, as human curators are typically trained to choose the single most upstream start codon.

Table 1. Frequencies of alternative isoform features predicted by Cufflinks [for the full transcript, including untranslated regions (UTRs)] and RSVP (in CDSs only) in the FPKM ≥ 1 test set, as compared with TAIR10 annotations

Event	Cufflinks	RSVP	TAIR10
Alternative donor splice site	182	456	507
Alternative acceptor splice site	336	1169	1041
Exon skipping	54	139	250
Intron retention	2390	1127	412
Alternative start codon	3011	8839	919
Alternative stop codon	2843	4301	1249

Because Cufflinks does not identify a reading frame, we were compelled to identify the longest ORF in its predictions, likely limiting the frequency of alternative translation start sites in its predictions; as such, the elevated number of start codons identified by RSVP may better correlate with the true prevalence of this type of alternative isoform feature.

Finally, in Figure 5, we show the receiver operating characteristic (ROC) curves for the classification task of discriminating annotated coding loci from either putative non-coding genes (lincRNAs) or from unannotated transcribed loci. For coding genes, we used 3888 TAIR10 loci with FPKM ≥ 1 not in the training set. Classification was performed by thresholding the maximal length-normalized exon score of the RSVP prediction ($\lambda_{\mathrm{RNA}} = 0$); note that although the lincRNAs were filtered by Liu *et al.* based on the longest ORF length, the RSVP exon scores used for classification are likelihood ratios and therefore do not

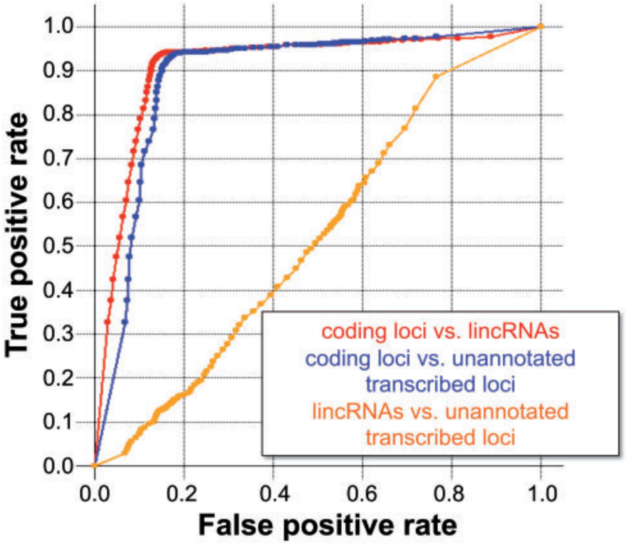

Fig. 5. ROC curves for discrimination of annotated coding loci from lincRNAs and unannotated transcribed loci via RSVP exon score (DNA only: $\lambda_{\mathrm{RNA}} = 0$). AUC: 0.908 for lincRNAs, 0.886 for unannotated transcribed loci. In contrast, the unannotated loci could not be discriminated from the lincRNAs (AUC = 0.520)

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

directly use ORF lengths in the classification. The solid curve plots the true-positive rate as a function of false-positive rate in discriminating the annotated coding loci from the putative lincRNAs. This produced an AUC value (area under the ROC curve) of 0.908, indicating that ORF graphs can be used to reliably discriminate coding from non-coding loci. The dashed ROC curve is the result of discriminating (using RSVP) the same set of annotated coding genes from the unannotated transcribed loci identified by Cufflinks (AUC = 0.886). In contrast, discrimination between the two sets of putative non-coding loci was poor (AUC = 0.520), suggesting that the vast majority of unannotated transcribed loci are non-coding and that few coding genes likely remain to be discovered. See Supplementary Material for additional results.

4 DISCUSSION

Reconstruction of splice isoforms from RNA-seq data has drawn substantial bioinformatic attention of late, yet, despite the proliferation of tools for this task, little attention has been given to combining RNA and DNA evidence in the case of protein-coding genes. We believe this is a glaring shortcoming of current methods, as there is no attempt to model the underlying biological processes (i.e. transcription, splicing, translation) and the patterns that evolution naturally imposes on the nucleotide sequences to accommodate these cellular processes. Sequence patterns such as the Kozak sequence near start codons (Kozak, 1987), polypyrimidine tracts upstream from acceptor sites (Gooding *et al.*, 1998), overall splice site strength (Reed and Maniatis, 1986) and codon biases (Grantham *et al.*, 1980) provide strong cues as to the likely exon–intron structure for any given protein-coding gene. Conversely, the lack of such signals within a transcriptionally active locus may indicate those transcripts are likely to be non-coding.

