

# Support de cours

## Annotation des génomes prédiction des CDS (Partie II)

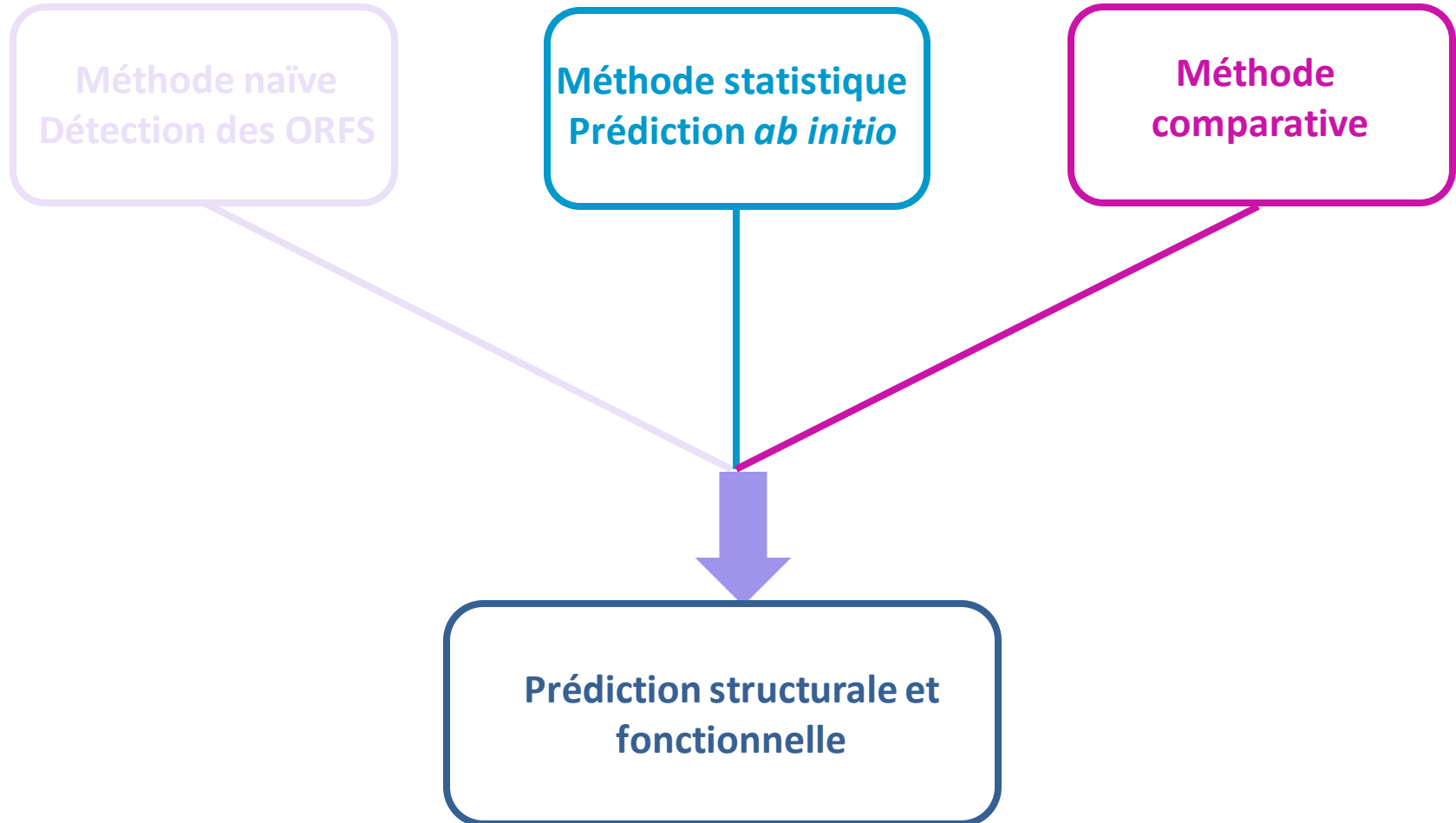
## Recherche des régions codant pour des protéines chez les procaryotes

- recherche des ORFs (Open reading frame)
- recherche des unités de traduction. Même si les gènes sont co-transcrits, ils sont en général traduits de façon indépendante (recherche des Shine Dalgarno en 5' du codon initiateur). Permet d'identifier le « bon » codon initiateur.
- recherche des unités de transcription. Chez les procaryotes, certains gènes sont co-transcrits donc recherche de la structure en opérons (promoteurs et terminateurs de transcription)

## Recherche des régions codant pour des protéines chez les eucaryotes

- recherche de la structure en exon/intron du gène
- recherche des 5'UTR et 3' UTR
- recherche des promoteurs et des sites de polyadénylation

# Recherche des régions codant pour des protéines chez les procaryotes



# Une méthode naïve : ORFfinder (NCBI)

Recherche les phases ouvertes de lecture, les ORFs, dans les 6 cadres de lecture (les 3 cadres du brin direct et les 3 cadres du brin complémentaire).

Attention problème de sémantique :

Une ORF peut être définie entre deux codons stop

stop XXXXXXXXXXXXXXXXXXXXXXXXXXXX stop  
n codons

Ou entre un codon initiateur (start) et un codon stop

ATG XXXXXXXXXXXXXXXXXXXXXXXXXXXX stop  
n codons

On considère en général que les ORFs supérieures à 100 codons (300 pb) comme étant potentiellement codantes (analyse statistique a montré que bien que des gènes de taille inférieure à 100 codons existent, la majorité des petites ORFs étaient des faux positifs).

>BS 1-8301

tttcgaggaaaatgtgcaataaccaactcatttcccgggcaattccgccg  
gttccgaatgatacgaacaactgagactgagccgcaaattgggttcagtctt  
tttacatggcagccagagggctttgtgcaacttgacatttgtgaaaaagaa  
agtaaaatattttactaaaacaatgcgagctgaataatggaggcagatac  
aatggcgacaattaaagatatcgcgaggaagcgggattttcaatctcaa  
ccgtttcccgcggttttaataacgatgaaagcctttctgttcctgatgag  
acacgggagaaaatctatgaagcggcggaaaagctcaattaccgcaaaaa  
aacagtaaggccgctgggtgaaacataattgcggtttttatattggctgacag  
ataaagaagaattagaagatgtctattttaaaacgatgagattagaagta  
gagaaactggcgaaagcattcaatgtcgatatgaccactataaaatagc  
ggatggaaatcgagagcattcctgaacatacgggaagggtttattgcccgtcg  
gcacattttcagatgaagagctggctttcctcagaaatctcactgaaaac  
ggcgtgttcacgatcaactcctgatcccgatcattttgactcggtaag  
gcccgatattggcacaatgacaaggaagacggtaaacatcctgactgaga  
aggggcataagagcatcgggttttatcggcggcacatacaaaaatccgaat  
accaatcaggatgaaatggacatccgtgaacaaaccttcagatcctatat  
gagggaaaaagccatgctggacgagcgtatattttctgtcatcgcggat  
tctctgtagaaaacggctaccgcctgatgtcagcagcagatcgacacatta  
ggcgatcagcttccgactgcttttatgattgcagcggaccgattgcagt  
gggctgtctgcaagccctgaacgaaaaaggaattgccataccaaacaggg  
taagcattgtgagtatcaacaacatcagcttcgcgaagtatgtctcgcct  
cctctgacgacgtttcatattgatatacatgaattatgtaaaaacgctgt  
tcaattactgcttgaacaagtgcaggacaagagaagaacggtaaaaacat  
tataatgtgggcgagaattaatcgtcaggaagagatgaattaaggatga  
cttaggacactaagtcattttttatttaggtaaaaaaatttactctatga  
agtaaatagtttgtttacacattttctcaggcatgctatattatctttaa  
agcgctttcattcctaccgaaagggtgacaatcaatgaaaatggcaaaaa  
agtgttccgtattcatgctctgcgcagctgtcagtttatccttggcggct  
tgcggcccaaaggaaagcagcagcgcgcaaatcgagttcaaaagggtcaga  
gcttgttgtatgggaggataaagaaaagagcaacggcattaagacgctg  
tggctgcatttgaaaaagagcatgatgtgaaggtcaaagtcgttgaaaa  
ccgtatgccaaagcagattgaagatttgcgaatggatggaccggccggcac  
aggccctgacgtgttaacaatgccaggggaccaaactcggaaccgctgtca  
cggaaaggattactcaaggaattacatgtcaaaaaagacgttcaatcactt  
tatactgacgcttccattcagctctcaaatggtagatcaaaagctttatgg  
actgccaaaagcggctcgaaacgactgtgcttttttacaacaaagatctca  
tcacagaaaaggaattgcccaaacgctggaagagtggtagcactattcc

## Exemple traité : fragment de 8300 pb du génome de *Bacillus subtilis*



## Open Reading Frame Finder

ORF finder searches for open reading frames (ORFs) in the DNA sequence you enter. The program returns the range of each ORF, along with its protein translation. Use ORF finder to search newly sequenced DNA for potential protein encoding segments, verify predicted protein using newly developed SMART BLAST or regular BLASTP.

This web version of the ORF finder is limited to the subrange of the query sequence up to 50 kb long. Stand-alone version, which doesn't have query sequence length limitation, is available for [Linux x64](#).

**Examples** (click to set values, then click Submit button) :

- NC\_011604 Salmonella enterica plasmid pWES-1; genetic code: 11; 'ATG' and alternative initiation codons; minimal ORF length: 300 nt
- NM\_000059; genetic code: 1; start codon: 'ATG only'; minimal ORF length: 150 nt

### Enter Query Sequence

Enter accession number, gi, or nucleotide sequence in FASTA format:

```
>BS 1-8301
tttcgaggaaaatgtgcaataaccaactcatttccgggcaattccgcccgttcggaatg
atcgaacaactgagactgagcgcgcaaatggttcagtccttttacatggcagccagaggg
ctttgtgcaacttgacatttggaaaaagaaagtaaaatatttactaaaacaatgcgagc
tgaataatggaggcagatacaatggcgacaattaaagatatcgcgaggaagcgggattt
tcaatctcaaccgtttcccgcttttaataacgatgaaagcctttctgttctgatgag
acacgggagaaaatctatgaagcggcggaaaagctcaattaccgcaaaaaaacagtaagg
ccgtggtgaaacatatgctgttttatattggctgacagataaagaagattagaagat
gtctattttaaaccgatgagattagaagttagagaaactggcgaagcattcaatgtcgat
atgaccacttataaaatagcggatggaatcgagagcattcctgaacatacggagggttt
attgccgtcggcacatttcagatgaagagctggcttctcagaaatctcactgaaaac
```

From:  To:

### Choose Search Parameters

Minimal ORF length (nt):

Genetic code:

ORF start codon to use:

- "ATG" only
- "ATG" and alternative initiation codons
- Any sense codon

Ignore nested ORFs:

### Start Search / Clear

Submit

Clear



# Résultat de ORFfinder : ORFs de plus de 300 pb

Options : - ATG only  
- Ignore nested ORF pas coché

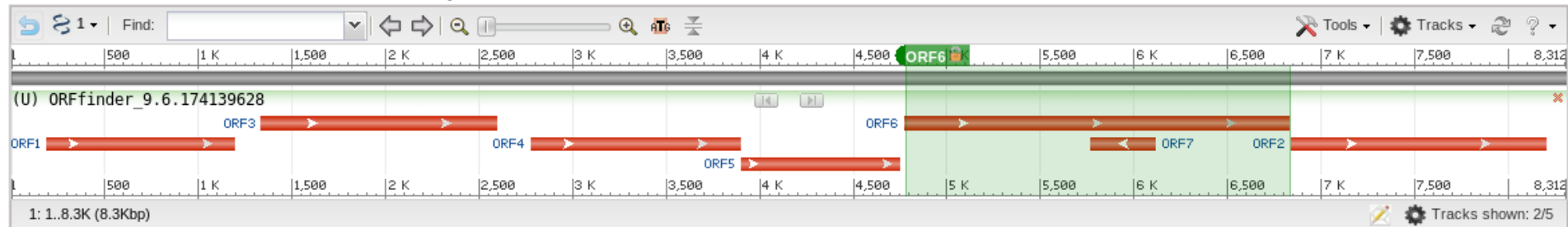


Open Reading Frame Viewer

Help

Sequence

ORFs found: 7 Genetic code: 1 Start codon: 'ATG' only



Six-frame translation...

ORF6 (686 aa)

Display ORF as...

Mark

Mark subset...

Marked: 0

Download marked set

as

Protein FASTA

```
>lcl|ORF6
MSKLEKTHVTKAKFMLHGGDYNPDQWLDRPDILADDIKMLKLSHTNTFSV
GIFAWSALEPEEGVYQFEWLDDIFERIHSIGGRVILATPSGARPAWLSQT
YPEVLRVNASRVKQLHGGRHNHCLTSKVYREKTRHINRLLAERYGHPAL
LMWHISNEYGGDCHCDLQHAFAREWLSKYDNSLKTLMHAWWTPFWSHTF
NDWSQIESPSPIGENGLHGLNLDWRRFVTDQTSIFYENEIIPLKELTPDI
PITTFMADTPDLIPYQGLDYSKFAKHVDAISWDAYPVWHNDWESTADLA
MKVGFINDLYRSLKQPPFLMECTPSAVNWHNVNKAARPGMNLSSMQMI
AHGSDSVLYFQYRKSRSSEKLGAVVDHNSPKNRVFEVAKVGETLER
LSEVVGTKRPAQTAILYDWHENHWALEDAQGFATKRYPTLQQHYRTFW
EHDIPVDVITKEQDFSPYKLLIVPMLYLISEDVSRKKAFTADGGTLVMT
YISGVVNEHDLTYTGGWHPDLQAFVGEPLETDTLYPKDRNAVSYRSQIY
EMKDYATVIDVKTASVEAVYQEDFYARTPAVTSHEYQQGKAYFIGARLED
QFQDFYEGRLITDLSLSPVPVRRHGKGVSVQARQDQNDYIFVMNFTEEK
QLVTFDQSVKDIINTGDILSGDLTMEKYEVRIVVNTH
```

Label	Strand	Frame	Start	Stop	Length (nt   aa)
ORF6	+	3	4773	6833	2061   686
ORF2	+	1	6838	8202	1365   454
ORF3	+	3	1335	2600	1266   421
ORF4	+	3	2778	3896	1119   372
ORF1	+	1	187	1194	1008   335
ORF5	+	3	3900	4751	852   283
ORF7	-	3	6117	5770	348   115

ORF6

Marked set (0)

SmartBLAST

SmartBLAST best hit titles...

BLAST

BLAST

# Résultat de ORFfinder : ORFs de plus de 300 pb

Options : - ATG and alternative initiation codons  
 - Ignore nested ORF coché

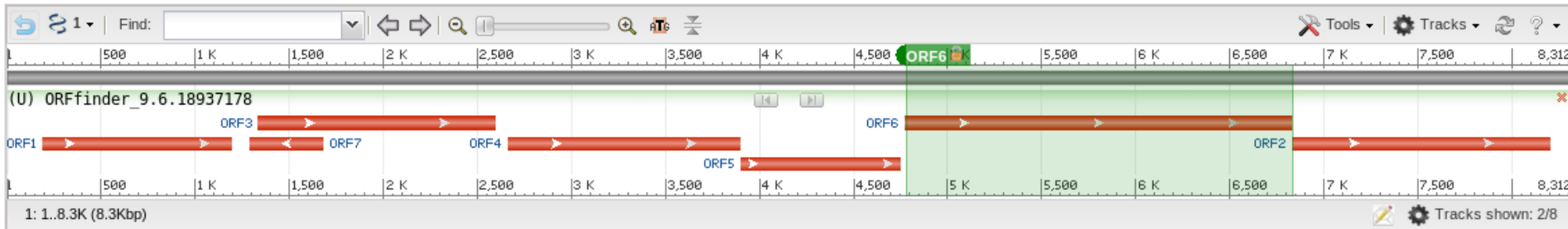


Open Reading Frame Viewer

Help

Sequence

ORFs found: 7 Genetic code: 1 Start codon: 'ATG' and alternative codons Nested ORFs removed



Six-frame translation...

ORF6 (686 aa)

Display ORF as...

Mark

Mark subset...

Marked: 0

Download marked set

as Protein FASTA

```
>lcl|ORF6
MSKLEKTHVTKAKFMLHGGDYNPDQWLDLRDPDILADDIKMLKLSHTNTFSV
GIFAWSALEPEEGVYQFEWDDIFERIHSIGGRVILATPSGARPAWLSQT
YPEVLRVNASRVKQLHGGRHNHCLTSKVYREKTRHINRLLAERYGHPAL
LMWHISNEYGGDCHCDLQAHAFREWLKSKYDNLKTLNHAWWTPFWSHTF
NDWSQIESPSPIGENGLHGLNLDWRRFVTDQTIISFYENEIIPLKELTPDI
PITTFMADTPDLIPYQGLDYSKFAKHVDAISWDAYPVWHNDWESTADLA
MKVGFINDLYRSLKQPFLLMECTPSAVNWHNVNNAKRPGMNLSSMQMI
AHGSDSVLYFQYRKSRSSEKLGAVVDHNSPKNRVQEVAKVGETLER
LSEVVGTRPAQTALYDWHENHWALEDAOGFAKATKRYPTLQOHYRTFW
EHDIPVDVITKEQDFSPYKLLIVPMLYLISETVSRKAKFTADGGTLVMT
YISGVVNEHDLTYTGGWHPDLQAIQVGEPLETDTLYPKDRNAVSYRSQIY
EMKDYATVIDVKTASVEAVYQEDFYARTPAVTSHEYQQGKAYFIGARLED
QFQRDFYEGILTDLSSLSPVFPVRHGGKGVSVQARQQDNDYIFVMNFTEEK
QLVTFDQSVKDIMTGDILSGDLTMEKYEVRIVVNTH
```

Label	Strand	Frame	Start	Stop	Length (nt   aa)
ORF6	+	3	4773	6833	2061   686
ORF2	+	1	6835	8202	1368   455
ORF3	+	3	1335	2600	1266   421
ORF4	+	3	2661	3896	1236   411
ORF1	+	1	187	1194	1008   335
ORF5	+	3	3900	4751	852   283
ORF7	-	2	1681	1286	396   131

ATG only : début 6838

ATG only : début 2778

Pas trouvé ATG only

ORF6

Marked set ( 0 )

SmartBLAST

SmartBLAST best hit titles...

