A Transcript Perspective on Evolution

Yann Christinat and Bernard M.E. Moret

Laboratory of Computational Biology and Bioinformatics
EPFL, 1015 Lausanne, Switzerland

Abstract. Alternative splicing is now recognized as a major mechanism for transcriptome and proteome diversity in higher eukaryotes. Yet, its evolution is poorly understood. Most studies focus on the evolution of exons and introns at the gene level, while only few consider the evolution of transcripts.

In this paper, we present a framework for transcript phylogenies where ancestral transcripts evolve along the gene tree by gains, losses, and mutation. We demonstrate the usefulness of our method on a set of 805 genes and two different topics. First, we improve a method for transcriptome reconstruction from ESTs (ASPic), then we study the evolution of function in transcripts. The use of transcript phylogenies allows us to double the specificity of ASPic, whereas results on the functional study reveal that conserved transcripts are more likely to share protein domains than functional sites. These studies validate our framework for the study of evolution in large collections of organisms from the perspective of transcripts; we developed and provide a new tool, TrEvoR, for this purpose.

Keywords: alternative splicing, transcript, evolution, phylogeny, protein domain, transcriptome reconstruction.

1 Introduction

Gene duplication and loss are the main driving forces for transcriptome and proteome diversity. However, alternative splicing—a greatly underestimated mechanism twenty years ago—has now been shown to play a major role for diversity in higher eukaryotes [1,2]. In many genomes, most genes are thus split into introns and exons. The standard splicing scheme keeps all exons and removes all introns, but alternative splicing permits removal of alternative exons. Some mRNAs are further translated into proteins—named isoforms—and alternative proteins can therefore vary in large regions or may even not overlap.

Alternative splicing is limited in plants and fungi but quite common in vertebrates. Some researchers conjecture that 90% of human multi-exon genes are alternatively spliced [3]. Yet the study of evolution from a transcript perspective has not seen much work, while the evolution of the mechanism itself is poorly understood. Several articles focus on the evolution of the gene structure—exons and introns—but none address the problem of transcript evolution [3]. The few studies on this matter are limited to mouse and human and agree on the fact that

L. Bleris et al. (Eds.): ISBRA 2012, LNBI 7292, pp. 48–59, 2012.

alternative splicing is a fast evolving mechanism [4,5]. For instance, Nurtdinov *et al.* showed that only three quarters of the human isoforms have an ortholog in mouse [6].

The number of alternative isoforms is specific to species but also unevenly documented. For instance, the Ensembl database [7] reports numerous transcripts for human, mouse, rat, several apes, and a few fishes but almost no alternative splicing for dogs or cats. Some gene families also display a very different rates of alternative splicing across their species; a member can be extensively spliced in a species and have only one transcript in another species. The available data comes mainly from experiments and it is expected to be incomplete. Current automated pipelines for transcriptome reconstruction are not trusted, so that large-scale multi-species analysis is not doable at present. Confronted with the incompleteness of the data and the presence of extensively spliced genes, some researchers conjecture that all alternative transcripts are possible and that the observed set reflects a regulated distribution of all transcripts. Nonetheless, some transcripts are conserved among homologous genes. The question remains to be answered whether the function—be it gene ontologies, tissue or sub-cellular localization, or developmental stages—is also conserved or if they just represent noise in the splicing machinery.

In this paper, we extend our previous work [8] and present a transcript evolution framework where ancestral transcripts evolve along the gene tree through transcript gains or losses, and exon gains or losses. This entire process is represented by a forest of transcript phylogenies that links the observed transcripts of a gene family. This framework has applications in many transcript-related fields. Among these last, we selected two to demonstrate the benefits gained through our method. In Section 3.1, we address the problem of transcriptome reconstruction, as it represents a core issue for transcript analysis. We refined the output of ASPic (the Alternative Splicing Prediction DataBase [9]): as measured using the RefSeq database [10], our method doubles the specificity of ASPic. In Section 3.2, we study the distribution of protein domains in transcript phylogenies, using many Ensembl tracks. Our results indicate that transcripts are more likely to conserve their domains than their functional sites through evolution.

These two studies establish the usefulness of our transcript phylogeny framework for large-scale biological analyses. The tool we developed for these analyses TrEvoR (Transcript Evolution Reconstruction software), is publicly available and can be downloaded at `http://lcbb.epfl.ch/trevor`.

