Supplementary Material

In this document, we give some details about the implementation of AUGUSTUS-PPX, and the full set of tables of the results, for the interested reader. The first section explains in detail how block hits are determined in the course of the algorithm, the second deals with performance issues.

Identifying block hits

To classify a given protein sequence s of length w as a *block hit*, we turn the scoring function into a decision function by requiring $\rho(s) > \tau$, with a block-specific threshold $\tau = \tau^{(b)}$. This section describes how τ is determined.

Two global parameters $\theta_0 (= \theta_{\text{spec}})$ and $\theta_1 (= \theta_{\text{sens}})$ independent from b, are specified by the user, designed to ensure that estimated error rates are low.

We consider two competing models, H_0 describing random sequences distributed according to P_{back}, and H_1 describing block motifs from a block b under consideration, distributed according to $P_{\text{block}} = (P_i)_{i=0,\ldots,w}$, the frequency matrix given in the profile. $P_i(a)$ denotes the probability to observe amino acid a at position i of a block motif, while $P_{\text{back}}(a)$ denotes the global probability for a to appear anywhere. The odds-ratio is given by $R_i(a) = \frac{P_i(a)}{P_{\text{back}}(a)}$; for convenience, we consider in the following the log-odds ratio $L_i(a) = \log R_i(a)$, turning the product $\rho(s) = R_0(s_0) \cdot \ldots \cdot R_{w-1}(s_{w-1})$ into a sum $\ell(s) = \log \rho(s) = L_0(s_0) + \ldots + L_{w-1}(s_{w-1})$. Each of the two models H_j gives rise to a different expectation value $\mu_j = \mathrm{E}_j(L)$ and variance $\sigma_j^2 = \mathrm{Var}_j(L)$ for the log-scores.

By putting the global parameters into the block-specific scale, we obtain thresholds $\tau^- = \mu_0 + \theta_0\,\sigma_0$, $\tau^+ = \mu_1 - \theta_1\,\sigma_1$. A score exceeding τ^- is *at least* θ_0 standard deviations above the expected score in the background model, and a score below τ^- is *at most* θ_1 standard deviations less than the expected score for a block motif. The probability for a random sequence (in either model) to have a score in this range can be approximated with the Gaussian distribution by

$$1 - \Phi(\theta_0) \qquad \text{and} \qquad 1 - \Phi(\theta_1),$$

where Φ is the cumulative Gaussian distribution function; hence, these numbers are bounding the estimated error rates.

For example, the values $\theta_0 = 4.5$ and $\theta_1 = 1.5$ used in the test runs correspond to a false-positive rate less than $1 - \Phi(4.5) = 3.3 \cdot 10^{-6}$ (one block hit in a random amino acid sequence of 300 000 residues), and to a sensitivity of at least $\Phi(1.5) = 93.3\%$.

In order to fulfill both conditions, τ must satisfy $\tau^- \leq \tau \leq \tau^+$. Provided that $\tau^- \leq \tau^+$, we set $\tau = \frac{1}{2}(\tau^- + \tau^+)$. Any sequence satisfying the condition $\ell^{(b)}(s) > \tau$, or equivalently, $\rho^{(b)}(s) > \exp(\tau)$, is then considered a block hit. In the case $\tau^- > \tau^+$, or if $w < 6$, the block is removed from the profile for the evaluation.

Performance issues of the modified Viterbi algorithm

An exhaustive evaluation of all combinations of block locations and substates would be computationally infeasible, causing an explosion of running time and memory requirements; therefore, several techniques are applied that eliminate most of the substate entries in order to control the computational cost. We store substate scores dynamically, reserving memory only for nonzero entries.

Precomputing block hit collections

To prepare the search for whole blocks found on the same exon, the target sequence is searched for block hits in parallel to the main algorithm, and the hits are stored in collections of consecutive hits satisfying the distance conditions. As mentioned in the article, motifs that do not score over the threshold are not considered for the collections. In practice, this commonly leads to exon candidates with very few block hits or none at all allowed on them, preventing the vast majority of substate entries to be created.

Scores for blocks truncated at exon borders are calculated only once for every location. Furthermore, similar to the case of full blocks, truncated blocks are subject to filtering with thresholds if they exceed a minimum length; again, this leaves only few values for $i > 0$ actually stored as substates.

Removing dominated substates

In the situation of Figure 1 of the article, an inter-block substate (b, i) constrains the admissible block start on the upcoming exon(s).The substate is *dominated* by two substates (b, i') and (b, i'') if they cover the same set of admissible block starts. This is the case if $i' < i < i''$ and $i'' - i' \leq d_b^{\max} - d_b^{\min}$. If both of the dominating substates have a higher score, the entry at (b, i) may be deleted from the Viterbi table.

