EuGène: an eucaryotic gene finder that combines several sources of evidence

Thomas Schiex[1], Annick Moisan[1], Lucien Duret[1], and Pierre Rouzé[2]

[1] INRA, Toulouse, France,
Thomas.Schiex@toulouse.inra.fr,
http://www-bia.inra.fr/T/schiex
[2] INRA, Gand, Belgique
Pierre.Rouze@gengenp.rug.ac.be

Abstract

In this paper, we describe the basis of **EuGène**, a gene finder for eucaryotic organisms applied to *Arabidopsis thaliana*. The specificity of **EuGène**, compared to existing gene finding software, is that **EuGène** combines the output of several information sources, including output of other software, in a weighted directed acyclic graph (DAG) designed in such a way that a shortest path in this graph represents the most likely gene structure of the underlying ADN sequence.
The usual simple Bellman linear time shortest path algorithm for DAG has been replaced by a shortest path with constraints algorithm. The constraints express minimum length of introns or intergenic regions. The specificity of the constraints leads to an algorithm which is still linear both in time and space.
EuGène effectiveness has been assessed on Araset, a recent dataset of *Arabidopsis thaliana* sequences used to evaluate several existing gene finding software. It appears that, despite its simplicity, **EuGène** gives results which compare very favorably to existing software. We try to analyze the reasons of these results.

Motivations

It is standard, in a thorough sequence annotation, to take into account several sources of evidence in order to try to precisely locate genes (exons/introns) in eucaryotic sequences. The sources exploited typically include:

- matches against databases (cDNA and protein databases);
- output of splice sites or translation start prediction software;
- more or less sophisticated "integrated" gene finding software, eg. GeneMark.hmm [6].

None of these sources of evidence is, alone, sufficient to decide gene locations and the manual integration of all these data is a painful and extremely slow work. The motivation of our work is, as far as possible, to automate this job using *Arabidopsis thaliana* as a first test organism.

1 Methods

Along this line of idea, we have designed a simple, general, efficient and yet effective graph-based approach for gene finding that allows to combine several sources of evidence. Rather than directly combining the output of existing gene finding software (as in [7]) we decided to combine the information at the lowest level in order to be able to:

1. maintain the consistency of the prediction;
2. globally assess the impact of each local choice w.r.t. all available evidence.

Given a raw DNA sequence, the basic idea is to consider a directed acyclic weighted graph such that all possible consistent gene structures are represented by a path in the graph. The gene structure currently used in **EuGène** is the simplest reasonable structure, the only signals taken into account being ATG, stops and splice sites. It can handle multiple genes which may be partial on the extremities of the sequence.

The directed acyclic graph used for a simple ad-hoc sequence is illustrated above. It is a series-parallel graph with 13 different tracks that correspond respectively to the 6 forward/reverse coding frames, 6 forward/reverse intronic phases and a non coding track. Each signal occurrence, between two successive nucleotides, generates one or more "switches" between two parallel tracks. Each source-sink path defines a sequence of consistent genes structures. The size of the graph is in $O(n)$ where n is the sequence length.