BLAST

BLAST



# Résultat de ORFfinder : ORFs de plus de 150 pb

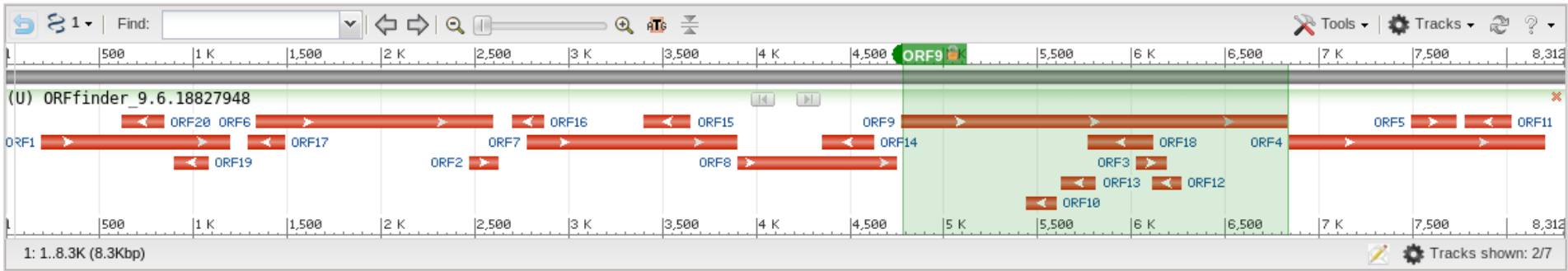
Options : - ATG only  
 - Ignore nested ORF pas coché

## Open Reading Frame Viewer

Help

### Sequence

ORFs found: 20 Genetic code: 1 Start codon: 'ATG' only



Six-frame translation...

ORF9 (686 aa) [Display ORF as...](#) [Mark](#) [Mark subset...](#) Marked: 0 [Download marked set](#) as [Protein FASTA](#)

```
>lcl|ORF9
MSKLEKTHVTKAKFMLHGGDYNPDQWLDRPDILADDIKLMKLSHTNTFSV
GIFAWSALEPEEGVYQFEWLDDIFERIHSIGGRVILATPSGARPAWLSQT
YPEVLRVNASRVKQLHGGRRNHCLTSKVYREKTRHINRLLAERYGHHPAL
LMWHISNEYGGDCHCDLCOHAFREWLKSKYDNSLKTLNHAWWTPFWSHTF
NDWSQIESPPIGENGLHGLNLDWRRFVTDQTSFYENEIIPKELTPDI
PITTNFMADTPDLIPYQGLDYSKFAKHVDAISWDAYPVVHNDWESTADLA
MKVGFINDLYRSLKQOPFLMECTPSAVNWHNVNKAARPGMNLSSMQMI
AHGSDSVLYFQYRKSRSSEKLGAVVDHNSPKNRVFEVAKVGETLER
LSEVVGTKRPAQTAILYDWHENHWALEDAQGFAKATKRYPQTLQQHYRTFW
EHDIPVDVITKEQDFSPYKLLIVPMLYLISEDVSRKFAFTADGGTLVMT
YISGVVNEHDLTYTGGWHPDLQAIIFGVEPLETDTLYPKDRNAVSYRSQIY
EMKYATVIDVKTASVEAVYQEDFYARTPAVTSHEYQQGKAYFIGARLED
QFQDFYEGLEITDLSLSPVFPVRHGGKGSVQARQDQNDYIFVMNFTEEK
QLVTFDQSVKDIMTGDILSGDLTMEKYEVRIVVNTH
```

Label	Strand	Frame	Start	Stop	Length (nt   aa)
ORF9	+	3	4773	6833	2061   686
ORF4	+	1	6838	8202	1365   454
ORF6	+	3	1335	2600	1266   421
ORF7	+	3	2778	3896	1119   372
ORF1	+	1	187	1194	1008   335
ORF8	+	3	3900	4751	852   283
ORF18	-	3	6117	5770	348   115
ORF14	-	2	4627	4349	279   92
ORF15	-	2	3652	3401	252   83
ORF11	-	2	8026	7775	252   83

ORF9

Marked set (0)

SmartBLAST

SmartBLAST best hit titles... ?

BLAST

BLAST

# Limites d'ORFfinder

- Seul critère la taille, mais tous les ORF ne sont pas des régions codantes (Coding Sequence : CDS) quand ils sont de petites tailles.
- Problème d'identification du « vrai » codon initiateur car plusieurs possibilités en fonction du choix « ATG » ou « codon initiateur alternatif », GTG et TTG chez les bactéries avec comme fréquence 13% et 9% chez *Bacillus subtilis* par exemple.
- ne prend pas en compte le biais de l'utilisation des triplets existant dans les phases codantes car structurées en codons.



Utilisation de méthode statistique

Prise en compte du biais de l'utilisation des triplets existant dans les phases codantes par rapport aux régions non codantes car structurées en codons.

Biais dans l'utilisation des codons dus à :

- la différence de fréquence des acides aminés (Leu plus fréquent que Trp par exemple)
- la dégénérescence du code génétique (61 codons -> 20 aa)
- pour un acide aminé donné, certains codons peuvent être plus fréquemment utilisés que d'autres. Ces préférences varient en fonction :
  - la composition en bases de l'organisme (riche ou pauvre en C+G) : **usage du code différent**
  - du taux d'expression du gène : il a été montré chez *E. coli* que les gènes fortement exprimés utilisaient préférentiellement certains codons correspondant aux ARNt les plus abondants dans la cellule (efficacité de la traduction, coadaptation codons/ARNt. (Calcul du **Codon Adaptation Index (CAI)** pour prédire gène fortement exprimé ou pas)

# Exemples d'usage des codons chez les bactéries

Espèce	GC% codant	GC% 1 <sup>ère</sup> pos. codon	GC% 2 <sup>ème</sup> pos. codon	GC% 3 <sup>ème</sup> pos. codon
<i>Synechocystis sp.</i>	48.25	55.82	39.74	49.19
<i>Streptomyces coelicolor</i>	72.30	72.67	51.39	92.83
<i>Escherichia coli</i> 0157:H7	51.54	58.44	41	55.17
<i>Bacillus subtilis</i>	44.36	52.10	36.08	44.91

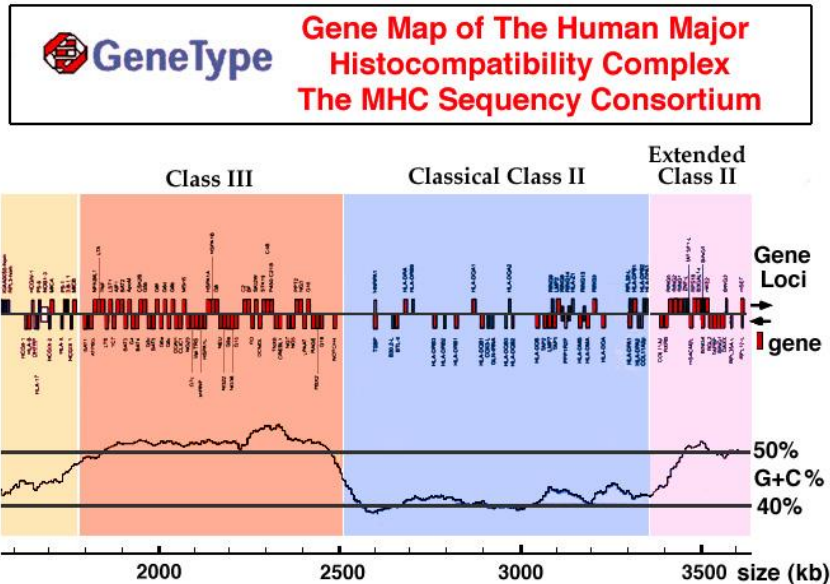
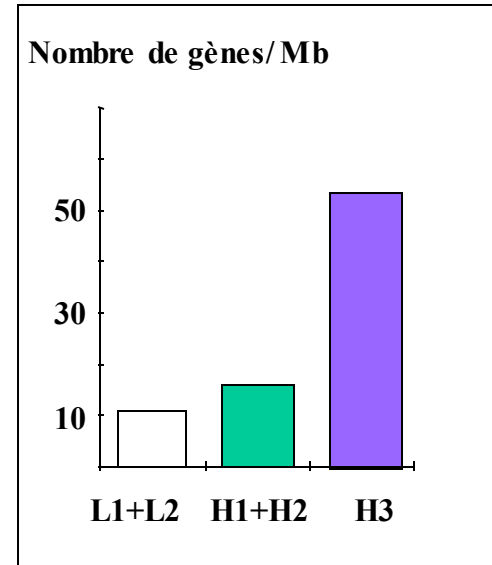
A.A.	codon	% S. sp. (cyano)	% S. coelicolor	% E. coli	% B. subtilis
Gly	GGG	0.24	0.19	0.164	0.16
Gly	GGA	0.18	0.075	0.123	0.315
Gly	GGT	0.27	0.096	0.331	0.187
Gly	GGC	0.31	0.64	0.382	0.337
Glu	GAG	0.264	0.846	0.325	0.32
Glu	GAA	0.736	0.154	0.675	0.68
Asp	GAT	0.646	0.05	0.631	0.636
Asp	GAC	0.354	0.95	0.369	0.364

# Modèle de la structure en isochores chez les vertébrés

Isochores : régions > 300 kb homogène dans sa composition en bases

5 types d'isochores en fonction de leur pourcentage en G+C (2 légers et 3 lourds)

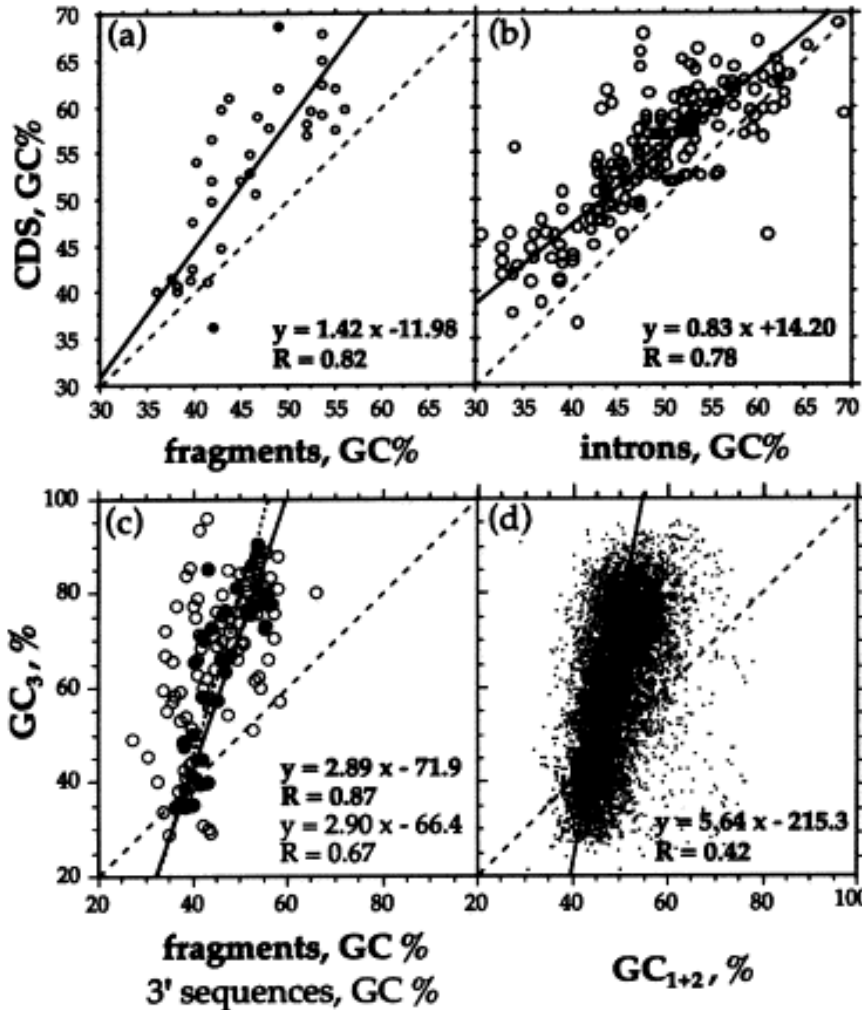
Isochore	%C+G	% total genomic DNA
L1+L2	33%-44%	62 %
H1+H2	44%-51%	31%
H3	51%-60%	3-5%



MHC locus (3.6 Mb) (The MHC sequencing consortium 99)

- Class I, class II (H1-H2 isochores): 20 gènes/Mb, beaucoup de pseudogènes
- Class III (H3 isochore): 84 gènes/Mb, pas de pseudogène

# Modèle de la structure en isochores chez les vertébrés



Correlation between GC levels of human coding sequences and :

- (a) the GC levels of the large DNA fragments in which sequences were localized, or
- (b) the GC levels of the corresponding introns (top frames).

The bottom frames show the correlations between GC<sub>3</sub> of human coding sequences and

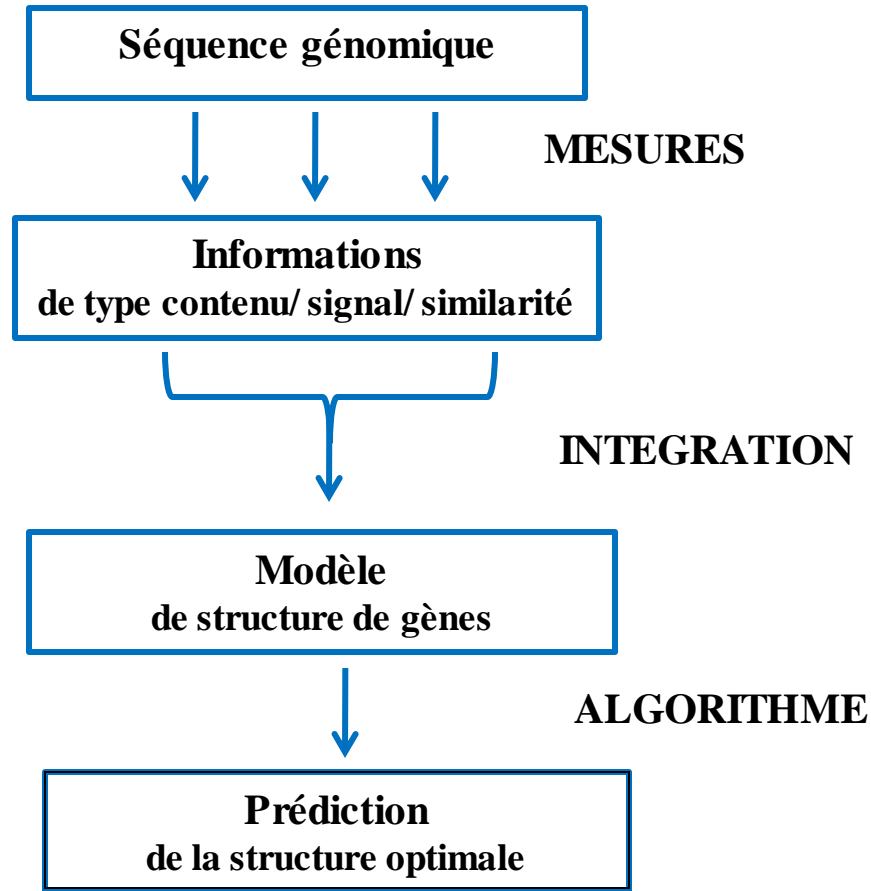
- (c) the GC levels of the DNA fractions in which the genes were localized (filled circles) and of 3' flanking sequences further than 500 bp from the stop codon (open circles; the solid and the broken lines are the regression lines through the two sets of points); or

- (d) GC<sub>1</sub>+GC<sub>2</sub> values of human sequences.

Diagonals (unity slope lines) are also shown

# Recherche des régions codant pour des protéines

## Fonctionnement schématique d'un logiciel de prédiction de gènes



# Méthode statistique

## Traitement de l'information de type contenu

Utilisation de méthodes statistiques prenant en compte ces biais d'utilisation des codons. Plus récemment avec l'augmentation des données pour établir les systèmes de référence, prise en compte de la composition en hexanucléotides (mots de longueur 6).