Pruning dead state graph branches

In order to reduce the memory needed for the inflated Viterbi table, we remove from it all entries that are not contained in any parse reaching the current DNA location. To this end, for each substate entry, a counter is installed and incremented every time the entry is maximizing the partial score for some successor state. A substate entry is deleted if its successor count is zero at the time the algorithm has progressed to a location in the target sequence beyond the maximal state length from the entry. Analysis showed that about half of the memory usage could be saved this way.

Sharing substate tables

As the profile location is constant throughout an intron, memory can be saved by sharing substate tables between consecutive intron states with a fixed length. In particular, AUGUSTUS' state model has intron states emitting a single nucleotide, and candidate parses evaluated in the course of the Viterbi algorithm contain long sequences of the single-nucleotide intron states in every long intron, mostly with identical substate entries just differing by a constant factor. Instead of using a separate copy for the substate table for each nucleotide position, only the constant factor is stored, and a link to the substate table of the predecessor (the last DNA location that an exon candidate contributing substates was considered).

Setup and Results

Reference gene sets were created my mapping the reference protein sequences to the genomes, using Scipio (Keller et al. 2008). In some cases, parts of the queries could not be mapped (see Table 2); we believe it is safe to assume that the corresponding genomic sequence is missing due to incomplete assemblies, and that the reference gene structures comprise all true exons present in the genome.

The runnings were performed on the genomic regions starting 20 Kbps before and ending 20 Kbps after the reference genes, but at least covering 200 Kbps.

Table 1. Genome resources and versions

species	full name	UCSC version	source (version)	NCBI project id RefSeq	NCBI project id assembly	assembly stage
human	*Homo sapiens*	hg19	GRC (37)	168	13178	chrom.
mouse	*Mus musculus*	mm9	GRC (37)	169	13183	chrom.
chicken	*Gallus gallus*	galGal3	WashU (2.1)	10808	13342	chrom.
frog	*Xenopus tropicalis*	xenTro2	JGI (4.1)	43581	12348	scaffolds
zebrafish	*Danio rerio*	N/A	Sanger (Zv9)	11776	13922	chrom.
snail	*Lottia gigantea*	N/A	JGI (1.0)	N/A	N/A	scaffolds

Version Zv9 of the *D. rerio* genome was downloaded from Ensembl:
`ftp://ftp.ensembl.org/pub/assembly/zebrafish/Zv9release`
The *L. gigantea* genome was downloaded from JGI:
`ftp://ftp.jgi-psf.org/pub/JGI_data/Lottia_gigantea/v1.0`
The other sequences were downloaded from UCSC:
`ftp://hgdownload.cse.ucsc.edu/goldenPath/<version>/bigZips`

Table 2. Genomic reference sequences used in DHC runs

species	# of genes	# of exons	# of bps	av. prot. length	assembly qual. (%)	av. gene len (Kbps)
human	16	1213	211788	4411.25	100.0	242.46
mouse	18	1338	235608	4368.89	99.8	206.84
chicken	12	891	152727	4363.67	97.2	93.88
frog	18	1275	218871	4367.33	92.8	143.05
zebrafish	14	963	163788	4293.29	90.8	127.65
snail	18	1358	241047	4518.39	98.8	44.69
total	96	7038	1223829	4392.01	96.7	144.75

Overview of the reference sequences used in the DHC runs. Shown as assembly quality is the percentage of the transcript that could be found in the genome.

Results of test runs

All inputfiles used in the runs, and the original output files can be downloaded from `http://augustus.gobics.de`. All output was generated with AUGUSTUS version 2.5.

Table 3. Accuracy at nucleotide level of DHC runs

species	AUGUSTUS-PPX			AUG.	Scipio	Genewise	
	full	fbs	ex-ortho	ab-initio	cross-sp.	full	ex-ortho
sensitivity (%)							
human	96.2	96.8	95.8	90.3	100.0	97.9	68.6
mouse	97.2	97.5	94.7	90.9	95.2	96.2	69.3
chicken	95.7	95.7	94.8	87.7	85.4	83.2	65.0
frog	93.2	92.6	92.0	85.1	68.7	82.4	66.2
zebrafish	96.1	96.1	95.2	89.4	70.8	86.3	66.0
snail	93.7	93.5	93.2	86.3	52.9	76.5	69.6
total	95.3	95.3	94.2	88.2	78.5	87.2	67.7
specificity (%)							
human	96.8	97.5	97.0	94.2	100.0	96.7	89.0
mouse	90.3	90.7	88.6	76.6	99.9	95.9	86.1
chicken	94.2	94.6	93.6	89.5	99.3	92.9	86.7
frog	86.9	88.2	86.0	82.0	99.4	92.2	85.5
zebrafish	90.3	90.2	88.9	78.7	99.1	90.1	86.4
snail	94.7	95.6	94.0	86.3	99.5	95.4	93.4
total	92.1	92.7	91.1	84.0	99.6	94.2	88.0