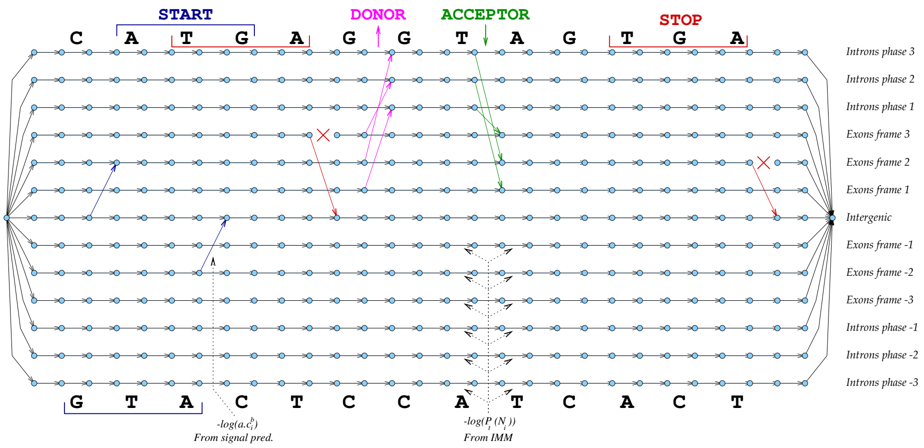

Fig. 1. The DAG explored by **EuGène** for a simple sequence

To choose one path among the $O(13^n)$ paths in this graph, each edge e is weighted by a positive number w_e in such a way that shortest paths in the graph correspond to gene structures that "best respect" the available evidence.

A probabilistic interpretation of this model can be given as follows: each edge e has a probability of existence P_e. Under simple independence assumptions, the reliability of a source-sink path is simply defined by the product of all the P_e in the path. A most reliable path is then a shortest path in the graph where $w_e = -\log(P_e)$. The approach is comparable (although not equivalent) to an explicit state duration HMM with uniform duration densities (see [11], pp. 270) with a non-homogeneous transition matrix between hidden states (our tracks).

Given an adequate directed acyclic graph, the simplest linear time, linear space shortest path algorithm is Bellman's algorithm [1]. It can output the best possible gene structure[1]. We use a slightly more sophisticated algorithm that can take into account constraints on the minimum length of some gene elements (introns, single exon genes, intergenic regions). The algorithm allows to specify different minimum lengths for intergenic regions that separate convergent ($\rightarrow\leftarrow$), divergent ($\leftarrow\rightarrow$) or confluent ($\rightarrow\rightarrow$ or $\leftarrow\leftarrow$) genes. The algorithm is still in $O(n)$ both in time and space. Compared to existing published gene finding algorithms, this extended version of Bellman's algorithm is closely related to the algorithm presented in [15] with the additional property that minimum length constraints are coped with in linear time whereas [15]'s algorithm becomes quadratic in this case. It can also be compared to eg. the recent exons assembly algorithm proposed in [3] which is linear in the number of potential exons i.e., quadratic in the length of the sequence.

The first version of our prototype, called **EuGène** I, integrates the following sources of information:

- output of five interpolated Markov models (IMM, [12]) for respectively frame 1, 2, 3 exons, introns and intergenic sequences. These models have been estimated on the AraClean v1.1 dataset [10]. Given the sequence, the IMMs allow to compute the probability $P_t(N_i)$ that the nucleotide N_i at position i appears on each track t. The corresponding edge is weighted $-\log(P_t(N_i))$ (see Figure 1).
  A difference with the usual HMM approach is that the Markov models used to estimate the probabilities $P_t(N_i)$ are used in such a way that the graph G that represents a sequence and the graph $\bar{G}$ that represents its reverse complementary are equivalent up to a half-turn of the graph. This guarantees that any sequence will be analyzed exactly has its reverse complementary which seems desirable. To do this, the Markov models on the forward strand are estimated supposing that the Markov property holds in one direction. The Markov model on the reverse strand assume that the Markov property holds in the other direction. The same matrix can therefore be used on both forward and reverse strand (after a reverse complementation) which reduces the number of coding models from 6 to 3.
- output of existing signal prediction software. These software typically output a so-called "confidence" $0 \leq c_i \leq 1$ on the fact that a possible signal occurring at position i is used (i.e., the corresponding switch used). This confidence cannot decently be interpreted directly as a probability. We make the

[1] Linear space is needed to recover the shortest paths found.

assumption that the switch's weights have the parametric form $-\log(a.c_i^b)$ where the constants a and b have to be estimated for each source of evidence (see Figure 1).