2 A Model of Transcript Evolution

Our model of transcript evolution is a refinement of the extended model we presented in [8]. Given the set of transcripts of the most ancient gene, a transcript evolves along the gene tree through three simple events: mutation (gain or loss of exons), fork (creation of new transcripts), and death (loss of transcripts). This process is represented in a set of transcript trees, one for each ancestral

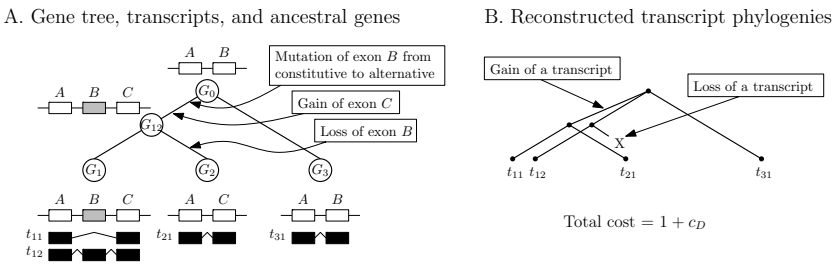

Fig. 1. Illustration of the two-level model. The first level is represented in A where the gene evolution happens. In B, one can see the transcript phylogeny. Transcripts t_{12} and t_{31} differ by exon C which was gained during the evolution from G_0 to G_{12}. This event belong to the first level and thus has zero cost in the transcript phylogeny. There one transcript loss (cost $= c_D$) and there is a new splicing pattern in G_{12} which cannot be explained by the evolution of the gene structure. The total edge costs is thus $1 + c_D$.

transcript. Mutations happen along the branches and affect the content of transcripts whereas forks and deaths affect the structure of the transcript trees.

In the absence of prior work, we opted for a maximum parsimony framework. Every event is assigned a unit cost, except for transcript death, which is the sole event of the model to be parameterized (c_D). Other frameworks such as maximum likelihood or Bayesian networks can be used, but they tend to yield higher computational costs.

Our model aims at reflecting the cost of transcript evolution alone; thus the cost of gene evolution is discarded. This implies a two-level model, where the evolution of the gene structure serves as a basis for the evolution of the transcriptome. For instance the loss of an exon at the gene level implies that all transcripts lose this exon. In a classical maximum parsimony framework, these events would add their cost to the score. In our model, however, they do not since they are the unavoidable consequence of a gene event. This concept is illustrated through an example in Figure 1.

3 Results

3.1 Transcriptome Reconstruction

Next generation sequencing methods yield an increasing amount of data and reconstructing transcripts from short reads is a complex problem [11]. Once ESTs are mapped on the genome and splice junctions are identified, a splice graph can be constructed—nodes represent exons and edges splice junctions. Any path on this graph is thus a potential transcript. The remaining problem is to identify the "true" transcripts within this graph.

Several methods exist to predict transcripts from ESTs. ESTGene, which is part of the Ensembl pipeline, reconstructs the minimal set of transcripts that

cover the splice graph [12,7]. ECGene, another method, parses the splice graph and clusters transcripts based on the nature of the splice sites [1]. ATP, the algorithm behind the ASPic database, is similar to ESTGene but includes additional rules to predict transcripts [11,9]. Other methods such as Scrip- ture [13], Cufflinks [14], or the EM algorithm by Xing *et al.* [15] also aim at transcriptome reconstruction but have no associated database. However, none of these methods make use of phylogenies.

Methods. We selected from the ASPic database 805 human genes that have an ortholog in rat, mouse, opossum, chimpanzee, marmoset, and macaque. These six species had the highest rate of alternative splicing in the Ensembl database while being the closest to human. Exons and transcripts were collected from the ASPic database for human and from Ensembl for the remaining species. Genes were aligned using MAFFT and orthologous exons were established when reciprocally overlapping at 70%. Alternative 3'- or 5'-end exons were assigned when the overlap was not reciprocal.