Les méthodes statistiques couramment utilisées :

- Modèles de Markov
- Modèles de Markov interpolés (IMM)
- Modèles de Markov caché (HMM)



Développé en 1907 par le mathématicien Andreï Markov :

- Décrit un processus permettant de calculer la probabilité d'occurrence de phénomènes aléatoires en considérant qu'ils sont liés comme les maillons d'une chaîne .
- Interdépendance des éléments

Ordre du modèle de Markov :

- Ordre 0 : les éléments sont indépendants
- Ordre 1 : la probabilité d'apparaître de l'élément  $n$  dépend (est lié) à l'élément  $n-1$
- Ordre 2 : la probabilité d'apparaître de l'élément  $n$  dépend (est lié) des éléments  $n-2$  et  $n-1$
- ....

# Modèle de Markov

Un modèle de Markov d'ordre  $k$  appliqué aux séquences ADN est entièrement défini par les deux probabilités suivantes :

$$\left[ \begin{array}{l} P_0(w_1^k) \\ P(x / w^k) \end{array} \right. \begin{array}{l} \longrightarrow \text{Probabilité initiale du mot } w^k \\ \longrightarrow \text{Probabilité d'observer } x \text{ sachant que le mot } w^k \text{ le} \\ \text{précède} \end{array}$$

Modèle probabiliste qui représente une séquence comme un processus qui peut être décrit comme une séquence de variable aléatoire  $X_1, X_2, \dots$  où  $X_i$  correspond à la position  $i$  de la séquence. Chaque variable aléatoire  $X_i$  prend une valeur dans l'ensemble des bases (A,C,G,T). La probabilité que va prendre la variable  $X_i$  dépend du contexte c'est à dire des bases immédiatement adjacentes à la base à la position  $i$ .

# Modèle de Markov : Présentation de GeneMark

(Borodovsky et al., Nucleic Acids Res.,22,4756-67)



La méthode repose sur le modèle probabiliste suivant appelé modèle de Markov:

Hypothèse 1: La probabilité d'observer une base à une position donnée dépend :

- Régions non codantes
  - ✓ des bases précédant cette position
  - ✓ modélisé par un modèle de Markov homogène
  
- Régions codantes
  - ✓ des bases précédant cette position
  - ✓ de sa localisation dans le codon
  - ✓ Modélisé par un modèle de Markov non-homogène 3-périodique (construction de 3 modèles différents un par position)

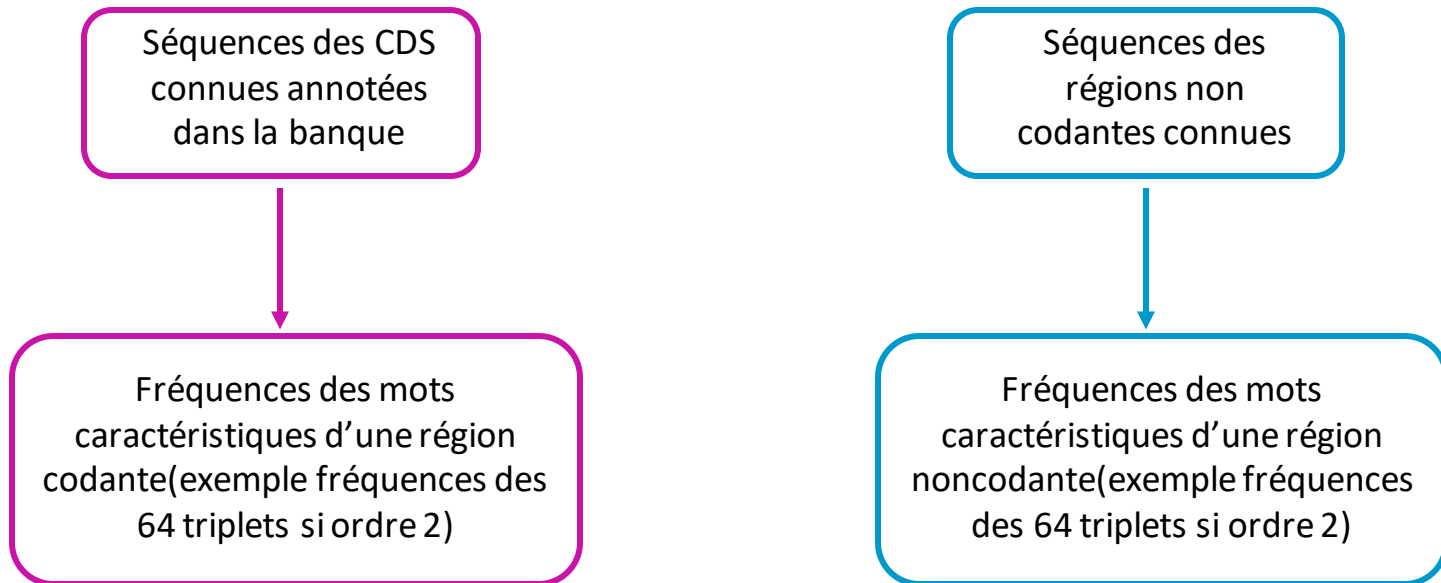
Hypothèse 2: Une région particulière ne peut être que dans un des 7 états suivants:

- 1. codant en phase 1 sur le brin direct
- 2. codant en phase 2 sur le brin direct
- 3. codant en phase 3 sur le brin direct
- 4. codant en phase 4 sur le brin indirect
- 5. codant en phase 5 sur le brin indirect
- 6. codant en phase 6 sur le brin indirect
- 7. non-codant

Prédiction : calculer les probabilités d'observer la région dans un état  $i$  sachant que l'un des 7 états est réalisé (formule de Bayes).

Nécessite d'avoir des tables de référence pour calculer la fréquence de chaque mots Annotation d'un fragment génomique issu d'une espèce bactérienne *B* :

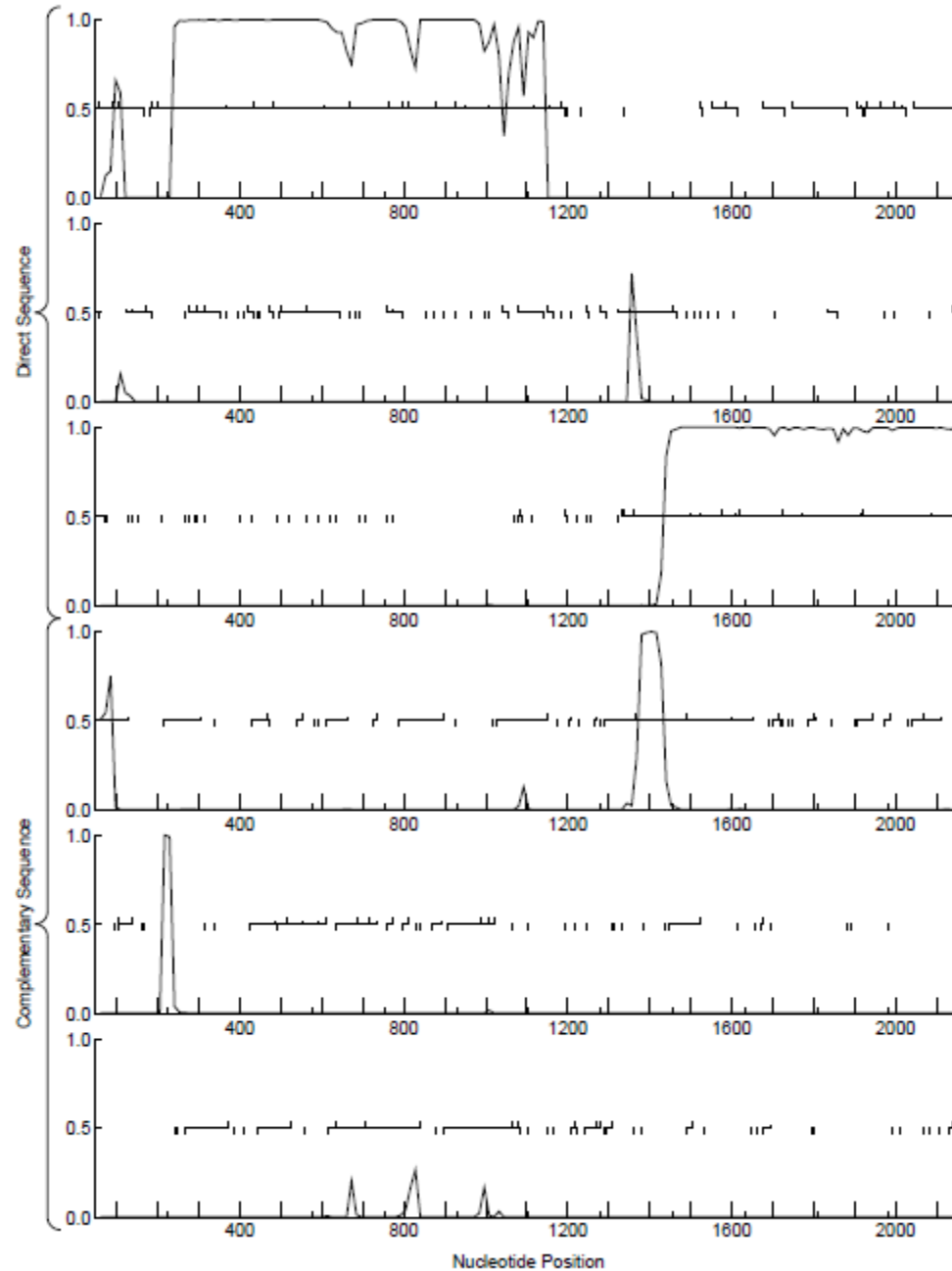
- Des données sont disponibles dans les bases de données comme GenBank ou EMBL pour l'espèce *B* ou une espèce proche dans l'évolution -> utilisation d'une référence externe
  - ✓ création de deux ensemble d'apprentissage et calcul des tables de référence



- ✓ Prédiction sur notre séquence : comparaison des mots rencontrés dans notre fragment avec chaque table : calcul de la probabilité que la portion de la séquence soit dans chacun des 7 états. Choix de l'état le plus probable
- Pas de données disponibles pour l'espèce *B* :
  - ✓ Tables obtenues en utilisant un modèle heuristique

# Résultat graphique de GeneMark sur le fragment de *B. subtilis*

EMBOSS\_001 Reversed:, Order 4, Window 96, Step 12, 2/5



# Résultat de GeneMark sur le fragment de *B. subtilis*



## Entête du fichier :

```
Sequence: EMBOSS_001 Reversed:
Sequence file: seq.fna
Sequence length: 8312
GC Content: 45.19%
Window length: 96
Window step: 12
Threshold value: 0.500
---
Matrix: Bacillus_subtilis_168
Matrix author: -
Matrix order: 4
```

## Fin du fichier :

### List of Regions of interest

(regions from stop to stop codon  
w/ a signal in between)

LEnd	REnd	Strand	Frame
181	1194	direct	fr 1
1286	1693	complement	fr 1
1326	2600	direct	fr 3
2610	3896	direct	fr 3
3894	4751	direct	fr 3
4749	6833	direct	fr 3
6820	8202	direct	fr 1

List of Open reading frames predicted as CDSs, shown with alternate starts  
(regions from start to stop codon w/ coding function >0.50)

Left end	Right end	DNA Strand	Coding Frame	Avg Prob	Start Prob	
187	1194	direct	fr 1	0.85	0.99	→ ORF Finder
202	1194	direct	fr 1	0.86	0.90	
367	1194	direct	fr 1	0.89	0.10	
436	1194	direct	fr 1	0.88	0.02	
481	1194	direct	fr 1	0.87	0.01	
1335	2600	direct	fr 3	0.87	0.04	→ ORF Finder
1341	2600	direct	fr 3	0.87	0.02	
1365	2600	direct	fr 3	0.89	0.08	
1500	2600	direct	fr 3	0.95	0.04	
1527	2600	direct	fr 3	0.95	0.02	
1581	2600	direct	fr 3	0.95	0.01	
2631	3896	direct	fr 3	0.80	0.75	
2640	3896	direct	fr 3	0.80	0.87	
2778	3896	direct	fr 3	0.82	0.28	→ ORF Finder (ATG only)
2814	3896	direct	fr 3	0.81	0.03	2661 start alternatif
2868	3896	direct	fr 3	0.80	0.49	
3900	4751	direct	fr 3	0.74	0.26	→ ORF Finder
3912	4751	direct	fr 3	0.75	0.01	
3966	4751	direct	fr 3	0.80	0.65	
4116	4751	direct	fr 3	0.80	0.27	
4137	4751	direct	fr 3	0.80	0.05	
4158	4751	direct	fr 3	0.79	0.14	
4770	6833	direct	fr 3	0.90	0.82	
4773	6833	direct	fr 3	0.90	0.79	→ ORF Finder
4815	6833	direct	fr 3	0.92	0.13	
4890	6833	direct	fr 3	0.92	0.01	
5226	6833	direct	fr 3	0.93	0.02	
6838	8202	direct	fr 1	0.85	0.06	→ ORF Finder (ATG only)
6877	8202	direct	fr 1	0.88	0.71	6835 start alternatif
6913	8202	direct	fr 1	0.89	0.74	
6925	8202	direct	fr 1	0.89	0.02	
6931	8202	direct	fr 1	0.89	0.01	

# Interpolated Markov Model (IMM)

## Glimmer (Salzberg et al., Nucleic Acids Res.,26,544-48)

Modèle de Markov d'ordre  $k$  : apprendre  $4^{k+1}$  probabilités

Dans le cadre de la prédiction des CDS prise en compte des 6 cadres de lecture, donc nécessité d'apprendre  $6 * 4^{k+1}$  probabilités

Si modèle de Markov d'ordre 5 : 4096 probabilités à définir (hexamères)

Si on considère les 6 cadres de lecture : 24 576 probabilité

Plus l'ordre du modèle est élevé, moins l'estimation des paramètres du modèles va être fiable

Pour certains kmers rares même avec un grand jeu d'apprentissage comme un génome entier, il peut être difficile d'obtenir des estimations précises et inversement pour certains kmers fréquents un modèle de markov d'ordre élevé donnera des estimations plus précises.

Souhait : un modèle de Markov qui utilise les ordres les plus élevés quand il y a assez de données disponibles et des ordres moins élevés dans les cas où les données sont insuffisantes.



Interpolation des modèles de Markov

# Interpolated Markov Model (IMM)

Autrement dit, un IMM utilise une combinaison de toutes les probabilités basées sur 0, 1, 2, ...,  $k$  bases précédentes, où  $k$  est un paramètre donné à l'algorithme. Dans le cas de GLIMMER  $k = 8$

Donc pour les oligomères fréquents, le IMM peut utiliser un modèle d'ordre 8 alors qu'il pourra utiliser par exemple un modèle d'ordre 5 voir d'un ordre inférieur pour des oligomères rares.

Afin de " lisser " ses prédictions, un IMM utilise les prédictions des modèles d'ordre inférieur, pour lesquels on dispose de beaucoup plus de données, afin d'ajuster les prédictions faites à partir des modèles d'ordre supérieur.

Donc les poids attribués aux différents modèles vont définir le poids des modèles inférieurs dans le calcul de la probabilité du modèle d'ordre supérieur.

Ces poids sont appelés les paramètres d'interpolation, avec  $0 \leq \lambda \leq 1$ .



# Interpolated Markov Model (IMM)

Interpolation linéaire simple : combinaison linéaire des probabilités associées aux mots de tailles inférieures à  $k$  contenus dans  $w_k$  (par exemple si  $k = 8$  on regarde les mots de taille 1 à 7)

$$P_{IMM}(x_i|x_{i-k}, \dots, x_{i-1}) = \lambda_0 P(x_i) + \lambda_{1(x_{i-1})}P(x_i|x_{i-1}) + \dots + \lambda_{k(x_{i-k}\dots x_{i-1})}P(x_i|x_{i-k} \dots x_{i-1})$$

Avec 
$$\sum_i \lambda_i = 1$$

Les poids peuvent aussi dépendre des données observées. Pour un ordre donné, on peut avoir plus de données pour estimer certains mots que d'autres.

$$P_{IMM}(x_i|x_{i-k}, \dots, x_{i-1}) = \lambda_0 P(x_i) + \lambda_{1(x_{i-1})}P(x_i|x_{i-1}) + \dots + \lambda_{k(x_{i-k}\dots x_{i-1})}P(x_i|x_{i-k} \dots x_{i-1})$$

Dans ce cas,  $\lambda$  est une fonction des données observées.

# Interpolated Markov Model (IMM)

Estimation des valeurs des paramètres  $\lambda_i$  :

- basée sur la fréquence du mot associé au  $\lambda_i$
- Partir d'un ensemble d'apprentissage

Problème quand nouveau génome, comment être certain que les ORFs sont réellement codantes.