Table 4. Accuracy at exon level of DHC runs

species	AUGUSTUS-PPX			AUG.	Scipio	Genewise	
	full	fbs	ex-ortho	ab-initio	cross-sp.	full	ex-ortho
sensitivity (%)							
human	93.3	93.6	91.0	84.5	N/A	83.7	51.8
mouse	93.6	93.7	90.5	85.1	86.0	80.4	50.1
chicken	89.3	89.8	87.5	80.5	66.0	68.8	46.2
frog	85.8	86.0	83.8	78.0	48.2	64.3	48.0
zebrafish	89.8	89.8	88.0	80.3	49.2	68.5	49.1
snail	85.9	86.5	84.3	78.9	31.6	59.3	49.1
total	89.6	89.9	87.5	81.3	63.5	70.9	49.2
specificity (%)							
human	91.7	93.4	90.6	88.1	N/A	84.9	71.9
mouse	90.3	91.1	86.9	80.1	87.9	82.8	69.2
chicken	87.0	87.6	84.1	80.3	73.7	80.6	70.1
frog	75.9	80.5	74.7	77.9	64.6	74.3	67.8
zebrafish	82.1	82.8	79.5	77.1	64.0	75.9	70.2
snail	84.3	87.6	82.5	78.6	51.9	75.9	68.8
total	85.0	87.2	82.9	80.3	76.5	79.3	69.6

Table 5. Accuracy at gene level of DHC runs

species	AUGUSTUS-PPX			AUG.	Scipio	Genewise	
	full	fbs	ex-ortho	ab-initio	cross-sp.	full	ex-ortho
highly accurate at 95/85 (%)							
human	62.5	68.8	62.5	31.3	N/A	93.8	6.3
mouse	72.2	77.8	61.1	22.2	88.9	55.6	0.0
chicken	58.3	58.3	50.0	8.3	8.3	16.7	0.0
frog	44.4	44.4	38.9	0.0	5.6	11.1	0.0
zebrafish	57.1	57.1	35.7	57.1	7.1	14.3	0.0
snail	44.4	50.0	44.4	5.6	0.0	11.1	0.0
total	56.3	59.4	49.0	11.5	36.5	34.4	1.0
highly accurate at 90/90 (%)							
human	87.5	93.8	81.3	25.0	0.0	93.8	6.3
mouse	72.2	77.8	66.7	27.8	88.9	88.9	0.0
chicken	83.3	83.3	75.0	16.7	33.3	41.7	0.0
frog	50.0	66.7	38.9	11.1	5.6	50.0	0.0
zebrafish	57.1	57.1	50.0	14.3	14.3	35.7	0.0
snail	77.8	83.3	72.2	22.2	0.0	38.9	16.7
total	70.8	77.1	63.5	19.8	24.0	59.4	4.2
identified genes (%)							
human	100.0	100.0	93.8		N/A	100.0	100.0
mouse	100.0	100.0	94.4		100.0	100.0	88.9
chicken	100.0	100.0	100.0		100.0	91.7	83.3
frog	94.4	88.9	94.4		88.9	88.9	83.3
zebrafish	92.9	92.9	92.9		92.9	92.9	78.6
snail	88.9	83.3	88.9		83.3	88.9	88.9
total	95.8	93.8	93.8		95.8	93.8	87.5

Percentage of genes predicted at high accuracy at *sens*/*spec* level. A gene is considered highly accurate if sequence overlap of prediction and reference is at least *sens*% of reference and *spec*% of prediction; overlap is measured in % of bps. Below the rate of genes identified as DHCs (not relevant for AUGUSTUS ab-initio).