To estimate these parameters, we have simply maximized the percentage of correct predictions on the same learning set (Araclean 1.1) using a simple genetic algorithm and/or a simple sampling of the parameter space (which has a reasonable number of dimensions) . A second version, called **EuGène** II can use, in conjunction with these basic information, results from cDNA and protein databases search:

- cDNA alignments in conjunction with splice sites are used to modify the graph as follows: matches (resp. gaps) delete intronic (resp. exonic) tracks in the graph. This forbids paths that would be incompatible with the cDNA data. Actually, cDNA/EST hits are found using Sim4 [5].
- similarly, **EuGène** II can exploit protein matches (found using BlastX). However, one cannot be confident enough in such information to directly use matches/gaps as constraints and we therefore simply modify the $P_t(N_i)$ using a simple pseudo-count scheme.

In practice, the structure and weights of the graph can be directly modified by the user using a very simple language that allows to include information about starts, splice sites, exonic, intronic, intergenic tracks on a per nucleotide basis.

2 EuGène in action

In this section we show how **EuGène** works in practice by applying it to the contig 38 from the Araset dataset [8] which contains two genes with respectively 3 and 13 exons. We first collect information about splice sites and ATG by submitting the sequence to NetGene2 [14], SplicePredictor [2], NetStart [9]... using a dedicated Perl script. This automatically builds a file containing positions and strengths of "switches" in the graph. This file, and the Perl script can be simply modified by the user to include other sources of evidence if desired. For example, the file contains sentences like "**start r4 vrai 3.7e-02 nocheck**" which states that there is a reverse start at position 4 and the weight of the corresponding edge should be $-\log(0.037)$. The "**nocheck**" indicates that the user does not want **EuGène** to verify at the end that this ATG has been effectively used in the prediction.

We then start **EuGène** and ask for a graphical zoom from nucleotide 3001 to 7000 (region of the second gene according to Araset's annotations). On this sequence, **EuGène** I locates all exons/introns border of the 2 genes. **EuGène** outputs images in PNG or GIF format which can directly be used on Web pages. The X-axis is the sequence. The Y-axis represents successive "tracks": reverse introns, frame -3, -2, -1 exons, intergenic sequences (this includes UTR), frame 1, 2, 3 exons and forward introns.

On each track, the black curve represents the output of the IMM models smoothed over a window of 100 nucleotides and normalized. The large horizontal

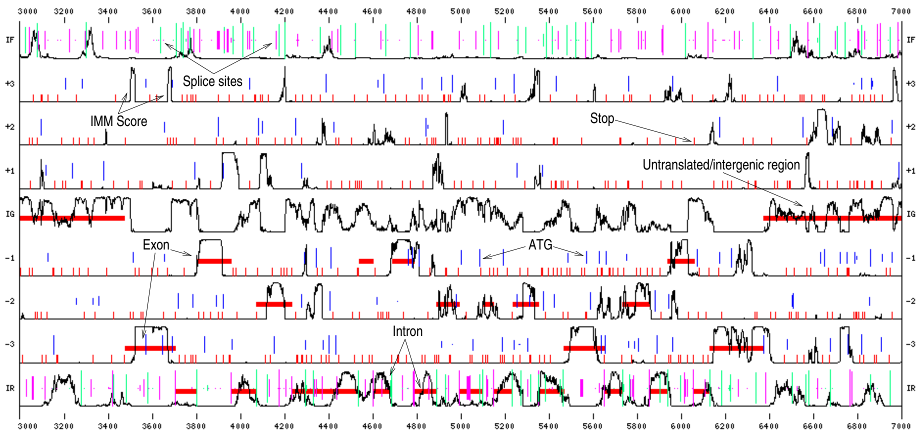

Fig. 2. **EuGène** I applied to contig 38 of Araset

blocks represent **EuGène**'s prediction. On the exonic tracks alone, small vertical bars represent potential stops and potential starts (ATG), the height of the bar being representative of the quality of the ATG according to the available evidence. On the intronic tracks, the vertical bars represent donors/acceptors. Again, the height of the bar is representative of the quality of the splice site.