We refine the prediction of ASPic through a simple algorithm. For each human transcript present in the ASPic database, we collected the Ensembl transcripts of the homologous genes and reconstructed a transcript phylogeny on this set plus the ASPic transcript. The total cost of this transcript phylogeny—as defined in our model—is assigned as the score of the ASPic transcript. Once every ASPic transcript is assigned a score, the algorithm discards all transcripts that have an unreasonable evolutionary score within the transcript set of a given gene. Since the evolutionary score is dependent of the number of exons and may vary greatly between genes, we considered the score ratio. Consequently the algorithm searches for the two groups of transcripts that maximize the ratio of their respective mean scores—a 2-mean clustering algorithm. If that ratio is larger than t, then the high-cost group is discarded. Therefore, if $t = \infty$ the performance of the refinement algorithm will be equal to the original algorithm since no transcript is removed.

Results. We tested the performance of ECGene and ASPic by matching their predicted transcripts (from their online database) to the RefSeq database [10]—a gold standard for RNA sequences. A true positive was defined as an exact sequence match between the query and a sequence in RefSeq. A false positive is thus a transcript predicted by ASPic that could not be matched in RefSeq.

As shown in Table 1, both methods performed quite poorly. The best method, ASPic, can only recover 7% of all refseq transcripts. The use of transcript phylogenies yielded a significant increase in specificity while withstanding a minor decrease in sensitivity. Note that since we filter the ASPic's transcripts, sensitivity cannot increase. The Ensembl database had a specificity of 97% on the same set of genes. It is however impossible to assess the part played by ESTGene as the Ensembl database includes all sequences from RefSeq. The ECGene results could not be refined as exon information was not available; only transcript RNA sequences are available for download.

Table 1. Results of the different algorithms using exact matches on the RefSeq database. Our refinement on the ASPic prediction yielded a slight decrease in sensitivity and a 2-fold increase in specificity. Both refined results were achieved with the same threshold t but different transcript loss cost.

	Sensitivity	Specificity
ECGene	0.9%	1.2%
ASPic	7.23%	5.2%
Ref. ASPic ($c_D = 1$)	6.4%	11.5%
Ref. ASPic ($c_D = 10$)	7.0%	9.7%

Following Bonizzoni *et al.* [11], we opted for a gentler setup where exact matches are not required at the end of the sequences. That is, a predicted transcript is a true positive if it is a substring of a sequence in RefSeq and if no additional exons could have been added by the algorithm. That is, there exists no predicted transcript that contains the missing ends. Under this setup, ASPic performed better—13.7% specificity and 22.8% sensitivity—and our refinement method pushed the specificity to 29.8%, again a two-fold increase, while lowering the sensitivity to 19.7%.

We then varied the threshold t in our refinement algorithm and investigated the influence of different models and phylogenies on the performance of the refinement step. We tested two evolutionary models,

- two-level model: Our standard model of transcript evolution where gene events have zero cost.
- simple model: Our standard model of transcript evolution but with gene cost. For instance, the cost of losing an exon at the gene level is passed onto each transcript.

and three different phylogenies.

- Best tree: The reconstructed phylogeny output by our algorithm.
- Random tree: Random transcript trees that still agree with the gene tree. (Average on 100 runs)
- Star tree: All transcripts are directly linked to the root.

The ROC curves in Figure 2 show that the two-level model performs better than the simple model. The star phylogeny outperforms random trees. In random trees, the signal is completely lost as orthologous transcripts may not be on the same tree, thus conveying wrong information, whereas, on a star phylogeny, no information is provided beyond the contents of the transcripts. It performs then a simple comparison of the human transcript to all non-human transcripts. This is enough to gain some specificity but it cannot reach the performance of the best tree setup—be it on the two-level or the simple model. If a random subset of ASPic transcripts was selected, the sensitivity would simply diminish without any gain in specificity.

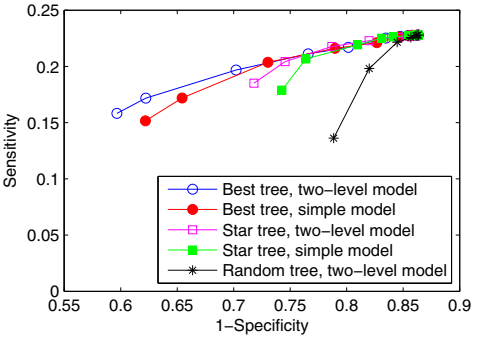

Fig. 2. Loose matches on RefSeq under the different setups and thresholds. All curves converge towards the ASPic performance at the top right corner. An optimal curve would go horizontally from the ASPic performance towards the left.