As we have shown, information extracted from reading frames can improve our ability to infer the correct combinations of exons for the coding portions of transcripts, resulting in higher-accuracy inference of complete coding transcripts and therefore of predicted protein sequences. Note that although these experiments were confined to the genome of the plant *A.thaliana*, our expectation is that the higher combinatorial complexity of mammalian multiexon genes would only skew the results more in our favor, as the underlying ORF graphs provide an effective means of reducing that combinatorial complexity via tracking of reading frames (see Supplementary Material for results supporting that expectation).

Though DNA signal strength in non-coding genes may be weaker than for CDSs with strong codon bias, as our understanding of non-coding transcripts improves, it may well become useful to model sequence characteristics of these transcripts as well. Although we have not tested this here, we suspect that modeling of exon lengths and splice site strengths might alone provide some improvement for prediction of spliced non-coding RNAs. As we have shown in the case of lincRNAs, accurate separation of non-coding from coding transcripts is possible based on scores from our coding transcript model. This may also prove useful in the case of natural antisense transcripts, which may be coding or non-coding (Li *et al.*, 2013).

In the case of protein-coding genes, isoforms may differ not only in their splicing patterns but also in their reading frame and length, the latter feature being most obviously affected by the choice of translation start site (start codon). Although human annotators are typically trained to choose the most upstream methionine as the ‘correct’ start codon (in accordance with the traditional ‘ribosome scanning’ model—e.g. Kozak, 1989), this is not a biologically absolute rule. Several plant genes have now been shown to support stable translation from multiple start codons (Yashitola *et al.*, 2009), in some cases resulting in differential protein targeting in the cell. Translation initiation will also be constrained by the transcription initiation site, so that shortening of transcripts at the 5′ end will force the use of more 3′ translation start sites. The use of alternative translation start sites will obviously affect the encoded protein and, in some instances, may cause the differential inclusion of functional protein domains, in particular signal peptides that influence the subcellular localization of the resulting protein. Therefore, it is valuable for a predictor to be able to identify isoforms differing based on the use of strong alternative start codons.

A prominent feature of our proposed framework is the ability to exploit the sensitivity-specificity trade-off inherent in choosing the number of predictions to be made. Although direct measurement of the specificity is generally not possible, because of the incomplete state of most genomic annotation efforts to date, the eventual sharp falloff in the rate of increase of sensitivity as numbers of predictions continue to rise indicates both a likely concomitant drop in specificity and also that a point of saturation has likely been reached for the expressed genes. Plots from Figure 2 suggest that current methods based only on RNA evidence may still be missing a substantial fraction of expressed isoforms.

The advantage of combining RNA and DNA evidence for protein-coding transcript prediction should be particularly large in the regime of lowly expressed genes or in the presence of highly non-uniform read coverage along a gene (e.g. as might result from sequencing or amplification bias), in which low sequencing coverage over specific parts of genes may leave those exons effectively invisible to conventional transcript assembly algorithms. In contrast, the use of ORF graphs allows one to ‘complete the picture’ by finding exons with high-coding potential and reasonably strong splice sites that result in a complete ORF, despite gaps in RNA coverage (e.g. the 153 nt interval of zero read depth within the annotated coding exon in Figure 4, which caused Cufflinks to prematurely terminate its prediction). The use of a probabilistic framework such as ours provides a means of doing so in a seamless principled manner.

There are a number of potential improvements to our approach that could be investigated, most prominent among them being the incorporation of additional information. Just as we have shown that integration of DNA and RNA evidence can benefit prediction, so the additional integration of yet other forms of evidence, such as evolutionary conservation, should prove fruitful. Previous and ongoing work in the field of gene prediction consistently shows that combining multiple lines of evidence tends to produce higher-quality gene structure predictions (e.g. Allen *et al.*, 2006; Zeller *et al.*, 2013), and this very likely extends to the case of predicting multiple isoforms. Additional information derived via the use of specialized RNA library preparation techniques such as paired-end sequencing could in principle be incorporated

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

via appropriate augmentation of the ORF graph; these remain promising avenues for future work.

Unlike the approach typically taken in the gene-finding literature, where gene-finding programs are monolithic programs with few independently reusable parts, the use of ORF graphs permits a modular approach to transcript isoform discovery. Just as RSVP uses an independent reweighting step to update the scores decorating the edges of the ORF graph so as to reflect RNA evidence, additional graph transformations could be modularly applied to incorporate additional evidence; the base ORF graph produced by our pipeline may thus be used by other groups to investigate alternative methods of graph-based prediction.