EuGène I is not always as successful, eg. on the sequence of a tRNA synthetase, where **EuGène** I chooses one wrong splice site for one exon border. However, enough cDNA data exists so that **EuGène** II is able to unambiguously locate all exons/introns borders. On the graphic below, this additional information is provided: the intronic tracks show blocks that represent cDNA matches interconnected by thin lines that represent gaps (connecting splice sites). In this case, no protein database information is used.

3 Evaluation

The parameters used in **EuGène** have been estimated on Araclean [10]. Araclean contains 144 genes with very few context around the genes. This lack of context makes it almost impossible for **EuGène** to decide whether an intergenic or intronic mode should be used on the sequences border. Anyway, **EuGène** I correctly identifies all the exons (including the ATG) of 76% of the sequences of Araclean. Taking into account the lack of context, if we inform **EuGène** I that all predictions must start and end in intergenic mode (i.e., no partial gene is allowed) then the percentage of perfectly recognized genes gets up to more than to 82% (without using any cDNA/EST/protein homology information).

However this test is not very satisfactory. To get a better idea of EuGene performances, we have used another recent dataset. The “Araset” dataset has originally been designed and used to assess several existing gene or signal finding software [8]. The clear winner of this evaluation is GeneMark.hmm [6]. We

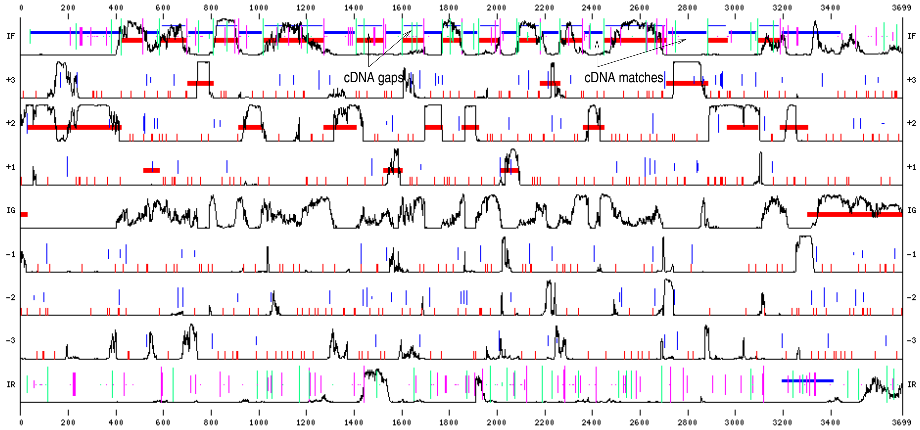

Fig. 3. EuGène II applied to a tRNA synthetase with cDNA data

therefore decided to compare **EuGène** to GeneMark.hmm on this dataset (which does not share any sequence with the Araclean dataset used for **EuGène** parameter estimation). The evaluation performed in [8] includes different performance measures, among which we will consider:

- sites prediction quality: for splice sites, the number of predicted, correct, overpredicted and missing sites has been measured.
- exon prediction quality: for all exons, the number of predicted, correct, overlapping, wrong and missing exons has been measured.
- gene level prediction quality: in this case, a gene is considered as correctly predicted if all its exons are correctly predicted. Il all exons are not correctly predicted, the gene can be missing (all exons are missing), partial (some exons are missing or partial), wrong (the predicted gene does not exist – as far as we know), split (a correct gene has been split in two or more predicted genes), fused (correct adjacent genes have been predicted as one gene).
- gene borders prediction: for predicted translation initiation sites (`AUG`) and terminator sites (stops), the number of correct prediction has been measured. For `AUG`, we also give the number of incorrect predictions which are nevertheless in phase with the real `AUG` site.

In each case, the overall performance is abstracted in the two usual *specificity* (S_p) and *sensitivity* (S_n) measures introduced in [13].