The previous experiment tested how a single transcript fits within the phylogeny. However, transcriptome reconstruction is about sets of transcripts. Therefore, we computed the cost of evolution for the Ensembl transcripts in the homologous genes plus different sets of human transcripts: all ASPic transcripts (ALL), no human transcripts (NO), and ASPic transcripts that were matched in RefSeq under the exact setup ($EXCT$) and loose setup (LOO). In this experiment, TrEvoR is only used to score different sets of human transcripts; our refinement algorithm is not applied. Note that human is the only species to be affected by these changes since we selected only human genes from the ASPic predictions. The NO setup corresponds to an evolutionary scenario where all human transcripts were lost in this gene family. The expected result is that the set of exactly matched transcript should have the lowest cost. We expect the loose match setup to sometimes have the lowest cost, as the RefSeq may not be exhaustive. We ran the algorithm on the 213 genes that had at least one exact match and observed that with a transcript loss cost of 1, removing all transcripts yielded the minimum cost in 76% of the cases. Nonetheless with a transcript loss cost of 10, 94% of the 213 genes had a exact match score lower than the loose match score and lower than the empty set. Table 2 summarizes these results and shows that the choice of the transcript loss cost (c_D) is an important matter.

To conclude, we demonstrate that transcript phylogenies can enhance transcriptome reconstruction from ESTs. Focusing on transcript evolution and discarding the cost of gene evolution yielded better results. A direction for future work is to integrate the transcript phylogenies directly into the reconstruction method. That way, both sensitivity and specificity could be increased.

3.2 Functional Study on Transcripts

To demonstrate the broad scope of our framework, we applied transcript phylogenies to the study of function in transcripts. We inquired whether transcript

Table 2. Statistics on the evolutionary cost of different subsets of the predicted transcripts. The first three conditions look at the expected behavior. (The *LOO* and *EXCT* setups should have the lowest score.) The last condition tests whether it is always profitable to remove transcripts. Values indicate the percentage of genes that fit the condition.

Condition	$c_D = 1$	$c_D = 10$
$s_{ALL} > s_{LOO} < s_{NO}$	21%	95%
$s_{ALL} > s_{EXCT} < s_{NO}$	23%	98%
$s_{ALL} > s_{EXCT} < s_{LOO}$	23%	94%
s_{NO} is min	76%	2%

phylogenies carry any functional information and, in the positive case, if two different transcript trees vary in functions. We thus studied the correlation between protein domains—structurally stable regions that often correspond to specific functions—and transcript phylogenies. Interestingly, conserved exons also displayed high correlation with protein domain boundaries [4].

Methods. We used the same dataset, 7 species and 805 genes, as for the transcriptome reconstruction problem. Transcripts and domain annotations were retrieved from the Ensembl database and transcript phylogenies were reconstructed with our algorithm. The sole parameter, the cost of transcript loss, was set as a proportion of the average number of exons in a set of homologous genes, denoted as $c_D = \alpha E[exs]$, and three values of α were tested: 0.1, 1, and 10. Different domain annotation databases, all available as tracks in Ensembl, were selected:

- *InterPro*: An integrated resource for protein families, domains, regions, and functional sites. It combines data from several databases such as *PROSITE*, *PRINTS*, *SMART*, *SUPERFAMILY*, *Pfam* and many others.
- *Pfam*: A database of protein families identified by sequence alignments and hidden Markov models.
- *PROSITE*: A collection of protein domains, families, and functional sites identified through patterns and profiles.
- *SMART*: A database of well annotated protein domains.
- *SUPERFAMILY*: A set of hidden Markov models that represent domains at the superfamily level in SCOP (structure-based classification of proteins).
- *SEG*: A software that divides the sequence into low- and high- complexity regions.
- *Transmembrane*: Identification of transmembrane helices with TMHMM.

We tested the transcript phylogenies for robustness and found 135 gene families where transcripts were grouped under the same parents across the three different transcript loss costs (c_D). A higher c_D may force two trees to be reunited under the same tree which is the reason why we only look at the last two levels of the transcript phylogenies. These genes are good candidates for having more than one well-conserved ancestral transcript.