Solution Glimmer :

- ✓ Conserver les séquences présentant des similarités avec des gènes connus d'autres organismes
- ✓ Garder les ORFs longs seulement

Procédure :

- ✓ Identification de toutes les "Open Reading Frames" (ORFs) dans les 6 cadres avec une longueur > seuil (~ 90 bp par défaut)
- ✓ Sélection des ORFs qui vont constituer l'ensemble d'apprentissage
  - à partir de gènes connus (expérimentalement)
  - à partir du génome ORF longueur > 500 bp  
pas chevauchement avec une ORF > 500 bp

**Obtenir un ensemble importants de gènes "fiables"**

# Interpolated Markov Model (IMM)

Estimation des valeurs des paramètres  $\lambda_i$  :

A partir de l'ensemble d'apprentissage :

- ✓ Calcul de la fréquence des mots de longueurs 1 à 9 (rappel ordre le plus élevé de Modèle de Markov fixé à 8)
- ✓ Estimer la probabilité d'observer une base de la séquence en fonction de son contexte, *i.e.*, en fonction des  $i$  bases qui la précèdent, avec  $i \leq 8$

Exemple : Probabilité du 5mer ATTCA

$$P(ATTCA) = P(A|ATTC) = \frac{f(ATTCA)}{f(ATTCA) + f(ATTCC) + f(ATTCT) + f(ATTCA)}$$

avec  $f$  fréquence observée du mot dans l'ensemble d'apprentissage

# Interpolated Markov Model (IMM)

Estimation des valeurs des paramètres  $\lambda_i$  dans GLIMMER :

$$P_{IMM}(x_i | x_{i-k}, \dots, x_{i-1}) = \lambda_0 P(x_i) + \lambda_{1(x_{i-1})} P(x_i | x_{i-1}) + \dots + \lambda_{k(x_{i-k} \dots x_{i-1})} P(x_i | x_{i-k} \dots x_{i-1})$$

avec  $\lambda_{k(x_{i-k} \dots x_{i-1})}$  le poids associé au mot  $x_{i-k} \dots x_{i-1}$

Soit  $c(x_{i-k} \dots x_{i-1})$  le nombre de mots correspondant à  $x_{i-k} \dots x_{i-1}$  dans notre ensemble d'apprentissage :

✓ Si  $c(x_{i-k} \dots x_{i-1}) > 400$  alors  $\lambda_{k(x_{i-k} \dots x_{i-1})} = 1$

✓ Sinon utilisation d'un test statistique : le test du  $\chi^2$

Comparaison de l'occurrence des mots entre deux ordres pour savoir si elles dépendent de l'ordre du modèle de Markov

# Interpolated Markov Model (IMM)

Comparaison des occurrences des mots entre deux ordres pour savoir si elles dépendent de l'ordre du modèle de Markov :

modèle ordre n + base	modèle ordre n-1 + base
$x_{i-n}, \dots, x_{i-1}, a$	$x_{i-n+1}, \dots, x_{i-1}, a$
$x_{i-n}, \dots, x_{i-1}, c$	$x_{i-n+1}, \dots, x_{i-1}, c$
$x_{i-n}, \dots, x_{i-1}, g$	$x_{i-n+1}, \dots, x_{i-1}, g$
$x_{i-n}, \dots, x_{i-1}, t$	$x_{i-n+1}, \dots, x_{i-1}, t$

Hypothèse nulle  $H_0$  : la distribution des valeurs de  $x_i$  est indépendante de l'ordre du modèle

Hypothèse alternative  $H_1$  : cette distribution dépend de l'ordre du modèle

Utilisation du test du  $\chi^2 \rightarrow$  si la p-valeur est grande, on ne peut pas rejeter  $H_0$

Calcul de  $d$  avec  $d = 1 - p\text{-valeur}$

- Si  $d$  est petit, pas besoin de prendre en compte l'ordre le plus grand,  $\lambda$  vaudra 0 pour cet ordre
- sinon pondération de  $d$  par le nombre de mots pour calculer  $\lambda$

# Interpolated Markov Model (IMM)

Résumé de la détermination des poids  $\lambda$  dans GLIMMER :

Le poids  $\lambda_{k(x_{i-k} \dots x_{i-1})}$  associé au mot  $x_{i-k} \dots x_{i-1}$  vaudra :

$$\lambda_{k(x_{i-k} \dots x_{i-1})} = \begin{cases} 1 & \text{si } c(x_{i-k} \dots x_{i-1}) > 400 \\ 0 & \text{si } d < 0.5 \\ d \times \frac{c(x_{i-k} \dots x_{i-1})}{400} & \text{si } d \geq 0.5 \end{cases}$$

# Interpolated Markov Model (IMM)

Exemple : Supposons que nous avons les fréquences observées des mots suivants dans notre ensemble d'apprentissage

ordre 3		ordre 2		ordre 1	
<b>ATC</b> A	25	<b>TC</b> A	100	<b>C</b> A	175
<b>ATC</b> C	40	<b>TC</b> C	90	<b>C</b> C	140
<b>ATC</b> G	15	<b>TC</b> G	35	<b>C</b> G	65
<b>ATC</b> T	20	<b>TC</b> T	75	<b>C</b> T	120
total	100		300		500

$\chi^2$  test :

$$d = 0.857$$

$$d = 0.140$$

$$\lambda_3(\text{ATC}) = 0.857 \times 100/400 = 0.214$$

$$\lambda_2(\text{TC}) = 0 \quad (d < 0.5 \text{ et } c < 400)$$

$$\lambda_1(\text{C}) = 1 \quad (c > 400)$$

# Interpolated Markov Model (IMM)

Comment ensuite calculer la probabilité qu'une nouvelle séquence soit codante :

GLIMMER calcule la probabilité que le modèle  $M$  génère la séquence étudiée  $S$  soit si la longueur de la séquence est  $n$  :

$$P(S|M) = \sum_{x=1}^n IMM_8(S_x) \quad \text{où } S_x \text{ est l'oligomère se terminant à la position } x$$

$IMM_8(S_x)$  est le score du modèle de Markov interpolé d'ordre 8 et il est calculé de façon récursive :

$$\begin{aligned} IMM_k(S_x) &= P_{IMM_k}(x_i | x_{i-k}, \dots, x_{i-1}) \\ &= \lambda_{k(x_{i-k} \dots x_{i-1})} \times P(x_i | x_{i-k} \dots x_{i-1}) + [1 - \lambda_{k(x_{i-k} \dots x_{i-1})}] \times P_{IMM_{k-1}}(x_i | x_{i-k+1}, \dots, x_{i-1}) \end{aligned}$$



# Interpolated Markov Model (IMM)

Rappel :

$$P_{IMM,n}(x_i|x_{i-n}, \dots, x_{i-1}) = \lambda_n(x_{i-n}, \dots, x_{i-1})P(x_i|x_{i-n}, \dots, x_{i-1}) \\ + [1 - \lambda_n(x_{i-n}, \dots, x_{i-1})]P_{IMM,n-1}(x_i|x_{i-n+1}, \dots, x_{i-1})$$

Si on veut calculer  $P_{IMM,3}(G|ATC)$  :

$$P_{IMM,1}(G|C) = \lambda_1(C)P(G|C) + [1 - \lambda_1(C)]P_{IMM,0}(G) = P(G|C) \quad \text{car } \lambda_1(C) = 1$$

$$P_{IMM,2}(G|TC) = \lambda_2(TC)P(G|TC) + [1 - \lambda_2(TC)]P_{IMM,1}(G|C) = P_{IMM,1}(G|C) = P(G|C) \\ \text{car } \lambda_2(TC) = 0$$

$$P_{IMM,3}(G|ATC) = \lambda_3(ATC)P(G|ATC) + [1 - \lambda_3(ATC)]P_{IMM,2}(G|TC) \\ = 0.214 P(G|ATC) + (1 - 0.214) P_{IMM,2}(G|TC) \quad \text{car } \lambda_3(ATC) = 0,214$$

$$P_{IMM,3}(G|ATC) = 0.214 P(G|ATC) + 0.786 P(G|C)$$

# Interpolated Markov Model (IMM)

$$P_{IMM,3}(G|ATC) = 0.214 P(G|ATC) + 0.786 P(G|C)$$

$$P(G|C) = P(CG) = \frac{f(CG)}{f(CA) + f(CC) + f(CG) + f(CT)} = \frac{65}{500} = 0.13$$

$$P(G|ATC) = P(ATCG) = \frac{f(ATCG)}{f(ATCA) + f(ATCC) + f(ATCG) + f(ATCT)} = \frac{15}{100} = 0.15$$

$$P_{IMM,3}(G|ATC) = 0.214 \times 0.15 + 0.786 \times 0.13 = 0.13428$$

# Identification des CDS avec GLIMMER

- ✓ n'utilise pas de fenêtre glissante
- ✓ Identifie en premier les ORF plus long qu'un certain seuil
- ✓ Calcul le score de chacun dans les six cadres de lecture comme suit :

La probabilité que le modèle M génère la séquence S est :

$$P(S|M) = \sum_{x=1}^n IMM_8(S_x)$$

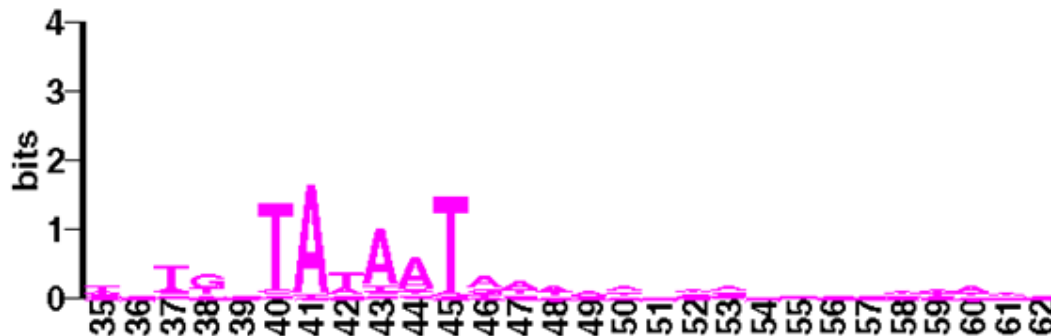
Avec  $S_x$  l'oligomer se terminant à la position  $x$  et  $n$  la taille de la séquence

- ✓ Les ORF qui ont un score supérieur à un seuil spécifié dans le cadre de lecture correct sont sélectionnées pour un traitement ultérieur des chevauchements

# Traitement de l'information de type signal

Différentes façon de représenter la conservation des séquences impliquées dans un processus donné (promoteur lors de la transcription, ribosome binding site lors de la traduction, jonction d'épissage etc...) et ensuite de rechercher ces « signaux » dans une nouvelle séquence.

Compilation of *Bacillus subtilis* sigma A-dependent promoter elements



# Petit rappel : Motifs

**Définition** : zone d'une séquence nucléique ou protéique présentant une conservation quand on compare plusieurs séquences.

- correspondent en général à des zones fonctionnelles
- ADN et ARN : aussi appelé **signal**, ces zones interviennent souvent dans des systèmes de régulation, ex :
  - -10 et -35 des promoteurs chez les procaryotes, jonction d'épissage,
  - boîte CRE (catabolite repression element) : après mise en évidence de certains gènes soumis à la répression catabolique chez *B. subtilis*, l'identification du signal permet de rechercher dans le génome complet les boîtes CRE et donc les gènes qui pourraient être soumis à la répression catabolique.
- différents des signaux reconnus par les enzymes de restrictions qui reconnaissent des séquences exactes, ex: GAATTC pour ECOR1.
- Les motifs et profils présentent une certaine **variabilité** (souvent impliquée dans la variabilité de la régulation par une reconnaissance plus ou moins forte des partenaires)

**Comment représenter cette variabilité ?**

- séquence consensus
- matrice de poids

# Représentation : Séquence consensus

## Exemples des boîtes CRE:

<i>acsA</i>	TGAAAGCGTTACCA
<i>acuA</i>	TGAAAACGCTTTAT
<i>amyE</i>	TGTAAGCGTTAACA
<i>gntR</i>	TGAAAGCGGTACCA
<i>hutP</i>	TGAAACCGCTTCCA
<i>licS</i>	AGAAAACGCTTTCA
<i>xylA</i>	TGGAAGCGTAAACA
<i>xylA</i>	TGAAAGCGCAAACA
<i>xylA</i>	AGTAAGCGTTTACA
<i>ackA</i>	TGTAAGCGTTATCA
consensus	<b>TGAAAGCGNTAACA</b>
	<b>T TC</b>

# Représentation : Matrice de poids

Exemples de 242 séquences de promoteurs (-10) chez *E. coli* :

Matrices des fréquences de chaque base  $b$  à chaque position  $i$  ( $f_{b,i}$ ) du motif -10 (6 positions) :

Pos .	1	2	3	4	5	6
A	0.04	0.88	0.26	0.59	0.49	0.03
C	0.09	0.03	0.11	0.13	0.22	0.05
G	0.07	0.01	0.12	0.16	0.12	0.02
T	0.80	0.08	0.51	0.13	0.18	0.89

Avec

$$f_{b,i} = n_{b,i} / n_{tot}$$

$n_{tot}$  : nombre total de séquences analysées

# Représentation : Matrice de poids position (Position Weight Matrix, PWM)

Exemples de 242 séquences de promoteurs (-10) chez *E. coli* :

Normalisation de la matrice : log matrice  $\log_2(f_{b,i}/P_b)$

$f_{b,i}$  = fréquence observée de la base  $b$  à la position  $i$  dans toutes les séquences

$P_b$  = fréquence de cette base dans l'ensemble du génome

Pos.	1	2	3	4	5	6
A	-2.76	1.88	0.06	1.23	0.96	-2.92
C	-1.46	-3.11	-1.22	-1.00	-0.22	-2.21
G	-1.76	-5.00	-1.06	-0.67	-1.06	-3.58
T	1.67	-1.66	1.04	-1.00	-0.49	1.84

Le rapport  $f_{b,i}/P_b$  est une mesure de l'écart entre fréquence observée et attendue.



# Utilisation d'une matrice de poids sur une séquence

Pos.	1	2	3	4	5	6
A	-28	18	1	12	10	-29
C	-15	-31	-12	-10	-2	-22
G	-18	-50	-11	-7	-11	-36
T	17	-17	10	-10	-5	18

**A** CTATAATCG

$$\text{Score1} = -15 - 17 + 1 - 10 + 10 - 29 = -60$$

**AC** TATAATCG

$$\text{Score2} = 17 + 18 + 10 + 12 + 10 + 18 = 85$$

**ACT** ATAATCG

$$\text{Score3} = -28 - 17 + 1 + 12 - 5 - 22 = -59$$

# Théorie de l'information : obtention de WebLogo

Shannon et Weaver (1949).



La valeur de l'information  $I$  à la position  $j$  d'un signal est donnée par :

$$I(j) = \sum_i f_{ij} \log_2 f_{ij} - \sum_i P_i \log_2 P_i$$

où :

$P_i$  ( $i = 1$  à  $4$ ) est la fréquence de la base  $i$  dans l'ensemble du génome (probabilité théorique)

$f_{ij}$  est la fréquence observée de la base  $i$  à la position  $j$  d'un signal sur un ensemble d'exemples.

Les  $P_i$  étant estimées à 0.25 pour chacune des 4 bases on a :

$$\sum_i P_i \log_2 P_i = -2$$

donc

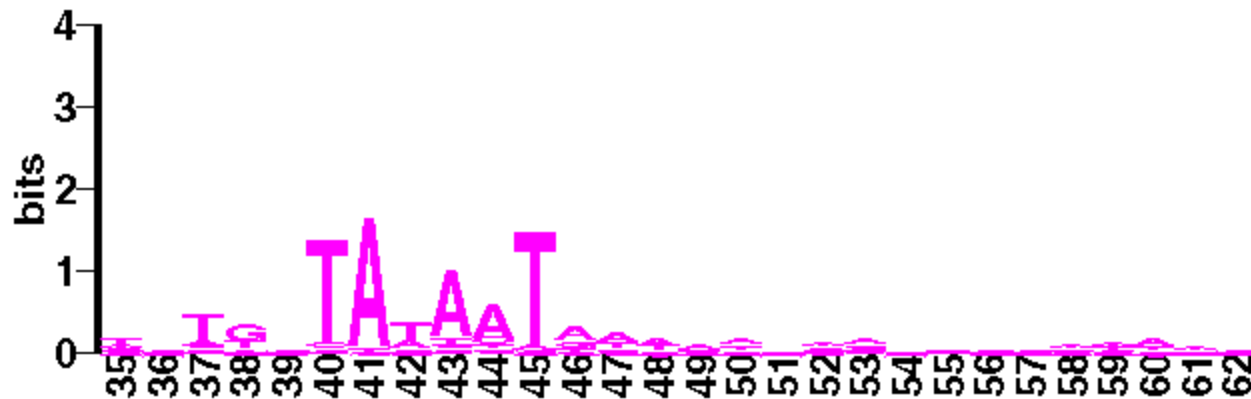
$$I(j) = \sum_i f_{ij} \log_2 f_{ij} + 2$$

Les positions du signal qui contiendront de l'information seront celles qui auront une composition très biaisées par rapport à ce qui est attendu.

Si à une position  $j$  du signal, présence d'une seule base invariante  $i$  alors  $f_{ij} = 1$  et  $\log_2 f_{ij} = 0$  donc  $f_{ij} \log_2 f_{ij} = 0$  et les fréquences observées des autres bases sont nulles. On aura

$$I(j) = 2 \text{ information maximale}$$

### Compilation of *Bacillus subtilis* sigma A-dependent promoter elements



# Recherche des signaux d'initiation de la traduction



Programme utilisé: Scan\_For\_Matches

Motif du Shine-Dalgarno recherché: **GGAGG 6...11 DTG** correspond à la présence de la séquence GGAGG à 6 ou 11 pb en amont d'un codon AUG, GUG ou UUG.