$S_n =$ number of correctly predicted items / number of actual items
$S_p =$ number of correctly predicted items / number of predicted items

For a more detailed definition of the measures, we invite the reader to refer to [8, 13]. Note also that the biological truth, considered here as unambiguously provided by Araset's annotations, is not so simple: alternative splicing or translation initiation exists and make these numbers less reliable than one may think *a priori*.

Table 1 presents the results[2] obtained for splice sites. **EuGène** I is less sensitive but more specific than GeneMark.HMM. Table 2 gives similar numbers for translation initiation (`AUG`) and terminator (stop codons) sites. Here, the results are clearly in favor of **EuGène**.

For the gene model level we also evaluated **EuGène** II. We used SPTR as the protein database. We also have a cDNA/EST database built using EMBL, and cleaned from documented partially or alternatively spliced cDNA. However, this database is built entirely automatically and is known to contain data inconsistent with Araset annotations for some of the genes of Araset (these inconsistencies are the results of contamination with genomic DNA and undocumented alternatively or partially spliced cDNA). Although using this cDNA/EST information alone improves the performances of **EuGène**, the current version of **EuGène** II tested uses only protein informations from SPTR because of its reliability. A clean cDNA/EST database would improve **EuGène** II's performances.

Although an in-depth analysis is still needed to be more conclusive, it appears from the measures above that the essential strength of **EuGène** lies in its ability to correctly identify gene's borders: compared to GeneMark.HMM, **EuGène** merges or split very few genes and identifies most translation initiation sites or terminator sites correctly. We think that this improvement is caused by two original elements of **EuGène**: one is the use of a dedicated intergenic Markov model, which is not a background 0^{th} order model and is different from the intronic model; the other lies in the fact that, except for the IMM, its parameters have been estimated by maximum of "successful recognition" rather than maximum likelihood. This probably tends to compensate for the weaknesses of the model. Initially, we thought that the good quality of the splice sites predictors used by **EuGène** (NetGene2 and Splice Predictor) was a major reason for its good performances. Looking to the splice sites prediction ability of **EuGène**, this is less obvious now although **EuGène** clearly inherits the excellent specificity of NetGene2.

This report is still quite preliminary and we expect to enhance **EuGène**'s effectiveness in a near future (and apply it to other organisms). Actually, compared to other gene finding algorithms, **EuGène** is relatively simple: it uses a single Markov model set independently of GC% (although this is probably more important for the human genome than for *Arabidopsis thaliana*), does not take into account signals such as polyA or promoters or does not model UTR. This should leave room for improvements. Another important direction lies in the use of protein and cDNA/EST hits. We think that the integration, at the lowest level, of spliced alignment algorithms in the spirit of [5, 4] could help both to better take into account similarity information and also to filter out most inconsistent data. Other sources of similarity such as conservation of exons between (orphan) genes of the same family should also be integrated (M-F. Sagot, personal communication). We are currently working on these points.

[2] Reference [8] gives a number of introns (and therefore acceptors and donors) of 860. However, parsing the reference file distributed with the instances, we only found 859 introns.

Program	predicted	correct	overpredicted	missing	S_n	S_p	predicted	correct	overpredicted	missing	S_n	S_p
GeneMark	981	797	184	63	0.93	0.81	914	772	142	88	0.90	0.84
EuGène I	798	742	56	117	0.86	0.93	801	733	68	126	0.85	0.92

Table 1. Donor and acceptor sites prediction (859 sites each)

Program	predicted	correct	in phase	S_n	S_p	predicted	correct	S_n	S_p
GeneMark	196	129	31	0.77	0.66	173	136	0.81	0.79
EuGène I	190	143	18	0.85	0.75	193	155	0.92	0.80

Table 2. Translation initiation and terminator sites prediction (168 sites each)