We want to test whether the ancestral transcripts had the same functional content or not. If equal then the distribution of the domain content in each tree should be roughly equal to the distribution of the domains across all transcripts. Therefore, for each gene family F, we computed the probability of a domain d to appear in a transcript. This is our background probability,

$$P_F[d] = \frac{\text{Nb of transcripts in } F \text{ that contain } d}{\text{Nb of transcripts in } F} \tag{1}$$

A similar value was computed for each tree, $P_t[d]$. Note that we only account for the presence of the domain. The number of occurrences per transcript does not matter.

We selected phylogenies with multiple trees and computed, for all domains, the deviation of each tree from the background probability. For a given domain, we have thus

$$Dev[d] = \frac{1}{|T|} \sum_{t \in T} (P_t[d] - P_F[d])^2 \tag{2}$$

where T represent the set of transcript trees. The mean value over all domains, $E[Dev]$, gives us an indication of the global deviation of the domain content in the trees from the domain content in all transcripts.

In order to test if the 135 "stable" genes had a different deviation from the rest, we performed a random sampling among the 805 genes. The sampling was repeated a hundred times then averaged. We refer to this sampling as the "random set".

Results. As can be seen in Figure 3, the deviation between the stable and the random sets differ mainly for low values of c_D. The stable set, except for *PROSITE*, has always a lower deviation than the average on the random set. As the cost increases, the stable set converges towards the random set. Remarkably, *Transmembrane* and *SEG* did not display any significant differences between random and stable sets for any values of c_D. A study by Cline *et al.* showed that transmembrane regions are not likely correlated with alternative splicing [16]—a finding that endorses our results. Interestingly, the *PROSITE* database is the only one to exhibit a higher deviation for the stable set (at $c_D = E[exs]$). It is also the only database to include functional sites. The *InterPro* database does include the *PROSITE* database but its behavior resembles the other databases. *InterPro* collects data from 11 databases. The *PROSITE* functional sites annotations may thus be a minority and have little influence on the global result.

To test if the difference in *PROSITE* for $c_D = E[exs]$ was indeed significant and not a visual artifact, we performed a one-way ANOVA on *PROSITE*, *Pfam*, *SMART*, and *SUPERFAMILY* and another on the same databases but without *PROSITE*. The first analysis returned a p-value of $1.5036 \cdot 10^{-28}$ while the second returned a value of 0.1632. Consequently we can confidently reject the null hypothesis—all means are equal—in the first case but not in the second. This indicates that the *PROSITE* database has a significantly larger deviation than the other databases between the two sets. Note that these databases may

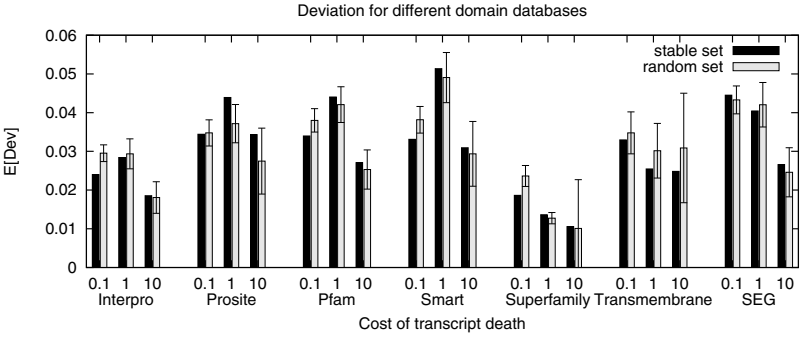

Fig. 3. Deviation from the expected domain presence for different domain annotation databases. For each database, three setups for the transcript loss cost were tested for the stable and the random sets: $c_D = 0.1E[exs]$, $c_D = 1E[exs]$, and $c_D = 10E[exs]$. The *PROSITE* is the only database to display a significantly higher deviation for the stable set.

cover similar domains and consequently may not be independent, which poses a problem for statistical analyses.

All databases, except *PROSITE*, have a lesser deviation than the random set for low c_D value. The *PROSITE* exception can be explained by an averaging effect. Proteins domains tend to yield a smaller deviation while functional sites create a larger deviation. The combination of both averages the score and results in a deviation similar to the random set. The *SUPERFAMILY* has, globally, a lower deviation than any other database. That is to be expected as the *SUPERFAMILY* database clusters domains with a higher abstraction level than families. That is, two different domain families may be reunited under a single superfamily and thus lower the deviation.