Résultats:

```
BS: [189, 204] : ggagg cagataca atg -> A
BS: [3175, 3192]: ggagg tcgacttttt ttg -> dans le gène C
BS: [3887, 3902]: ggagg cataagggt atg -> D
BS: [4760, 4775]: ggagg agaatgtg atg -> E
BS: [7501, 7516]: ggagg atttgccg gtg -> dans le gène F
```

Donc:

**Gène A : début en 202**  
**Gène D : début en 3900**  
**Gène E : début en 4773**

Les autres SD des gènes B, C et F trouvés avec une autre représentation (matrice de poids) car ils sont modifiés.

```
AAGGAGGTG consensus
GAAAGGGTG GTG pour B
AGAGAGGTG GTG pour C
GGGGGATG ATG pour F
```

## Unités de traduction prédites

202	1194	direct	fr 1	->	A	
1335	2600	direct	fr 3	->	B	SD modifié
2640	3896	direct	fr 3	->	C	SD modifié
3900	4751	direct	fr 3	->	D	
4773	6833	direct	fr 3	->	E	
6877	8202	direct	fr 1	->	F	SD modifié

# Recherche des unités de transcription

Chez *B. subtilis*, l'initiation de la transcription fait intervenir le facteur sigma A qui reconnaît une séquence spécifique localisée environ en -10 et -35 pb du +1 de transcription.

**Séquence consensus:            TTGACA 16...35 TATAAT**

Grand nombre de promoteurs de type sigma A identifiés expérimentalement chez *B. subtilis* :



**matrices de poids**

# Résultats de la recherche des promoteurs

Utilisation du programme Scan\_For\_Matches et de la matrice de poids



**-35** **-10**  
BS:[1264,1292]: tttaca cattttctcaggcatgc tatatt  
BS:[131,158] : ttgaca tttgtgaaaaagaaag taaaat

# Liste des logiciels pour la prédiction des promoteurs chez les bactéries

July/August 2020 Volume 5 Issue 4 e00439-20

**TABLE 1** General information on the tools used here

Tool	Method	Training sequence data set <sup>a</sup>	No. of <i>E. coli</i> sigma factors	Availability	Yr	Reference	No. of citations (Google Scholar) <sup>b</sup>
BPROM	Weight matrices of different motifs combined with linear discriminant analysis	Positive: Experimentally validated promoters from <i>E. coli</i> (14). Negative: Inner regions of protein-coding ORFs.	70	Web server <a href="http://www.softberry.com/berry.phtml?topic=bprom&amp;group=programs&amp;subgroup=gfindb">http://www.softberry.com/berry.phtml?topic=bprom&amp;group=programs&amp;subgroup=gfindb</a>	2011	33	427
bTSSfinder	Position weight matrices for promoter elements, oligomer frequencies, physicochemical properties as features, and Mahalanobis distance for feature selection and with neural network for classification	Positive: Experimentally validated TSSs from Regulon DB. [-200, +51]. Negative: Genomic regions with no experimental evidence for the presence of TSSs.	24, 28, 32, 38, 70	Stand-alone and Web server <a href="http://www.cbrc.kauit.edu.sa/btssfinder/">http://www.cbrc.kauit.edu.sa/btssfinder/</a>	2016	23	26
BacPP	Weighted rules extracted from neural network	Positive: Regulon DB available promoters. [-60, +20]. Negative: randomly generated sequences (with established nucleotide frequencies) and intergenic sequences.	24, 28, 32, 38, 54, 70	Web server <a href="http://www.bacpp.bioinfoc.com/home">http://www.bacpp.bioinfoc.com/home</a>	2011	17	22
Virtual Footprint	PWMs from different available databases			Web server <a href="http://www.prodoric.de/vfp/vfp_promoter.php">http://www.prodoric.de/vfp/vfp_promoter.php</a>	2005	36	370
IBBP	Image-based and evolutionary approach which generates "images" (template-image strings that keep features of spatial sequence relationships)	Positive: sigma 70 promoters from Regulon DB. [-60, +20]. Negative: randomly generated from protein-coding sequences.	70 (expandable approach)	Source code <a href="https://github.com/hahatcdg/IBBP">https://github.com/hahatcdg/IBBP</a>	2018	35	1
iPro70-FMWin	22,595 features extracted from sequence and AdaBoost to select the most representatives among them; logistic regression classifier	Positive: Regulon DB annotated promoters. [-60, 20]. Negative: randomly generated from protein-coding and intergenic region sequences.	70	Webserver <a href="http://ipro70.pythonanywhere.com/">http://ipro70.pythonanywhere.com/</a>	2019	38	4
70ProPred	Support vector machine using position-specific tendencies of trinucleotide and electron-ion interaction pseudopotentials as features	Positive: promoters from Regulon DB. [-60, 20]. Negative: randomly generated from coding and noncoding sequences.	70	Webserver <a href="http://server.malab.cn/70ProPred/">http://server.malab.cn/70ProPred/</a>	2017	39	33
CNNProm	Convolutional neural networks	Positive: promoters from Regulon DB. [-60, 20]. Negative: the opposite chain of randomly selected protein-coding genes.	70	<a href="http://www.softberry.com/berry.phtml?topic=index&amp;group=programs&amp;subgroup=deeplearn">http://www.softberry.com/berry.phtml?topic=index&amp;group=programs&amp;subgroup=deeplearn</a>	2017	34	60
MULTiPly	Support vector machine using biprofile Bayes, KNN features, k-tuple nucleotide compositions, and dinucleotide-based auto-covariance as features	Positive: promoters from Regulon DB. [-60, 20].	24, 28, 32, 38, 54, 70	Web server and stand-alone <a href="http://flagshipnt.erc.monash.edu/MULTiPly/">http://flagshipnt.erc.monash.edu/MULTiPly/</a>	2019	41	30
iPromoter-2L	Multiwindow-based pseudo k-tuple nucleotide composition with physicochemical properties as features and Random Forest as a predictor	Positive: promoters from Regulon DB. [-60, 20]. Negative: randomly extracted from the middle regions of long coding sequences and convergent intergenic region (none of the promoters in each set has more than 0.8 pairwise sequence identity)	24, 28, 32, 38, 54, 70	Web server <a href="http://bioinformatics.hitsz.edu.cn/iPromoter-2L/">http://bioinformatics.hitsz.edu.cn/iPromoter-2L/</a>	2018	40	180

<sup>a</sup>Positive, positive sequences, sequences expected to be promoters. Negative, negative sequences, sequences expected to not include promoters. The interval of the sequence with the boundary numbers related to a TSS is indicated within brackets ([-60, +20], [-60, +19], or [-200, +51]).

<sup>b</sup>Citations checked on 3 May 2020.

mSystems 2020, 5(4):e00439-20



# Liste des logiciels pour la prédiction des promoteurs chez les bactéries

**TABLE 2** Usage characteristics of the tools analyzed here

Tool	Multifasta	Big files	Shows promoter core	Score or probability	Uppercase only	Output format	Execution time	Follow up	Interface	Comment(s)
BPROM	No	Yes	Yes	Yes	No	Text on screen	Fast	Progress on screen	Webform, simple, intuitive	Multifasta not supported and sequence boxes are not shown; difficult to process the results
bTSSfinder	Yes	Yes	Yes	Yes	No	Text file, GFF file, BED file	Fast	Progress on screen	Login needed, webform, simple, intuitive	Flexible configurations of cutoff values; results saved for 1 week; Linux tool available for download; it needs a large promoter sequence (-200, +50, related to the putative TSS)
BacPP	Yes	No	No	Yes	No	Text on screen or text file	Fast	N	Login needed, webform, simple, intuitive	Short tests per time
Virtual Footprint	No	Yes	Yes	Yes	No	Text on screen	Medium fast	Progress on screen	Webform, many fields, and option in the screen	Integrated with a large PWM database of TFBS; applicable to many species; limited to the position weight matrix available
IBBP	No	Yes	It shows the putative TSS	Yes	No	Text file	Fast	Progress on screen	Command line	Windows SO only execution; requires the manual input files; training and test procedures are separated; fast for big files; it can be used as an approach to the initial prediction of any type of promoters
iPro70-FMWin	Yes	Yes	No	Yes	No	Text on screen	Fast	No	Webform, simple, intuitive	High accuracy
70ProPred	Yes	Yes	No	No	Yes	Text on screen	Fast	No	Webform, simple, intuitive	High accuracy; it does not accept a file as input, just text on a form
CNNProm	Yes	Yes	No	Yes	Yes	Text on screen	Long time (for genomes), fast for multifasta	No	Webform, simple, intuitive	Useful for large sequences (genomes); without a follow-up display; multifasta not supported and sequence boxes are not shown; difficult to process the results
MULTIPly	Yes	Yes	No	No	No	Text on screen or text file	Medium time	Progress on screen, job ID to find the result later	Webform, simple, intuitive	Good accuracy; it saves the result; time-consuming for large sequences
iPromoter-2L	Yes	Yes	No	No	No	Text on screen	Fast	Progress on screen	Webform, simple, intuitive	Good accuracy

Extrait de : Cassiano MHA, Silva-Rocha R. Benchmarking Bacterial Promoter Prediction Tools: Potentialities and Limitations. *mSystems*. 2020 Aug 25;5(4):e00439-20. doi: 10.1128/mSystems.00439-20.

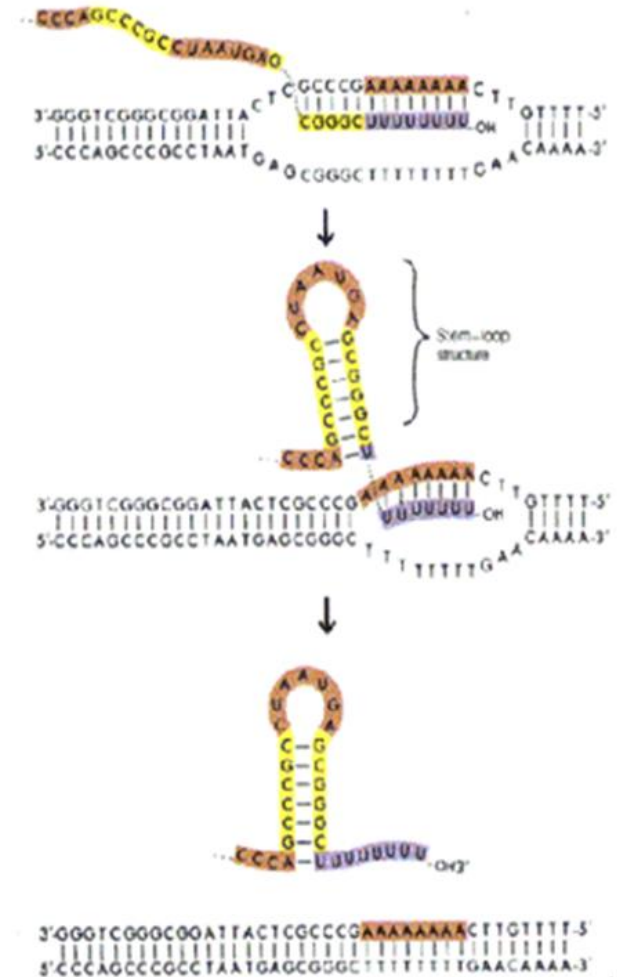
Au niveau séquence, on ne sait modéliser que les terminateurs Rho indépendants.

Mécanisme proposé pour les terminateurs Rho indépendants.

Quand l'ARN est en cours de transcription, on a une hybridation ARN/ADN sur environ 12 pb.

Le site de terminaison de la transcription est précédé par une séquence capable de former une structure secondaire stable. Il y a compétition entre la formation de cette structure et l'appariement avec l'ADN. La présence d'un poly(U) en cours de synthèse déplace l'équilibre en faveur de la tige-boucle et il y a alors décrochage de l'ARN et arrêt de la transcription.

Dans les séquences, on va donc rechercher des séquences répétées inversées suivies d'un poly(U).



# Terminateurs rho indépendant

Le modèle : Formation d'une tige boucle en amont d'une région riche en U qui déstabilise l'appariement ADN/ARN et conduit au décrochage de l'ARN.

Deux classes de terminateurs:

- ✓ petite tige de 5 à 7 pb très stable et d'une boucle de 4 pb suivie d'une région riche en U.
- ✓ une longue tige qui peut se décomposer en deux tiges imbriquées l'une dans l'autre.
  - La première plus stable doit faire au moins 3 pb de long avec un appariement GC à son pied.
  - La seconde est incluse dans la première et comporte au moins 3 appariements. Elle est généralement moins stable que la première. La boucle est de 3 à 7 pb de long.

Utilisation de scan\_for-matches

Un site éventuel pour prédire les terminateurs de transcription ou au moins vérifier la probabilité de ceux sélectionnés avec scan\_for\_matches

<http://lin-group.cn/server/iTerm-PseKNC/predictor.php>

# Résultat de la recherche des terminateurs sur le fragment de *B. subtilis*

1199-1223

```

  A C
G   A
G-C
T-A
T-A
C-G
A-T
G-C
T-A
  T
  T
  T
  T
  T
  T
  
```

6843-6866

```

      T
  A   A
  G.T
  C-G
  T.G
  C-G
  G-C
  C-G
  C-G
    A
    T
    T
    C
    T
    T
    T
  
```

75-103

```

  A A
  C   A
  G.T
  C-G
  C-G
  G.T
  A-T
  G-C
  T-A
  C-G
  A-T
  G-C
    T
    T
    T
    T
    T
  
```

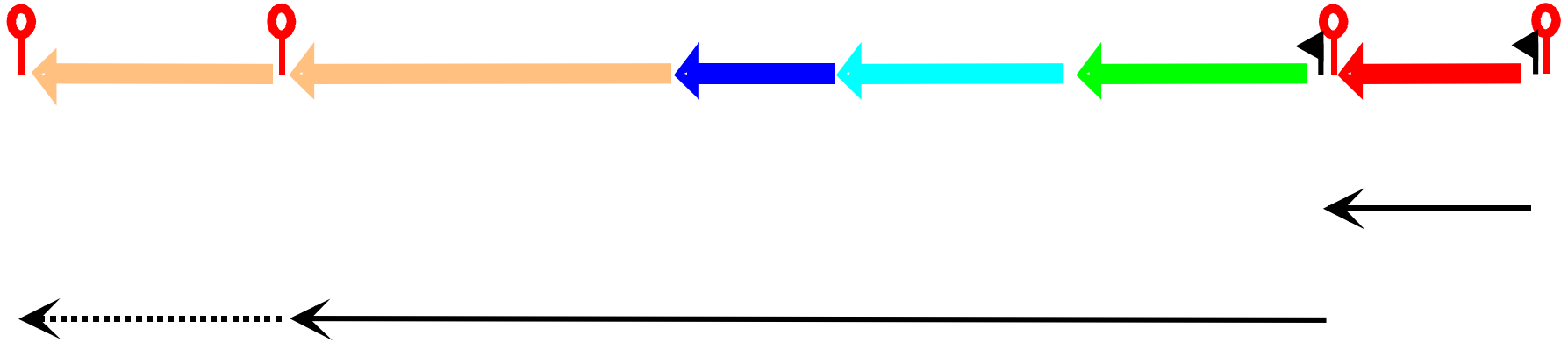
8215-8256




```

      T C
  A   A
  A-T
  A-T
  C-G
  A-T
  A-T
  T-A
  G.T
  T.G
  A-T
  G-C
  A-T
  G-C
  T-A
  A-T
  C-G
  C-G
  
```

ATCATT

# Prédiction des unités de traduction et de transcription



-  terminateur rho-indépendant
-  promoteurs de transcription de type sigma
-  transcrit putatif

# Prédictions fonctionnelles

## Identification

- homologues
- motifs
- domaines

## Localisation cellulaire

- fragments trans-membranaires
- peptide signal

## Structure

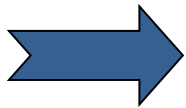
- secondaire
- tertiaire

## Recherche de liens fonctionnelles

- réseaux de régulation
- voies métaboliques
- interactions moléculaires

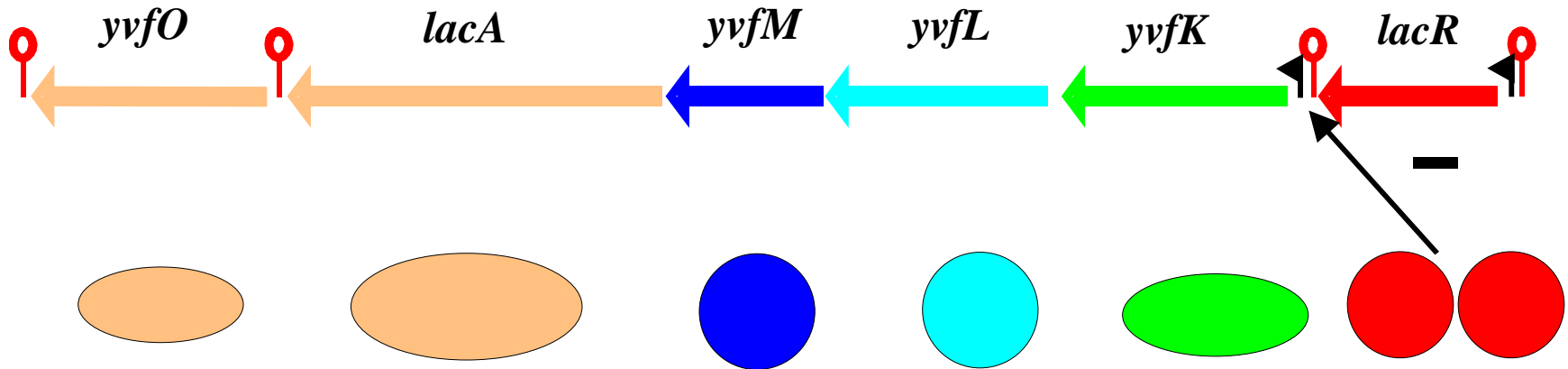
## Prédiction fonctionnelle

Recherche par similitude dans les bases de données : programme BLAST

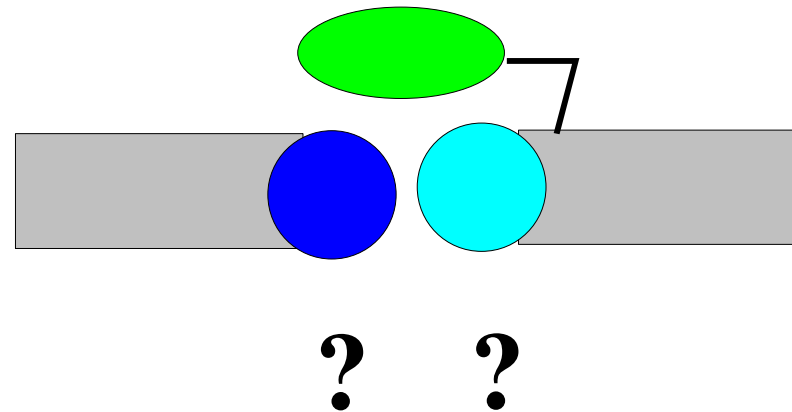


- A LACR protéine régulatrice de type LacI/GalR
- B YVFK protéine affine d'un ABC transporteur
- C YVFL perméase d'un ABC transporteur
- D YVFM perméase d'un ABC transporteur
- E LACA galactosidase
- F YVFO arabino-galactosidase