Finally, we believe that ORF graphs may prove useful not only as substrates for prediction but also as repositories for later re-annotation in the light of additional evidence, when that evidence becomes available. Their ability to be stored, annotated and decoded efficiently in both space and time renders them a highly useful substrate for an array of possible downstream analyses.

Funding: Funding from the NSF Arabidopsis 2010 project (IOS-1021619).

Conflict of Interest: none declared.

REFERENCES

Allen,J. *et al.* (2006) JIGSAW, GeneZilla, and GlimmerHMM: puzzling out the features of human genes in the ENCODE regions. *Genome Biol.*, **7** (**Suppl. 1**), S9.1–S9.13.

Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Burge,C. (1997) *Identification of Complete Gene Structures in Human Genomic DNA*. PhD thesis. Stanford University, Stanford, CA.

Gooding,C. *et al.* (1998) Role of an inhibitory pyrimidine element and polypyrimidine tract binding protein in repression of a regulated alpha-tropomyosin exon. *RNA*, **4**, 85–100.

Grantham,R. *et al.* (1980) Codon catalog usage and the genome hypothesis. *Nucleic Acids Res.*, **8**, r49–r62.

Guttman,M. *et al.* (2010) *Ab initio* reconstruction of cell type–specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincRNAs. *Nat. Biotechnol.*, **28**, 503–510.

Kim,D. *et al.* (2013) TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. *Genome Biol.*, **14**, R36.

Kozak,M. (1987) An analysis of 5'-noncoding sequences from 699 vertebrate messenger RNAs. *Nucleic Acids Res.*, **15**, 8125–8148.

Kozak,M. (1989) The scanning model for translation: an update. *J. Cell. Biol.*, **108**, 229–241.

Lamesch,P. *et al.* (2012) The arabidopsis information resource (TAIR): improved gene annotation and new tools. *Nucleic Acids Res.*, **40**, D1202–D1210.

Langmead,B. and Salzberg,S.L. (2012) Fast gapped-read alignment with Bowtie 2. *Nat. Methods*, **9**, 357–359.

Li,H. *et al.* (2009) The sequence alignment/map (SAM) format and SAMtools. *Bioinformatics,*, **25**, 2078–2079.

Li,S. *et al.* (2013) Integrated detection of natural antisense transcripts using strand-specific RNA sequencing data. *Genome Res.*, **23**, 1730–1739. doi: 10.1101/gr.149310.112.

Liu,J. *et al.* (2012) Genome-wide analysis uncovers regulation of long intergenic noncoding RNAs in *Arabidopsis*. *Plant Cell*, **24**, 4333–4345.

Majoros,W.H. *et al.* (2004) TIGRscan and GlimmerHMM: two open source *ab initio* eukaryotic gene finders. *Bioinformatics*, **20**, 2878–2879.

Majoros,W.H. *et al.* (2005) Efficient decoding algorithms for generalized hidden Markov model gene finders. *BMC Bioinformatics*, **6**, 16.

Majoros,W.H. (2007) *Methods for Computational Gene Prediction*. Cambridge University Press, Cambridge, UK.

Marquez,Y. *et al.* (2012) Transcriptome survey reveals increased complexity of the alternative splicing landscape in Arabidopsis. *Genome Res.*, **22**, 1184–1195.

Mezlini,A.M. *et al.* (2012) iReckon: simultaneous isoform discovery and abundance estimation from RNA-seq data. *Genome Res.*, **23**, 519–529.

Murzin,A.G. *et al.* (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, **247**, 536–540.

Reed,R. and Maniatis,T. (1986) A role for exon sequences and splice-site proximity in splice-site selection. *Cell*, **46**, 681–690.

Trapnell,C. *et al.* (2010) Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat. Biotechnol.*, **28**, 511–515.

Viterbi,A.J. (1967) Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. Inf. Theory*, **13**, 260–269.

Yashitola,W. *et al.* (2009) Participation of leaky ribosome scanning in protein dual targeting by alternative translation initiation in higher plants. *Plant Cell*, **21**, 157–167.

Zeller,G. *et al.* (2013) mTim: rapid and accurate transcript reconstruction from RNA-Seq data. http://arxiv.org/abs/1309.5211 (16 April 2014, date last accessed).

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016