We also tested the stable set against the 805 genes for GO enrichment with GOrilla [17] but could not find any over- or under-represented ontologies.

Based on these results, one could conjecture that well conserved transcripts contain similar sets of domains but different functional sites. A deeper study could focus on the functional sites and potentially identify unknown functions in some transcripts.

4 Reconstruction of Transcript Phylogenies

Reconstructing transcript phylogenies is nontrivial: the problem is at least as hard as the standard phylogeny reconstruction problem, which is NP-hard. In a standard tree reconstruction, the tree structure is unknown but we know that only one tree exists. In a transcript phylogeny reconstruction, the tree structure is partially known, as it has to be a subtree of the gene tree, but the number of trees is unknown. Our previous algorithm did not scale well, hence we designed a heuristic based on neighbor-joining and packaged it into a convenient tool:

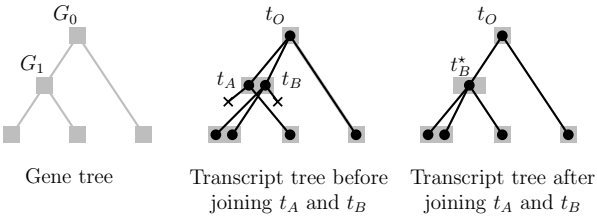

Fig. 4. A join operation on a simple transcript tree. $\textsc{Join}(t_A, t_B)$ is valid because they share a common ancestor and belong to the same gene, G_1. Note that the two transcript deaths are lost after the operation.

TrEvoR (Transcript Evolution Reconstruction). The latter is available online and can be downloaded at `http://lcbb.epfl.ch/trevor`. A manual and some toy examples are also present.

4.1 TrEvoR Algorithm

Our two-level model of transcript evolution depends on the gene structure and, similar to our previous algorithm, the first step is thus to reconstruct the exons of the ancestral genes. Sankoff's algorithm for the small parsimony problem is applied and backtracking yields the ancestral states [18]. Our algorithm searches then for the most parsimonious forest of transcript trees.

Parsimony methods for the "standard" phylogeny reconstruction problem use a specific operation to search the tree space—nearest neighbor interchange, subtree pruning and regrafting, or tree bisection and reconnection. In our case, we define the "join" operation, which given an ancestral gene, merges two of its transcripts. The join operation simply assigns all children of transcript t_A to transcript t_B, deletes transcript t_A, and updates transcript deaths in t_B. Note that a join operation is only possible if the two transcripts share a common ancestor. Figure 4 illustrates the join operation on a simple example.

In the initialization step of the algorithm, every current transcript has its own ancestor (trivial trees). A neighbor-joining algorithm with the "join" operation is then applied at the root of the gene tree. At each iteration all possible join operations on two trees are tested, the best candidates are retained, their root are joined, and the algorithm moves to the next iteration. Similar to the leaf assignment procedure of the first algorithm, the score of a tree is tested by propagating the join operation from the root to the leaves. For each possible join operation on two roots, we apply a recursive neighbor-joining algorithm to find the best transcript tree (Algorithm 1). Note that any join operation that was done when computing the score of the transcript tree is undone before passing to the next iteration.

Algorithm 1. A recursive Neighbor-Joining algorithm using the JOIN operation.

```
1: procedure RecNJ(Transcript t)
2:     if no children of t can be joined then
3:         return ComputeScore(t)   ▷ Compute the score of the tree containing t.
4:     s_best = ∞
5:     while some children of t can be joined do
6:         s* = ∞
7:         for (a, b) s.t. a, b ∈ ChildrenOf(t) do
8:             Join(a,b)                    ▷ Assume that joining a and b is feasible.
9:             s = RecNJ(b)
10:            unJoin(a,b)                  ▷ Revert to the situation before joining a and b.
11:            if s < s* then
12:                s* = s
13:                a* = a and b* = b                              ▷ Save the best join.
14:        Join(a*,b*)                                            ▷ Apply the best join.
15:        s = RecNJ(b*)                                          ▷ Iterate on the "new" node.
16:        if s < s_best then
17:            s_best = s
18:    return s_best
```

5 Conclusion

We presented a model of transcript evolution and an associated tool, TrEvoR, to reconstruct transcript phylogenies. The model represents the evolution of transcripts as a second layer above the exon evolution.