# Synthèse des résultats



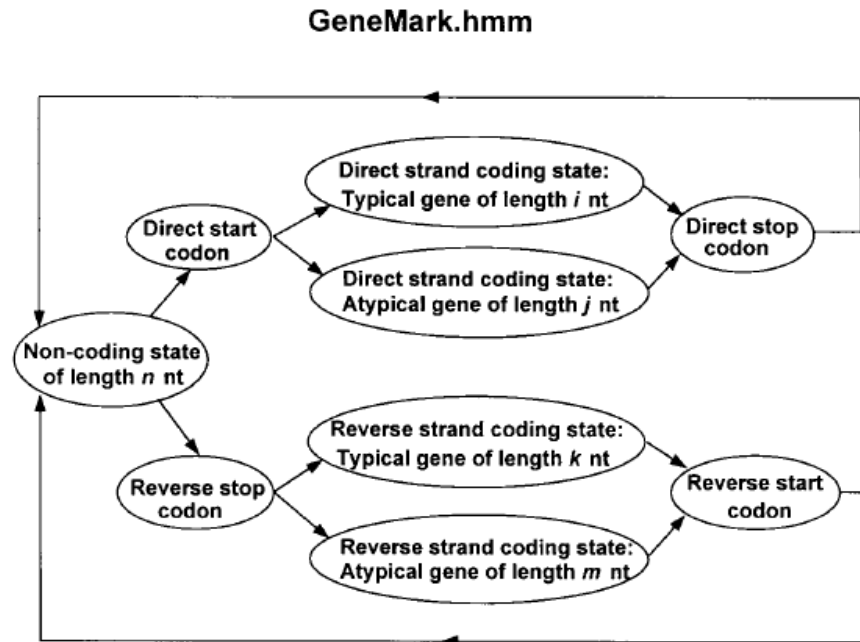
## Systeme à la membrane





# Modèles de gènes : génomes procaryotes

## Modèle de Markov caché (HMM : Hidden Markov Model)



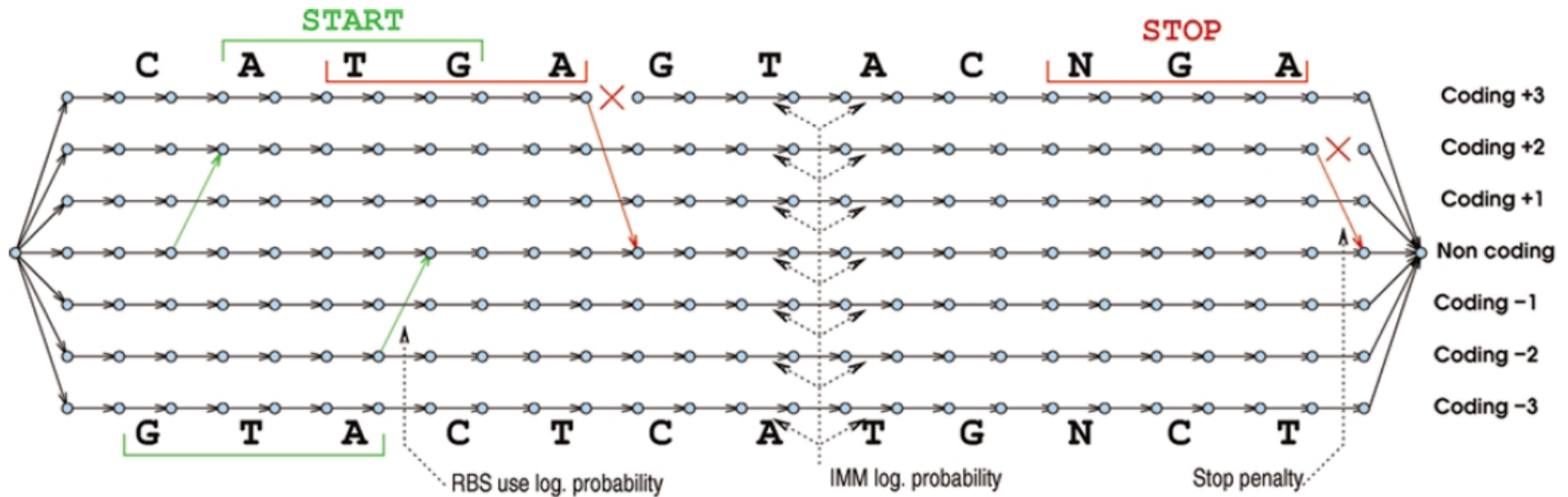
**Figure 1.** Hidden Markov model of a prokaryotic nucleotide sequence used in the GeneMark.hmm algorithm. The hidden states of the model are represented as ovals in the figure, and arrows correspond to allowed transitions between the states.

(extrait de Nucleic Acids Res. (1998), 26, 1107-1115)

# Modèles de gènes : génomes procaryotes

## Graphe orienté sans circuit (DAG : Directed Acyclic Graph)

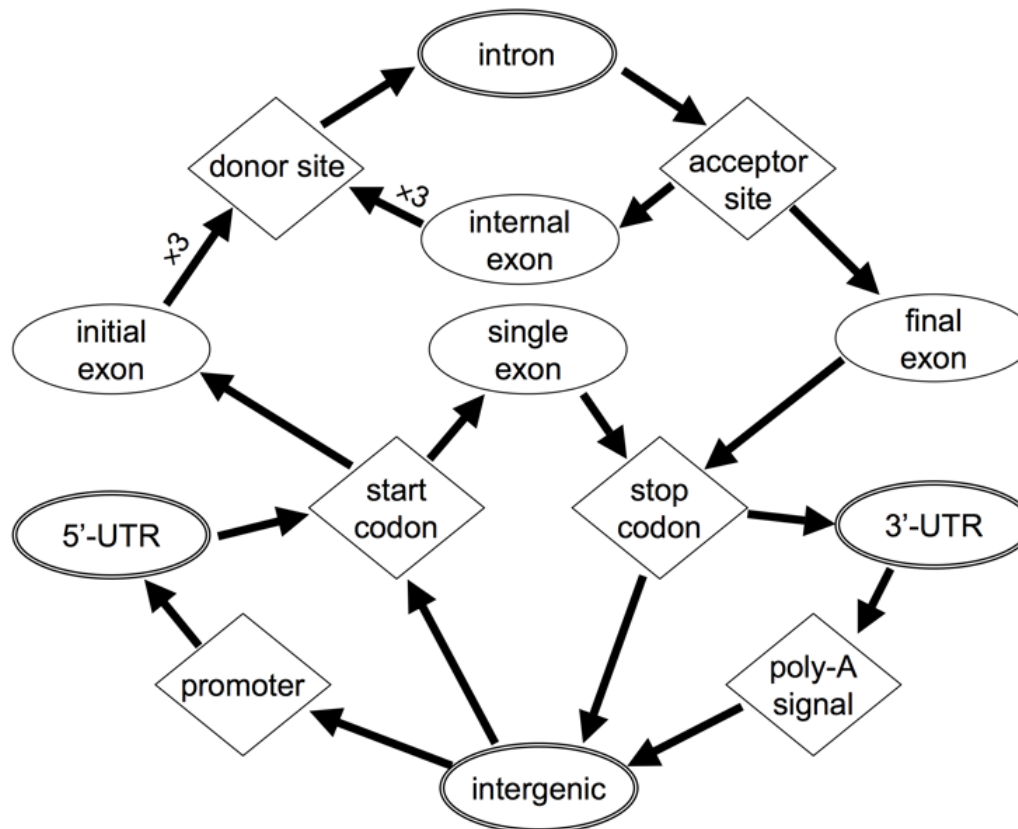
**FramedD** : Nucleic Acids Res., (2003), 31, 3738-41



**Figure 1.** A simplified view of the directed acyclic graph built for analyzing the sequence CATGAGTACNGA. This view ignores the additional complexity induced by gene overlapping regions and frameshift modeling. The occurrence of a START codon at position 2 to 4 induces a 'signal' edge that goes from the non-coding track to the +2 coding track. Similarly, the occurrence of the NGA codon at the end induces a STOP signal edge. Edge weights sources are indicated using dotted arrows.

## Modèle de Markov caché (HMM : Hidden Markov Model)

Exemple du modèle simplifié de GENSCAN  
(extrait de J. Mol. Biol. (1997) 268, 78-94)



Information externe à la séquence elle-même (de type extrinsèque) contrairement au contenu statistique ou aux signaux qui sont internes à la séquence (de type intrinsèque)

Comparer la séquence à analyser avec des séquences connues peut permettre de refléter la présence de gènes et donner des informations sur leur structure. Notamment, la structure en exons/introns pour les gènes eucaryotes.

Types de séquences utilisées pour la comparaison :

- les ADNc
- les EST
- les protéines
- des séquences génomiques

# Information de type similarité

Méthodes prédisant la structure en exons-introns par alignement de la séquence génomique soit avec un ARNm (ou un ADNc), soit avec une protéine. Parmi les plus utilisés, on trouve :

Méthode	Séquence de référence	Référence
BLAT	ARNm ou protéine	Genome Research 12(4):656-64 (2002).
sim4	ARNm	Genome Research 8:967-74 (1998).
Scipio	ARNm ou protéine	BMC Bioinformatics 9:278 (2008)
GeneWise	protéine	Genome Research 14(5):988-95 (2004)
GenomeWise	ADNc, EST	Genome Research 14(5):988-95 (2004)
WebScipio	protéine	Bioinformatics 12:270 (2011)

L'extension du logiciel WebScipio permet de rechercher une forme spécifique d'épissage alternatif (exons mutuellement exclusifs)

Avec des séquences protéiques : GENOMESCAN (intégration de cette information dans le modèle GENSCAN).

Combine le résultats d'une méthode *ab initio* de recherche de la structure en exons-introns des gènes (GenScan) avec une recherche par similarité avec les protéines connues (BlastX)

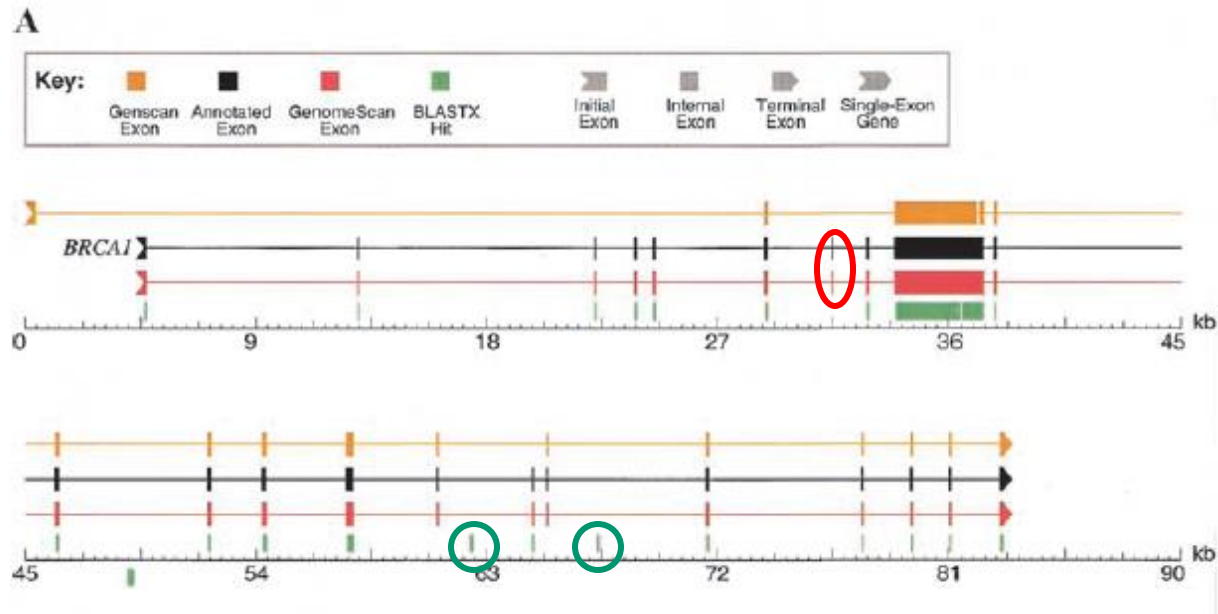
Différence par rapport à GenScan (méthode utilisant un HMM) : la probabilité du chemin est renforcée si la région prédite comme exon présente un hit avec BlastX et est diminuée dans le cas contraire

- Les hits de BlastX qui tombent en N-ter ou C-ter de la protéine sont utilisés pour mieux prédire les codons initiateur et terminateur
- Deux hits BlastX adjacents sur la protéine mais séparés par une distance  $\geq 60$  pb dans la séquence génomique sont utilisés pour prédire les régions introniques putatives
- si plusieurs hits BlastX chevauchants (information redondante), les résultats sont pré-traités pour ne garder que le hit le plus significatif.

# Prise en compte de la similarité

Avec des séquences protéiques : GENOMESCAN (intégration de cette information dans le modèle GENSCAN.

Exemple d'un résultat de prédiction (extrait de Genome Research (2001), 11, 803-816)



Hit BlastX faux-positifs -> pas de prédiction d'exon

Exon 7 : prédit par GenomeScan alors que pas prédit par GenScan et pas de hit BlastX -> pas seulement addition des deux résultats mais fait également sa propre inférence. Utilise le fait qu'il y a une incompatibilité de phase de l'intron entre les exons 6 et 8 -> nécessite la présence d'un autre exon entre les deux.

# Prise en compte de la similarité

Avec des séquences génomiques : TWINSCAN (Bioinformatics (2001), 17 suppl. 1, S140-S148, intégration de cette information dans le modèle GENSCAN.

Exploite la similarité entre deux génomes apparentés : ceci permet de détecter les régions codantes et les éléments de régulation.

target sequence : séquence à annoter

informant sequence : séquence génomique apparentée

Les répétitions sont masquées. Alignement des deux séquences avec le programme Wu-BlastN

## Codage de la conservation

La conservation de la séquence est traduite en symboles sous la séquence ADN :

| = match

: = mismatch

. = non aligné

Les gaps dans la séquence « target » sont ignorés et ceux dans la séquence « informant » sont considérés comme des mismatches.

	10	20	30
position dans la séquence target	123456789	123456789	123456789
séquence target	ATTTAGCCTACTGAAATGGACCGCTTCAGCATGGTATCC		
séquence conservation	:    ..... : :       : : : : :		

**Fig. 1.** An example DNA sequence together with the corresponding conservation sequence.



## Calcul de la probabilité

Quand on recherche le chemin le plus probable dans le HMM, calcul de la probabilité que la séquence soit dans l'état  $i$  (par exemple : exons, UTR etc.)

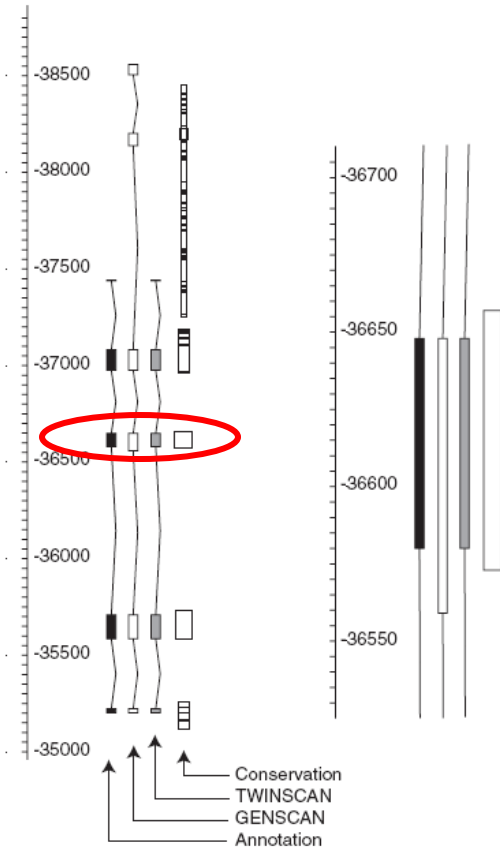
$$P_e(i) = \text{Proba\_GeneScan}(i) \times \text{Proba\_Conservation}(i)$$

## Calcul de la probabilité de conservation

Modèle de Markov d'ordre 5

Estimée par état (codant, UTR, non codant...) à partir de séquences annotées formant l'ensemble d'apprentissage.