Program	predicted	correct	overlap	wrong	missing	S_n	S_p	ratio wrong	splitted	fused
GeneMark	1104	845	172	87	26	0.82	0.77	0.08	10	4
EuGène I	991	837	102	57	89	0.82	0.84	0.06	1	5

Table 3. Exons prediction

Program	actual	predicted	correct	missing	partial	wrong	split	fused	S_n	S_p
GeneMark	168	208	67	1	100	27	18	12	40%	32 %
EuGène I	168	194	104	1	63	19	8	2	62%	54 %
EuGène II	168	198	125	1	42	22	9	0	74%	63 %

Table 4. Gene level measures

Acknowledgements : this work has been supported by a grant from the « Programme prioritaire génome & fonction » from the *Institut national de la recherche agronomique* (INRA). We would also like to thank N. Pavie and C. Mathé (INRA, Gand) for their help with Araset, S. Rombauts (INRA, Gand), B. Lescure (INRA, Toulouse), H. Chiapello and I. Small (INRA, Versailles) for sharing their experience de l'arabette, P. Dehais and S. Aubourg (INRA, Gand) for their feedback on EuGène.

References

[1] R. Bellman, *Dynamic Programming*, Princeton Univ. Press, Princeton, New Jersey, (1957).

[2] V. Brendel and J. Kleffe, (1998), *Prediction of locally optimal splice sites in plant pre-mRNA with application to gene identification in Arabidopsis thaliana genomic DNA*, Nucleic Acids Res., 26, pp. 4749–4757.

[3] M. Burset and R. Guigo, (1996), *Evaluation of gene structure prediction programs*, Genomics, 34, pp. 353–367.

[4] U. J., Z. W., and B. V., (2000), *Optimal spliced alignment of homologous cDNA to a genomic DNA template*, Bioinformatics, 16, pp. 203–211.

[5] F. L., H. G., Z. Z., R. G.M., and M. W., Sept. 1998, *A computer program for aligning a cDNA sequence with a genomic DNA sequence*, Genome Res, 8, pp. 967–974.

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

[7] K. Murakami and T. Takagi, (1998), *Gene recognition by combination of several gene-finding programs*, BioInformatics, 14, pp. 665–675.

[8] N. Pavy, S. Rombauts, P. Déhais, C. Mathé, D. Ramana, P. Leroy, and P. Rouzé, (1999), *Evaluation of gene prediction software using a genomic dataset: application to Arabidopsis thaliana sequences*, in Proc. of 2^d Georgia Tech conference on BioInformatics, Atlanta.

[9] A. Pedersen and H. Nielsen, (1997), *Neural network prediction of translation initiation sites in eukaryotes: prespectives for EST and genome analysis*, in Proc. of ISMB'97, AAAI Press, pp. 226–233.

[10] P. R. P.G. Korning, S.M. Hebsgaard and S. Brunak, (1996), *Cleaning the genbank arabidopsis thaliana data set*, Nucleic Acids Res., 24, pp. 316–320.

[11] L. Rabiner, (1989), *A tutorial on hidden markov models and selected application in speech recognition*, Proc. IEEE, 77, pp. 257–286.

[12] S. L. Salzberg, A. L. Delcher, S. Kasif, and O. White, (1998), *Microbial gene identification using interpolated Markov models*, Nucleic Acids Res., 26, pp. 544–548.

[13] E. Snyder and G. Stormo, *DNA and protein sequence analysis: a practical approach*, IRL Press, Oxford, (1995), ch. Identifying genes in genomic DNA sequences, pp. 209–224.

[14] N. Tolstrup et al., (1997), *A branch-point consensus from Arabidopsis found by non circular analysis allows for better prediction of acceptor sites*, Nucleic Acids Res., 25, pp. 3159–3163.

[15] T. D. Wu, (1996), *A segment-based dynamic programming algorithm for predicting gene structure*, Journal of Computational Biology, 3, pp. 375–394.