Chapter 2

Alien_Hunter algorithm

2.1 Introduction

There is a growing literature on the detection of Horizontal Gene Transfer (HGT) events by means of compositional-based, non-comparative methods (Garcia-Vallve *et al.*, 2003; Hsiao *et al.*, 2003; Karlin, 2001; Lio and Vannucci, 2000; Merkl, 2004; Sandberg *et al.*, 2001; Tsirigos and Rigoutsos, 2005). Such approaches rely only on sequence information and utilize different low (e.g. G+C% content) or high order (e.g. 8mers) indices to capture deviation from the genome backbone composition. The superiority of high order over lower order indices, in detecting local compositional bias, has been shown previously (Sandberg *et al.*, 2001; Tsirigos and Rigoutsos, 2005).

More specifically Tsirigos *et al.* (Tsirigos and Rigoutsos, 2005) simulated HGT events by inserting (*in silico*) genes from a gene pool into a query genome and analyzed the sensitivity of increasing order indices in predicting the simulated, manually inserted genes. Overall, using low order indices (e.g. single-nucleotides or di-nucleotides), 40-55% of the manually inserted genes were correctly predicted as HGT events, whereas higher order indices e.g. 8mers showed overall a much higher sensitivity, predicting correctly 50-65% of those genes, depending on the number of simulated HGT events. Another example showing the increased sensitivity of high order indices is shown in Figure 2.1; two compositionally very similar sequences (seq1 and seq2) can only be predicted as compositionally distinct, if indices of order two (i.e. tri-nucleotides) or higher are exploited, while zero or first order compositional analysis predicts those two sequences to be compositionally identical. However, given the increased dimensionality of the compositional alphabet, in a fixed-order based implementation of compositional distributions even high order indices may actually be poor estimators of the local sequence composition; this is likely

to be the case when insufficient information is available, e.g. in short sequence samples or sliding windows, or when local, low-order compositional biases exist (Figure 2.2). Consequently methods exploiting multiple, different order indices can be more powerful in detecting compositional biases at various levels (Karlin, 2001).

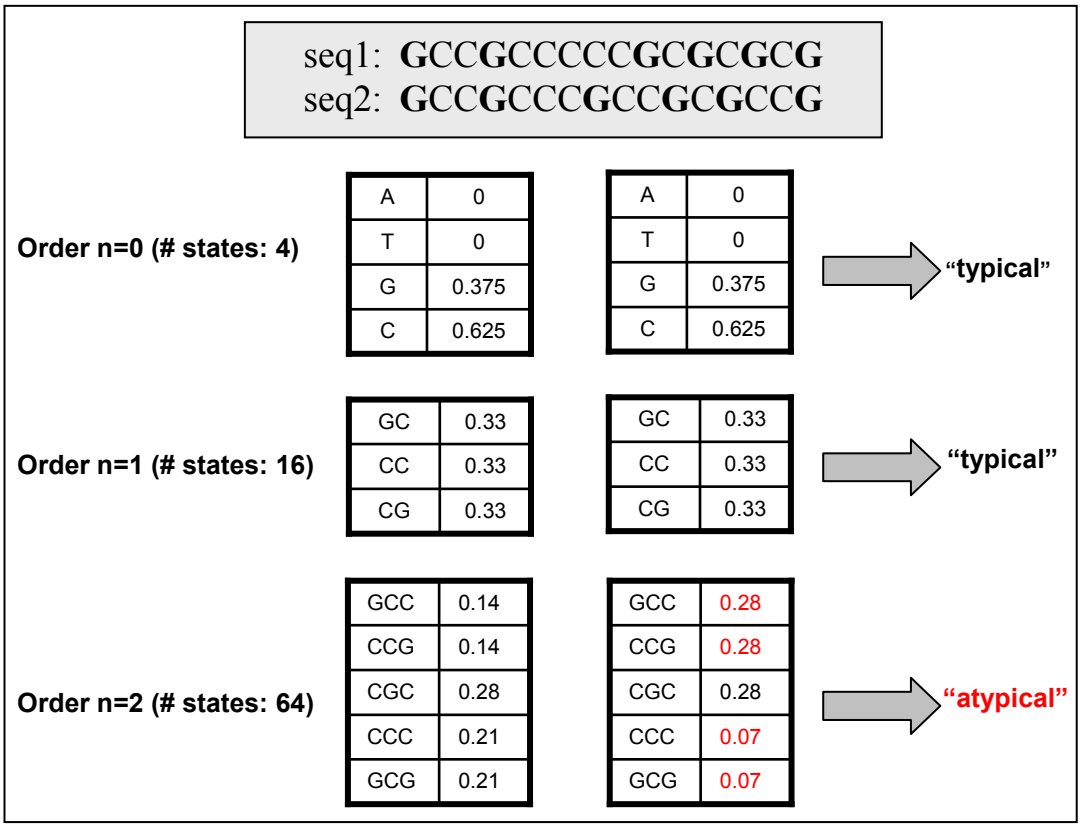

Figure 2.1: An example of two different sequences (seq1 and seq2) and the discrimination efficiency of increasing order indices. Only second (or higher) order indices can discriminate the two sequences as compositionally distinct.

In this chapter I describe a novel algorithm for the prediction of putative horizontally transferred regions by means of variable order compositional distributions with the aim of overcoming the limitations of fixed-order compositional approaches and exploiting the advantages of both low order (small-alphabet) and high order (increased sensitivity) compositional indices. This approach does not require pre-existing annotation (e.g. gene prediction), and can therefore be applied directly to newly sequenced genomes and used as a supplementary tool in the

annotation pipelines. Moreover I discuss the application of two different methods for determining a genome-specific score threshold, as well as the implementation of region specific two-state, second-order Hidden Markov Models (HMMs) to optimize the localization of the boundaries of the predicted regions. Finally I describe the pipeline followed to obtain a test dataset of manually curated putative horizontally transferred regions, the performance benchmarking against other, existing methods and the biological significance of the *in silico* predictions.