## Avec des séquences génomiques : TWINSCAN (intégration de cette information dans le modèle GENSCAN.



Exemple d'un résultat de prédiction (extrait de Bioinformatics (2001), 17 suppl. 1, S140-S148)

GenScan ne prédit pas le premier exon et en prédit deux supplémentaires.

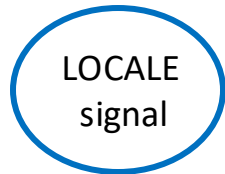
TwinScan prédit correctement le premier exon car le niveau de conservation de séquence à son niveau augmente la probabilité totale de l'exon.

Exon 3 : mauvais site d'épissage en 5' choisi par GenScan. Bonne prédiction de TwinScan car la conservation de séquence s'arrête avant la position de fin de l'exon prédit par GenScan

Fig. 5. Detailed view of the annotation, gene predictions and conservation at the L44L gene (AAB47245.1) from the *Mus musculus* Bruton's tyrosine kinase locus (U58105.1). The magnification at right shows the region around exon 3. The width of boxes representing BLAST alignments corresponds to the quality of the alignment. The image comes from ACEDB.

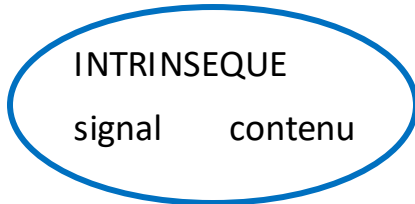
# Evolution de l'intégration des sources d'informations

(extrait de la thèse de Sylvain Foissac, 2004)



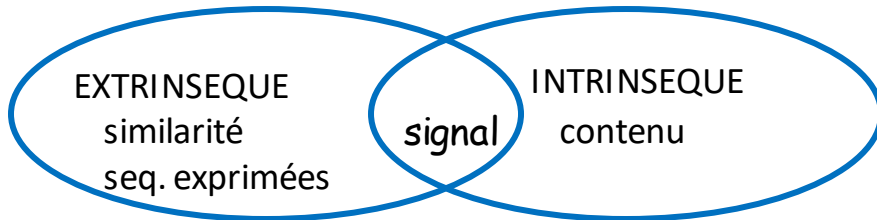
Deux sources traitées par des méthodes indépendantes (ex : Staden, 1984; Gelfand, 1990)

---



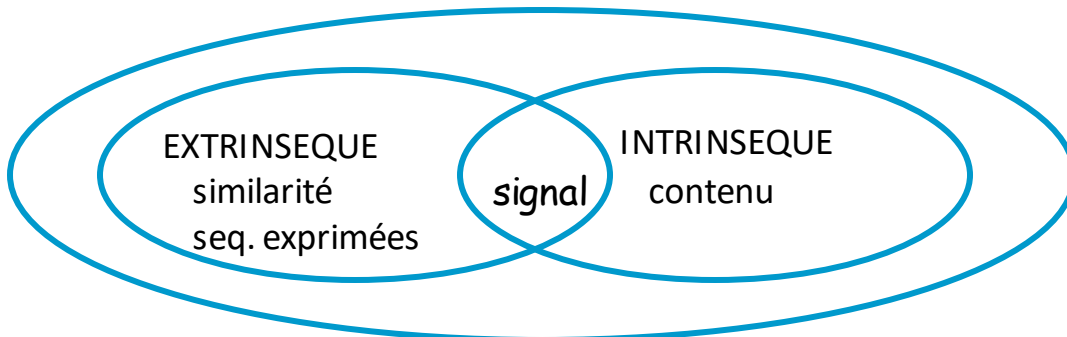
Intégration de ces deux sources dans un même logiciel (ex : Guigo et al., 1992, logiciel GENSCAN (Burge et Karlin, 1997))

---

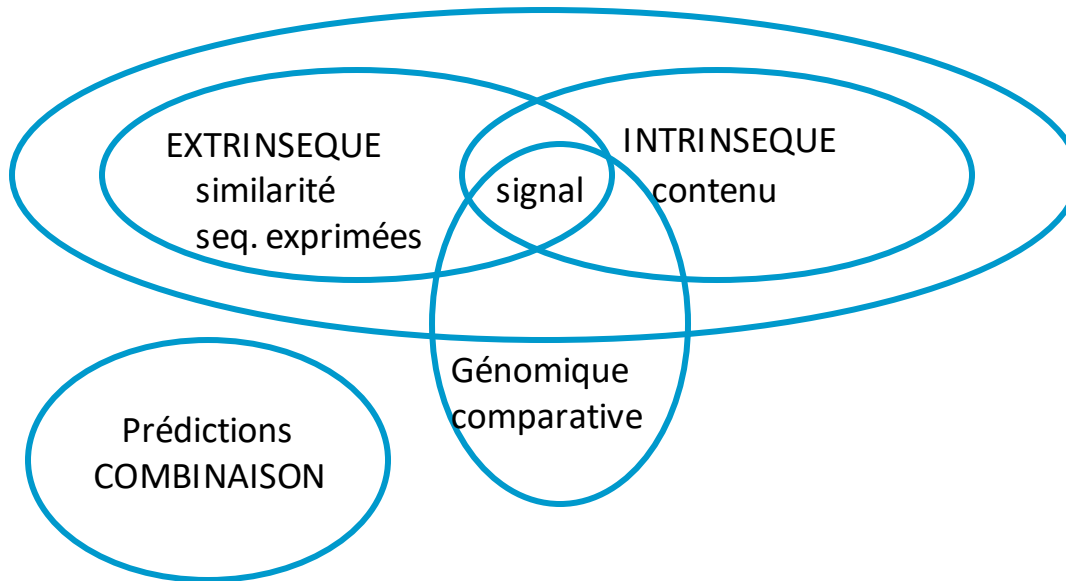


Augmentation des données d'expression, prise en compte de la similarité de séquences (Borodovsky et al., 1994, Fickett, 1995)

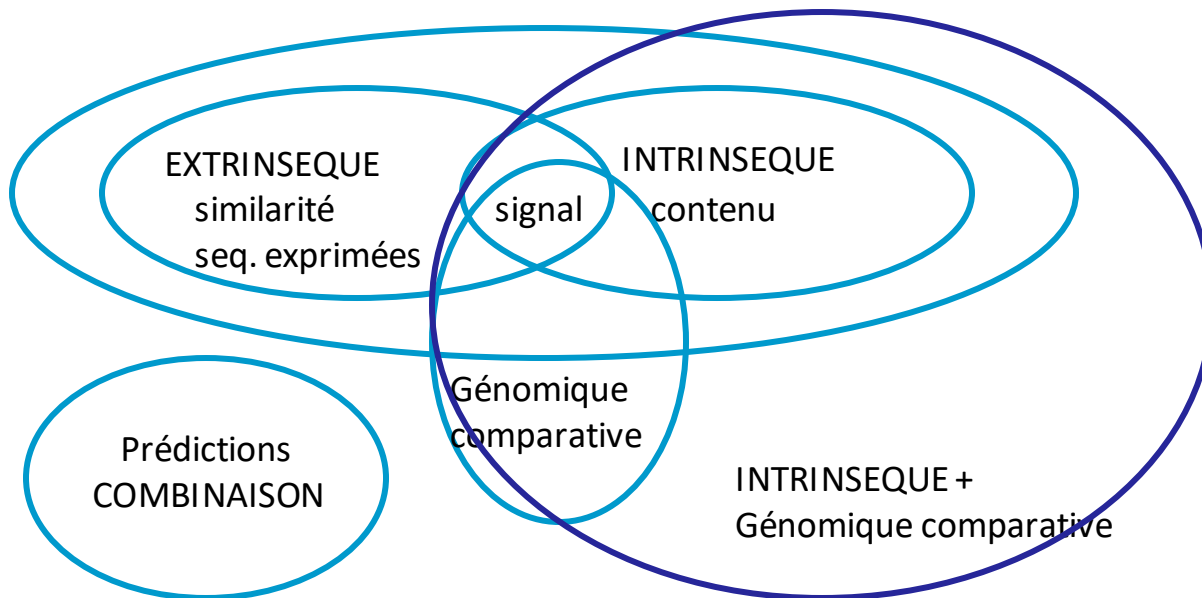
---



Intégration de ces deux types d'information dans de nombreux logiciels. GENOMESCAN (Yeh et al., 2001) résulte de l'intégration de similarité protéique dans GENSCAN



Conservation entre séquences génomiques exploitées en combinaison seulement avec des informations de type signal



Génomique comparative intégrée uniquement dans des méthodes intrinsèque. Par exemple, TWINSCAN (Korf et al., 2001) intègre la génomique comparative dans GENSCAN.

# Environnements intégrés pour l'annotation des génomes procaryotes



- ✓ **RAST server** (BMC Genomics. 2008 Feb 8;9:75. doi: 10.1186/1471-2164-9-75)

## DESCRIPTION:

We describe a fully automated service for annotating bacterial and archaeal genomes. The service identifies protein-encoding, rRNA and tRNA genes, assigns functions to the genes, predicts which subsystems are represented in the genome, uses this information to reconstruct the metabolic network and makes the output easily downloadable for the user (<https://rast.nmpdr.org/>)

- ✓ **Prokka** (Bioinformatics. 2014, 30:2068-69)

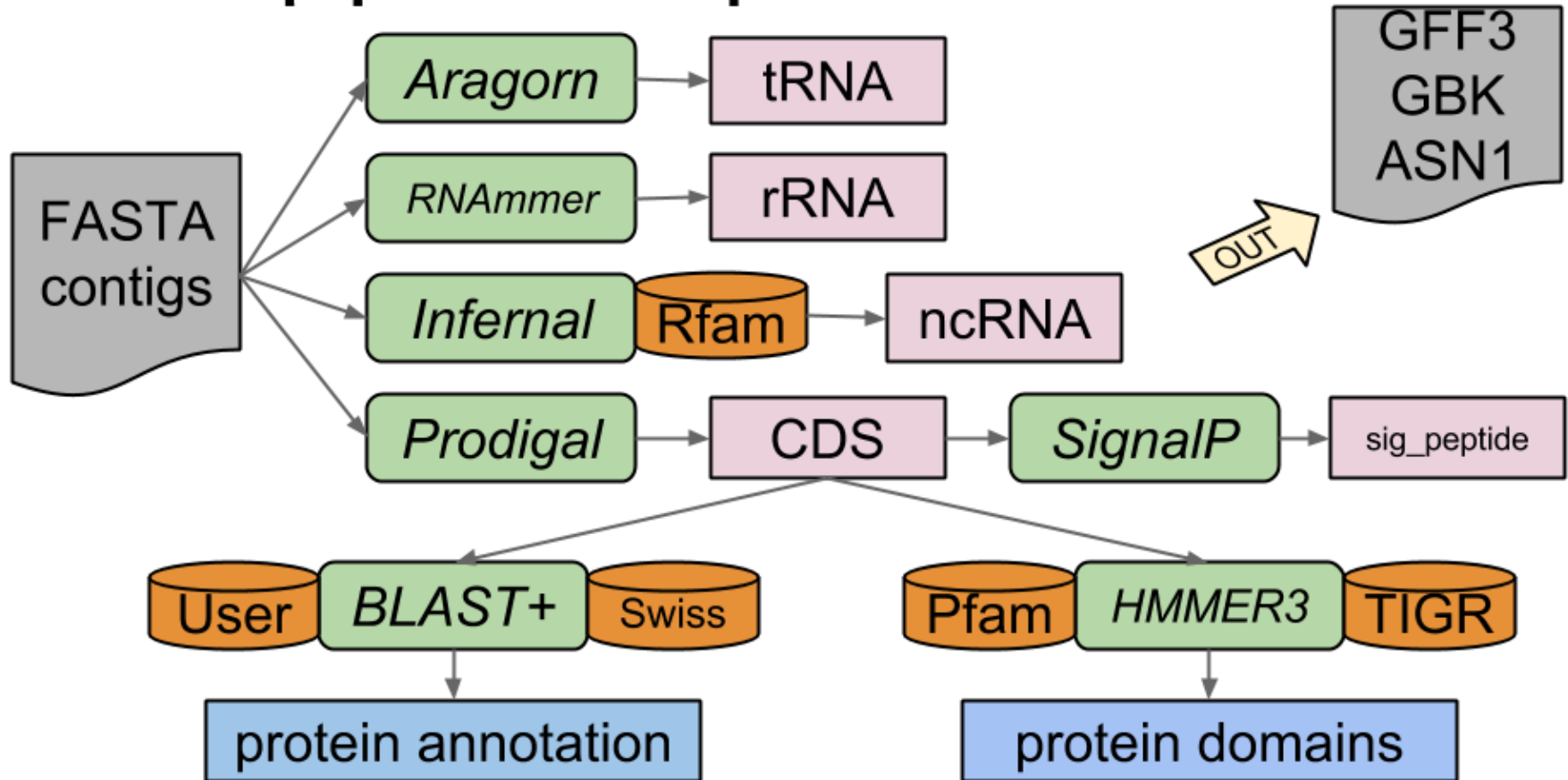
## DESCRIPTION:

Prokka, a command line software tool to fully annotate a draft bacterial genome in about 10min on a typical desktop computer. It produces standards-compliant output files for further analysis or viewing in genome browsers.

Tool (reference)	Features predicted
Prodigal (Hyatt 2010)	Coding sequence (CDS)
RNAmmer (Lagesen <i>et al.</i> , 2007)	Ribosomal RNA genes (rRNA)
Aragorn (Laslett and Canback, 2004)	Transfer RNA genes
SignalP (Petersen <i>et al.</i> , 2011)	Signal leader peptides
Infernal (Kolbe and Eddy, 2011)	Non-coding RNA

Méthodes utilisées dans Prokka  
(issu de Bioinformatics. 2014,  
30:2068-69)

# Prokka pipeline (simplified)



# Environnements intégrés pour l'annotation des génomes procaryotes



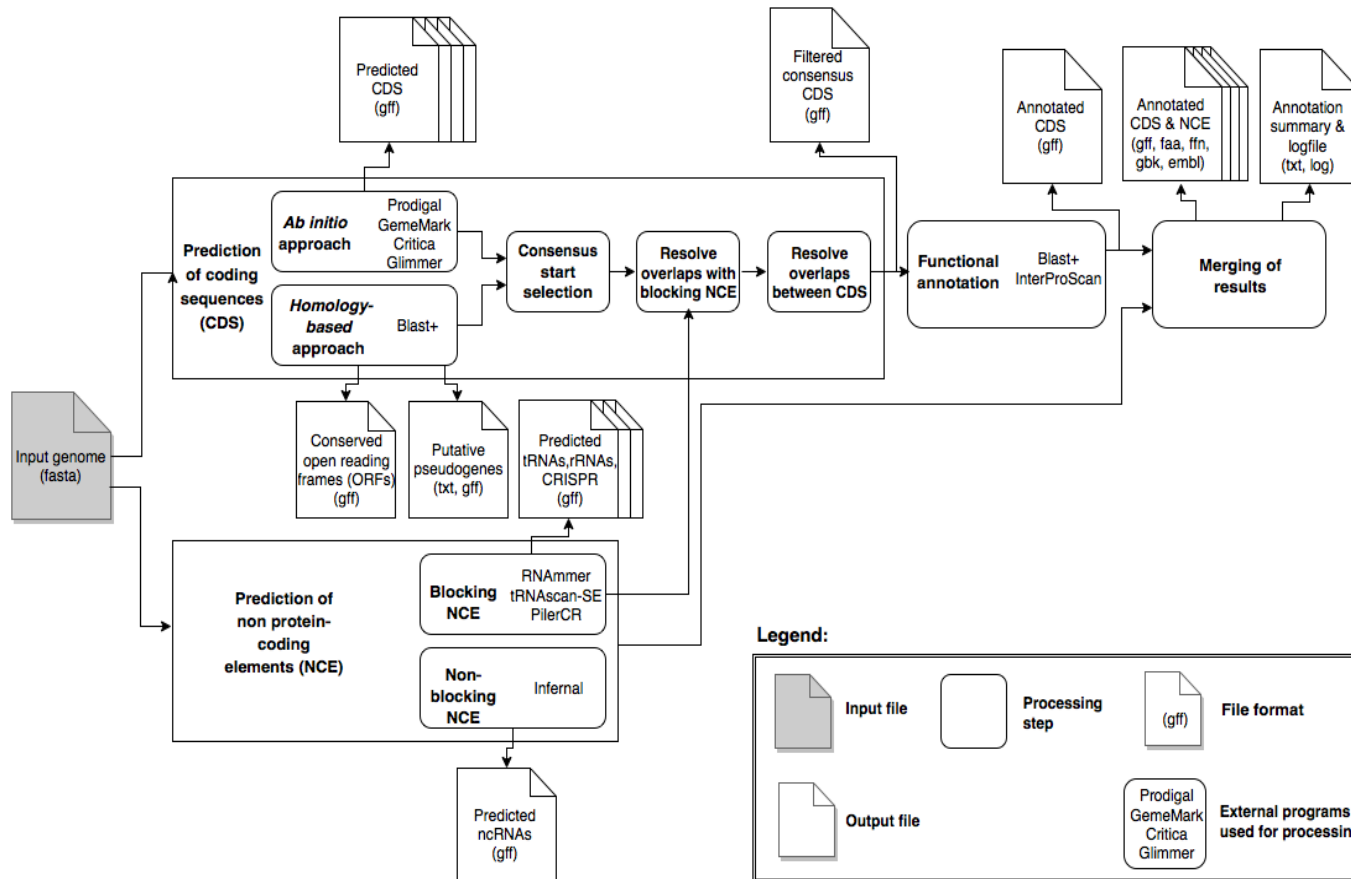
✓ **ConsPred** (Bioinformatics. 2016, 32:3327-29)

## DESCRIPTION:

We present ConsPred, a prokaryotic genome annotation framework that performs intrinsic gene predictions, homology searches, predictions of non-coding genes as well as CRISPR repeats and integrates all evidence into a consensus annotation. ConsPred achieves comprehensive, high-quality annotations based on rules and priorities, similar to decision-making in manual curation and avoids conflicting predictions. Parameters controlling the annotation process are configurable by the user.