On 805 genes from the ASPic database, we demonstrated that transcript phylogenies can enhance transcriptome reconstruction from ESTs. The use of transcript phylogenies doubled the specificity while retaining a similar sensitivity. Results also showed that our two-level model performed better than a gene-centric model. This implies that a transcript-focused approach is more powerful for this particular task.

Additionally, we broadened the scope of transcript phylogenies by correlating them with the protein domains of their isoforms. It turned out that transcript trees indeed contain useful functional information and may be used in studies on function evolution. Domain information was gathered from different tracks in Ensembl and results revealed that conserved transcripts show a greater variability in functional sites than in protein domains.

Future work can be directed in several directions. Different models—for instance, a model based on splice sites and not exons—and different hypotheses can be tested through TrEvoR. The accuracy of automated pipelines for transcriptome reconstruction could be improved by developing a method that includes transcript phylogenies of model organisms. Deeper studies on functional sites within a transcript phylogeny framework could shed some light on the evolution of functions.

In previous work, we proposed the concept of transcript phylogenies and demonstrated its feasibility. Here, we applied this concept to two large-scale

analyses, demonstrated good improvements on transcriptome reconstruction, new findings on the evolution of function in transcripts, and consequently validated the usefulness of our method for transcriptome studies.

References

1. Kim, N., Shin, S., Lee, S.: ECgene: genome-based EST clustering and gene modeling for alternative splicing. Genome Research 15(4), 566–576 (2005)
2. Modrek, B., Lee, C.: A genomic view of alternative splicing. Nature Genetics 30(1), 13–19 (2002)
3. Keren, H., Lev-Maor, G., Ast, G.: Alternative splicing and evolution: diversification, exon definition and function. Nature Reviews Genetics 11(5), 345–355 (2010)
4. Artamonova, I.I., Gelfand, M.S.: Comparative genomics and evolution of alternative splicing: the pessimists' science. Chemical Reviews 107(8), 3407–3430 (2007)
5. Harr, B., Turner, L.M.: Genome-wide analysis of alternative splicing evolution among Mus subspecies. Molecular Ecology 19(suppl.1), 228–239 (2010)
6. Nurtdinov, R.N.: Low conservation of alternative splicing patterns in the human and mouse genomes. Human Molecular Genetics 12(11), 1313–1320 (2003)
7. Flicek, P., et al.: Ensembl 10th year. Nucleic Acids Research 38(suppl.1), D557–D562 (2010)
8. Christinat, Y., Moret, B.: Inferring transcript phylogenies. In: Proc. of IEEE International Conference on Bioinformatics and Biomedecine, pp. 208–215 (2011)
9. Martelli, P., et al.: ASPicDB: a database of annotated transcript and protein variants generated by alternative splicing. Nucleic Acids Research 39(suppl.1), D80 (2011)
10. Pruitt, K., et al.: NCBI Reference Sequences: current status, policy and new initiatives. Nucleic Acids Research 37(suppl.1), D32–D36 (2009)
11. Bonizzoni, P., et al.: Detecting alternative gene structures from spliced ESTs: a computational approach. Journal of Computational Biology 16(1), 43–66 (2009)
12. Eyras, E., et al.: ESTGenes: alternative splicing from ESTs in Ensembl. Genome Research 14(5), 976–987 (2004)
13. Guttman, M., et al.: Ab initio reconstruction of cell type-specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincrnas. Nature Biotechnology 28(5), 503–510 (2010)
14. Trapnell, C., et al.: Transcript assembly and quantification by rna-seq reveals unannotated transcripts and isoform switching during cell differentiation. Nature Biotechnology 28(5), 511–515 (2010)
15. Xing, Y., et al.: An expectation-maximization algorithm for probabilistic reconstructions of full-length isoforms from splice graphs. Nucleic Acids Research 34(10), 3150 (2006)
16. Cline, M., et al.: The effects of alternative splicing on transmembrane proteins in the mouse genome. In: Pac. Symp. Biocomput. 2004, pp. 17–28 (2004)
17. Eden, E., et al.: GOrilla: a tool for discovery and visualization of enriched GO terms in ranked gene lists. BMC Bioinformatics 10(1), 48 (2009)
18. Sankoff, D.: Minimal Mutation Trees of Sequences. SIAM Journal on Applied Mathematics 28(1), 35–42 (1975)