Table 6. Overview of PFAM families used

family	accession	# of seqs in database	# of core seqs used	# of seqs (human)	# of core seqs (human)
HSP70	PF00012	9069	2374	14	8
Aldedh	PF00171	16528	3669	19	9
AA_permease	PF00324	13073	5185	24	15
Cullin	PF00888	924	489	10	7
Sec1	PF00995	917	349	8	7

Table 7. Genomic reference sequences used in PFAM runs

family	# of genes	# of exons	# of bps	av. prot. length	assembly qual. (%)	av. gene len (Kbps)
HSP70	8	38	15576	648.00	100.0	4.63
Aldedh	9	121	16203	599.11	100.0	40.59
AA_permease	15	205	32520	721.67	100.0	47.55
Cullin	7	130	16710	794.71	100.0	64.67
Sec1	7	132	12609	600.57	99.8	56.90
total	46	626	93618	677.57	100.0	42.75

Table 8. Accuracy at nucleotide level of PFAM runs

family	AUGUSTUS-PPX			AUGUSTUS	Genewise
	full	<80%	<60%	ab-initio	
sensitivity (%)					
HSP70	98.8	98.8	98.8	97.7	68.8
Aldedh	97.2	96.7	96.1	95.2	58.4
AA_permease	93.0	92.5	93.1	90.8	44.5
Cullin	93.1	93.0	93.0	88.5	73.6
Sec1	95.3	95.3	95.9	90.7	70.7
total	95.0	94.7	94.9	92.3	59.7
specificity (%)					
HSP70	97.9	97.9	97.7	90.7	99.6
Aldedh	87.7	86.9	85.9	87.5	98.2
AA_permease	87.4	87.3	87.5	75.7	84.5
Cullin	90.9	90.1	90.1	90.5	95.8
Sec1	82.1	82.1	83.5	81.3	87.4
total	88.9	88.6	88.7	83.2	92.3

Table 9. Accuracy at exon level of PFAM runs

family	AUGUSTUS-PPX			AUGUSTUS	Genewise
	full	<80%	<60%	ab-initio	
sensitivity (%)					
HSP70	92.1	92.1	89.5	84.2	10.5
AldeDh	92.6	91.7	90.1	86.0	48.8
AA_perm	83.9	83.9	85.4	81.0	24.4
Cullin	87.7	86.9	86.9	83.1	56.9
Sec1	93.9	93.9	94.7	87.1	44.7
total	89.0	88.7	88.8	83.9	39.3
specificity (%)					
HSP70	92.1	92.1	89.5	71.1	25.0
AldeDh	76.7	75.5	71.2	77.0	76.6
AA_perm	71.7	71.4	72.6	65.4	40.3
Cullin	90.5	85.6	85.6	89.3	67.9
Sec1	87.3	87.3	88.7	83.9	59.0
total	80.5	79.3	78.9	75.9	57.7

Table 10. Accuracy at gene level of PFAM runs

family	AUGUSTUS-PPX			AUGUSTUS	Genewise
	full	<80%	<60%	ab-initio	
highly accurate genes at 95/85 (%)					
HSP70	87.5	87.5	87.5	75.0	0.0
Aldedh	77.8	66.7	55.6	66.7	0.0
AA_permease	53.3	53.3	53.3	33.3	0.0
Cullin	28.6	28.6	28.6	0.0	0.0
Sec1	28.6	28.6	42.9	14.3	0.0
total	56.5	54.3	54.3	39.1	0.0
highly accurate genes at 90/90 (%)					
HSP70	87.5	87.5	87.5	75.0	75.0
Aldedh	88.9	77.8	55.6	55.6	11.1
AA_permease	53.3	53.3	60.0	40.0	0.0
Cullin	57.1	57.1	57.1	14.3	0.0
Sec1	42.9	42.9	57.1	42.9	14.3
total	65.2	63.0	63.0	45.7	17.4
completely correct genes (%)					
HSP70	75.0	75.0	75.0	62.5	0.0
Aldedh	44.4	44.4	33.3	22.2	0.0
AA_permease	20.0	20.0	20.0	13.3	0.0
Cullin	0.0	0.0	0.0	0.0	0.0
Sec1	14.3	14.3	14.3	14.3	0.0
total	30.4	30.4	28.3	21.7	0.0
identified genes (%)					
HSP70	100.0	100.0	87.5		50.0
Aldedh	100.0	100.0	100.0		77.8
AA_permease	93.3	93.3	86.7		86.7
Cullin	100.0	100.0	100.0		100.0
Sec1	100.0	100.0	100.0		85.7
total	97.8	97.8	93.5		80.4

Percentage of genes predicted at high accuracy at *sens*/*spec* level, cf. Table 5. Gene level sensitivity (accuracy 100/100) is shown as well. Below the rate of genes identified as members of their families (not relevant for AUGUSTUS ab-initio).