# Environnements intégrés pour l'annotation des génomes procaryotes

## Exemple de ConsPred (Weinmaier *et al.*, 2016, Bioinformatics, 32:3327-29)



(Extrait de Weinmaier *et al.*)

**Figure S1.** ConsPred workflow

Coding sequences (CDS) are predicted by combining different *ab initio* gene predictions, and conserved open reading frames (ORFs) detected by homology search against the NCBI nr database. Database entries from closely related taxa are excluded to prevent possible misannotations due to low phylogenetic distance. Putative pseudogenes are exported for user inspection. From all predicted non-protein-coding elements (NCE) those that biologically must not overlap with CDS are considered blocking NCE. CDS overlapping with blocking NCE are removed. Filtered consensus CDS are obtained from predicted CDS and conserved ORFs by using predefined weights and rules and subsequent removal of CDS that overlap with blocking NCEs. Filtered consensus CDS are functionally annotated and then merged with the NCE into the final annotation files.



# Environnements intégrés pour l'annotation des génomes prokaryotes et eucaryotes

GenSAS v6.0

<https://www.gensas.org/>

Home

Use GenSAS

Available Tools

Available Libraries

New Features

Help

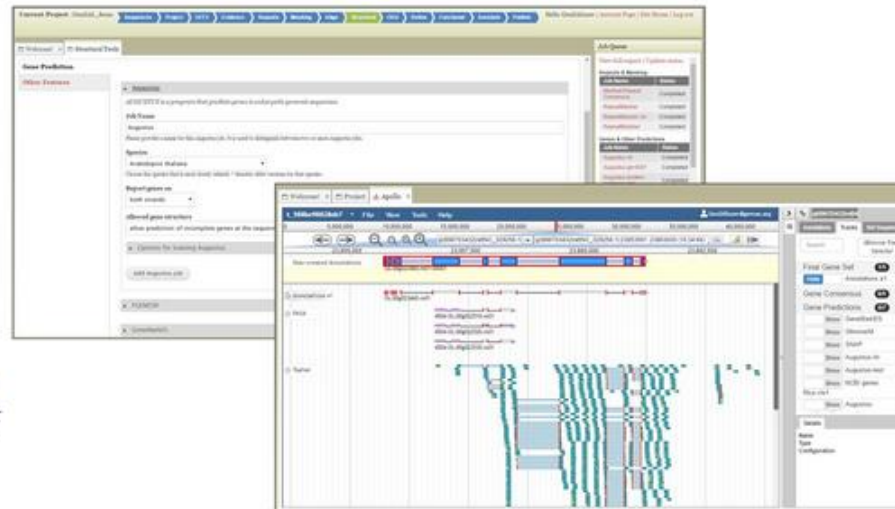
Cite GenSAS

GenS

JL Humann et al. "Structural and Functional Annotation of Eukaryotic Genomes with GenSAS." Gene Prediction: Methods and Protocols. Ed. Martin Kollmar. New York: Humana Press, 2019. 29-51.

## Welcome to GenSAS

The Genome Sequence Annotation Server (GenSAS) is an online platform that provides a pipeline for whole genome structural and functional annotation for eukaryotes and prokaryotes. Users can upload genome sequences and select from a variety of tools for repeat masking, prediction of gene models and other structural features as well as functional annotation tools. GenSAS integrates with **JBrowse** and **Apollo** to provide visualization and editing. For more details about GenSAS, please see our **User's Guide**.



# Méthodes disponibles dans GenSAS



Category	Tool Name	Version	Description
Repeat Masking	RepeatMasker	4.1.1	Masks repeated sequences
	RepeatModeler	2.0.1	<i>de novo</i> repeat masker
Gene Prediction	Augustus	3.4.0	Eukaryotic gene predictor
	BRAKER2	2.1.5	Eukaryotic gene predictor
	GeneMarkES	4.48	Eukaryotic gene predictor
	GeneMarkS	4.32	Prokaryotic gene predictor
	Genscan	1.0	Eukaryotic gene predictor
	Glimmer3	3.0.2	Prokaryotic gene predictor
	GlimmerM	2.5.1	Eukaryotic gene predictor
	SNAP	11/29/2013	Semi-HMM-based Nucleic Acid Parser, gene predictor
Alignments	BLAST+	2.12.0	Alignment tool for transcripts and proteins
	pBLAT	2.5	Alignment tool for transcripts
	DIAMOND	2.0.6	Alignment tool for protein evidence
	HISAT2	2.2.1	Aligns RNA-seq reads to genome
	PASA	2.4.1	Aligns EST or transcript evidence
	TopHat	2.1.1	Aligns RNA-seq reads to genome
Genomic Features	Getorf	EMBOSS-6.6.0	Open reading frame finder
	RNAmmer	1.2	Finds rRNA sequences
	SSR Finder	N/A	Finds user defined simple sequence repeats
	tRNAScan-SE	2.0.7	Identifies tRNA sequences
Consensus Gene Models	EVidenceModeler	1.1.1	Creates consensus gene structures for eukaryotes
Functional Annotation	BLAST+	2.12.0	Alignment tool for proteins
	DIAMOND	2.0.6	Alignment tool for proteins
	InterProScan	5.53-87.0	Classifies proteins by families and identifies functional domains
	Pfam	1.6	Uses Pfam database to identify functional domains and families in proteins
	SignalP	5.0	Predicts signal peptide cleavage sites in protein sequences
	TargetP	2.0	Predicts the subcellular location of eukaryotic proteins based on signal peptide sequences
Genome Browser	Apollo	2.0.7	Annotation editor
	JBrowse	1.12.1	Genome browser
Assembly/annotation metrics	BUSCO	5.2.2	Determines genome assembly and annotation completeness based on orthologs
	One code to find them all	1.0	Reports repeat types from RepeatMasker
	PRINSEQ-lite	0.20.4	Used to analyze assembly file for sequence number and size

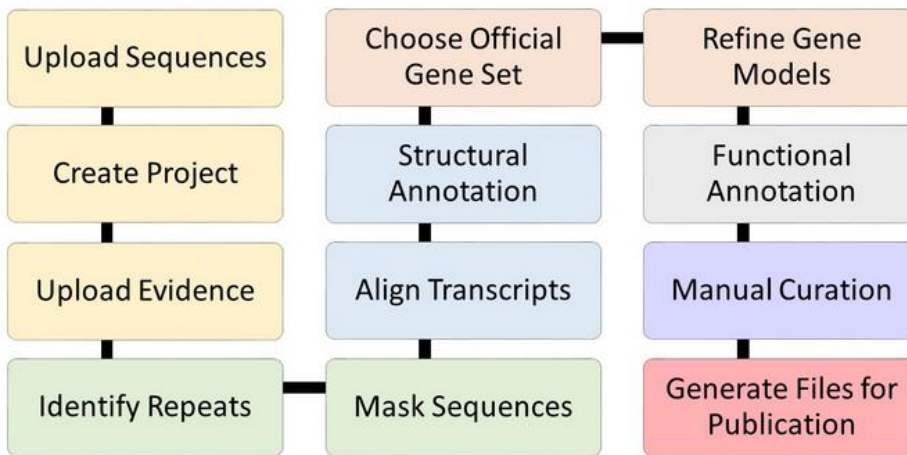
# Bases de données disponibles dans GenSAS

GenSAS provided library	RNA number	Protein number	Release date
refseq_archaea	1,354	2,936,792(nr)	5/4/2023
refseq_bacteria	28,459	210,034,561(nr)	5/4/2023
refseq_fungi	5,154,195	5,163,994	5/5/2023
refseq_invertebrate	9,708,846	8,686,644	5/4/2023
refseq_mitochondrion	NA	240,117	5/4/2023
refseq_plant	8,805,618	8,192,225	5/4/2023
refseq_plasmid	7	2,080,794(nr)	5/4/2023
refseq_plastid	44	1,064,130	5/5/2023
refseq_protozoa	1,083,003	1,164,421	5/4/2023
refseq Vertebrate_mammalian	12,211,560	10,215,066	5/4/2023
refseq Vertebrate_other	15,510,388	13,672,291	5/4/2023
refseq_viral	NA	655,246	5/4/2023
sprot	NA	569,793	6/28/2023
trembl	NA	248,272,897	6/28/2023
RepBase library	NA	NA	12/24/2018

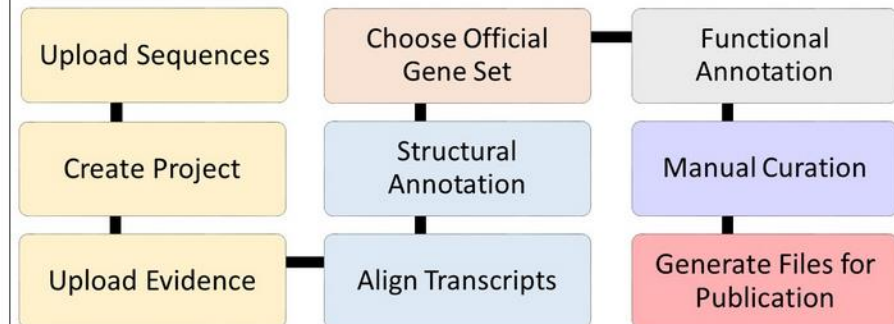
The steps on the flowcharts with the same color represent the major steps of annotation:

1. Project creation/uploading sequences and evidence
2. Repeat identification and masking (for eukaryotes)
3. Structural annotation
4. Selection of official gene set (and refinement for eukaryotes)
5. Functional annotation
6. Manual curation of the annotation
7. Preparing files for publication

### GenSAS eukaryotic annotation process



### GenSAS prokaryotic annotation process



# Environnement pour des génomes eucaryotes

## Ensembl (Howe *et al.*, *Nucleic Acids Res.* 2021, vol. 49(1):884–891)

### Tools

[All tools](#)

### BioMart >

Export custom datasets from Ensembl with this data-mining tool

### BLAST/BLAT >

Search our genomes for your DNA or protein sequence

### Variant Effect Predictor >

Analyse your own variants and predict the functional consequences of known and unknown variants

### Search

All species  for

Go

e.g. BRCA2 or rat 5:62797383-63627669 or rs699 or coronary heart disease

### All genomes

-- Select a species --



### Pig breeds

Pig reference genome and 12 additional breeds

[View full list of all species](#)

### Favourite genomes



### Human

GRCCh38.p13

[Still using GRCh37?](#)



### Mouse

GRCm39



### Zebrafish

GRCz11

Ensembl is a genome browser for vertebrate genomes that supports research in comparative genomics, evolution, sequence variation and transcriptional regulation. Ensembl annotate genes, computes multiple alignments, predicts regulatory function and collects disease data. Ensembl tools include BLAST, BLAT, BioMart and the Variant Effect Predictor (VEP) for all supported species.

### Ensembl Release 104 (May 2021)

- Update to the Ensembl Canonical transcript set.
- Human and mouse gene sets updated to GENCODE 38 and GENCODE M27, respectively.
- Retirement of gene names derived from BAC clones.

[More release news](#) on our blog

### Ensembl Rapid Release

New assemblies with gene and protein annotation every two weeks.

Note: species that already exist on this site will continue to be updated with the full range of annotations.

Go

The Ensembl Rapid Release website provides annotation for recently produced, publicly available vertebrate and non-vertebrate genomes from biodiversity initiatives such as Darwin Tree of Life, the Vertebrate Genomes Project and the Earth BioGenome Project.

[Rapid Release news](#) on our blog

### Other news from our blog

- 23 Aug 2021: [Job: Outreach Officer](#)
- 16 Aug 2021: [Free Ensembl Browser and REST API virtual workshops in September](#)
- 10 Aug 2021: [Job: Bioinformatician](#)

Compare genes across species



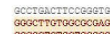
Find SNPs and other variants for my gene



Gene expression in different tissues



Retrieve gene sequence



Find a Data Display



Use my own data in Ensembl



<https://www.ensembl.org/index.html>

C'est un projet conjoint de l'European Bioinformatics Institute (EBI) et du Wellcome Trust Sanger Institute dont l'idée centrale est d'organiser de vastes champs d'information biologique autour de séquences génomiques.

# Diagramme des étapes d'annotation de CDS dans Ensembl

Targetted stage: species-specific proteins aligned to the genome to produce transcript models

Similarity stage: proteins from closely-related species aligned to the genome to produce transcript models

Species-specific cDNA and ESTs aligned to the genome to produce transcript models and add UTRs to protein-derived models

RNA-seq reads aligned to the genome to produce transcript models and add UTRs to protein-derived models

Multi-transcript genes produced by merging where exons match

species-specific proteins are aligned to the genome and Genewise and/or Exonerate is used to build a transcript structure for the protein on the genome.

proteins from closely related species are used to build transcript structure in regions where a Targeted transcript structure is absent. For species with fewer species-specific protein sequences the Similarity stage plays an important role in predicting gene structures

Alignment of species-specific cDNA and EST sequences to the genome. Where cDNA alignments overlap transcripts predicted in the preceding stages, any non-translated region from the cDNA is spliced onto the transcript prediction as UTR.

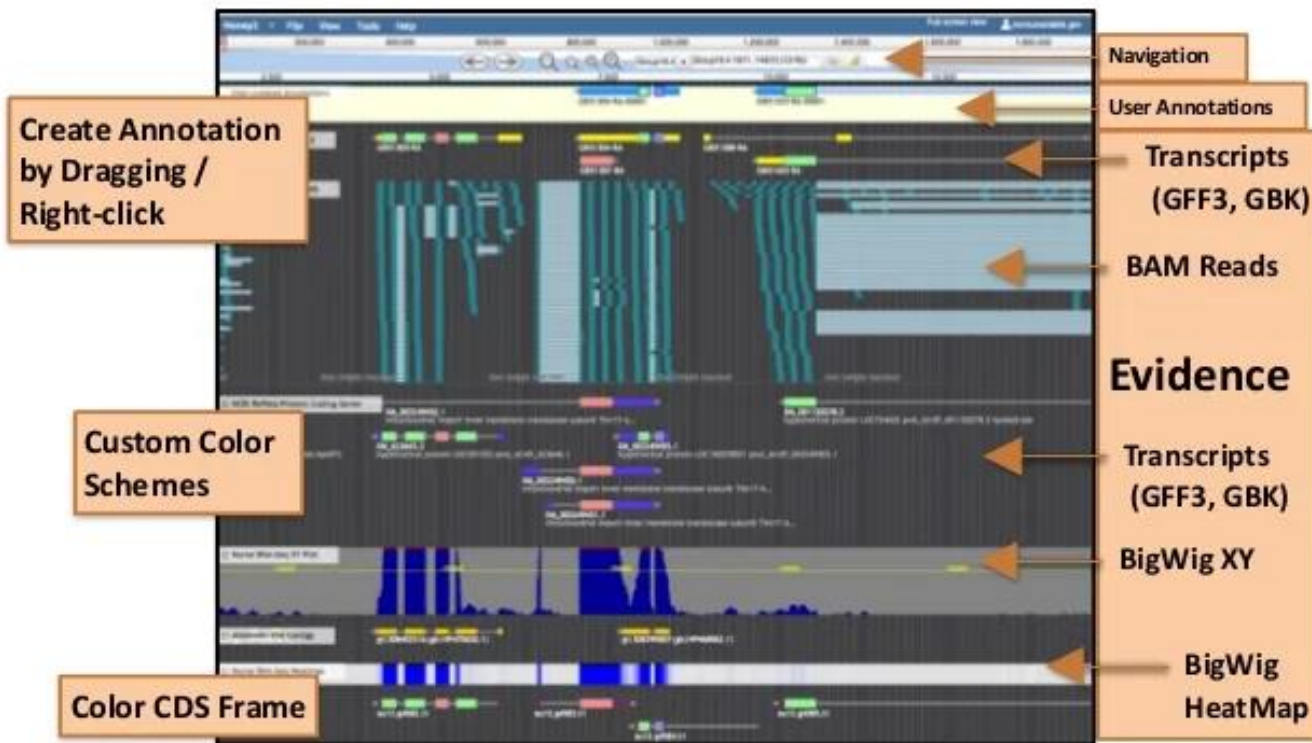
Where available, RNA-seq reads are aligned to the genome and build RNA-seq-based transcript models from these data. These models may be used to add new genes or transcript isoforms into the gene set, and to add UTR to protein-coding models.

The final set: merging identical transcripts built from different proteins sequences to produce multi-transcript gene predictions, each with a non-redundant set of transcripts models. Where transcripts have identical exons, these are combined to form a gene



# Apollo: genome annotation editor

## Apollo Genome Editor



Référence : Dunn NA, Unni DR, Diesh C, Munoz-Torres M, Harris NL, Yao E, et al. (2019) Apollo: Democratizing genome annotation. PLoS Comput Biol 15(2): e1006790. <https://doi.org/10.1371/journal.pcbi.1006790>

Documentation : <https://genomearchitect.readthedocs.io/en/latest/>