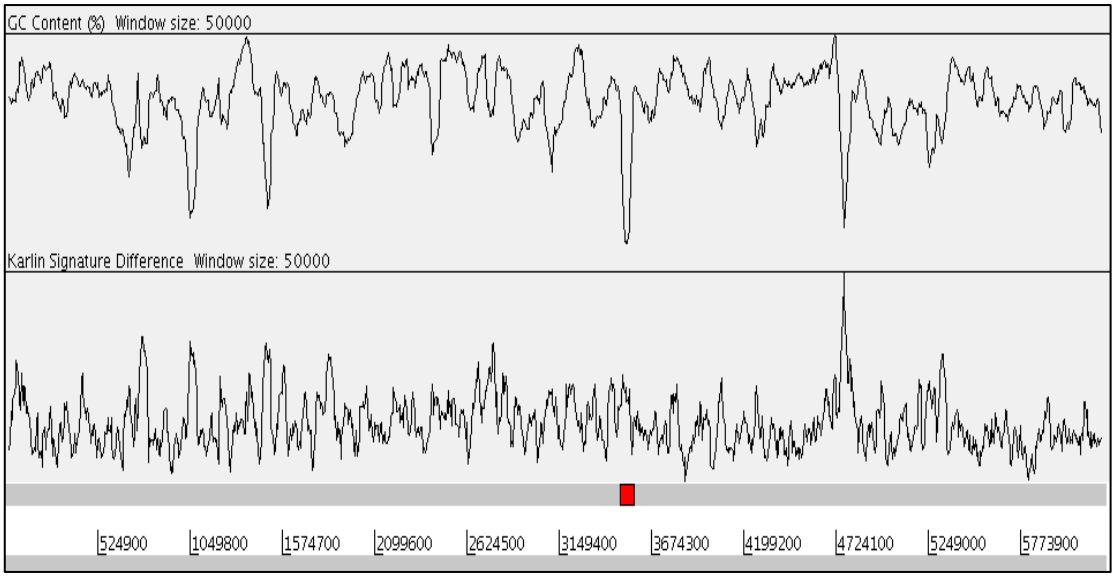

Figure 2.2: *Pseudomonas aeruginosa* PAO1 genome. The G+C% content and the di-nucleotide signature δ^* (Karlin, 2001) have been plotted genome-wide, with a window size of 50kb. A region carrying CDSs encoding products involved in the lipopolysaccharide (LPS) biosynthesis is shown as a red coloured feature. The LPS region deviates from the backbone composition mainly due to low order compositional bias (enriched in Adenine and Thymine); a compositional bias not captured by higher order indices e.g. di-nucleotides.

2.2 Methods

2.2.1 Interpolated Variable Order Motifs

Usage of low order compositional indices may not provide sufficient discrimination of regions with atypical high order (e.g. 6mers) composition. The total number of all different possible motifs (or indices) increases exponentially with the size k of the motifs. For k-mers of size k

(e.g. k=6) there are 4^k (e.g. 4096) different possible k-mers (parameters). Consequently, because a much higher number of parameters are exploited, utilizing high order motifs is more likely to capture deviation from the genome background compositional distribution, as long as there is enough data to produce reliable probability estimates. However for high order motifs in short or biased sequences, a significant amount of data is likely to be missing. For example, using 8mers in a sliding window of 5kb, approximately 60,000 out of 65,536 (4^8) different possible 8mers will have an observed frequency of zero. Even for 8mers of non-zero frequency the information may not be enough to provide reliable estimates of the local sequence composition of a region, e.g. most 8mers will be present only once in a 5kb window.

An Interpolated Variable Order Motif (IVOM) approach (Vernikos and Parkhill, 2006) overcomes this problem, implementing variable order k-mers, "preferring" information derived from high order motifs, but when this information is insufficient, relying more on lower order motifs. Let B be the DNA alphabet, defined as: B = {a, t, g, c}. In an IVOM approach all k-mers with $1 \leq k \leq 8$ are exploited. Each k-mer can be seen as a linear combination of its component lower order motifs including itself. In a first step, for each k-mer m_k in the sequence S, its observed frequency $P_{m_k}(S)$ is calculated as follows:

$$P_{m_k}(S) = \frac{A_{m_k}(S)}{N - k + 1} \tag{2.1}$$

where $A_{m_k}(S)$ is the number of occurrences of m_k in the sequence S and N is the size of S. Generally a high order motif occurs less frequently (small number of occurrences) in a sequence compared to motifs of lower order, given that the total number of all different possible motifs is higher (larger alphabet) in the first case. In order to use in combination the different order k-mers, both the difference in the number of occurrences and in the total number of different possible k-mers have to be taken into account.

For each m_8 a weight is calculated for all ($1 \leq k \leq 8$) its interpolated k-mers, including itself, as follows:

$$W_{m_k^{m_8}}(S) = \frac{A_{m_k^{m_8}}(S) \cdot |B|^k}{\sum_{i=1}^{8} A_{m_i^{m_8}}(S) \cdot |B|^i} \qquad (2.2)$$

where $m_k^{m_8}$ denotes the interpolated k-mer m_k starting at position 8-k+1 and ending at position 8 in m_8; $|B|^k$ denotes the total number of all different possible motifs of size k. In this framework a high and a low order motif have equal chances of producing bias given that both number of counts and dimensionality have been taken into account. For example, for a given 8mer if the number of occurrences of the corresponding interpolated 3mer ($|B|^3$ = 64) and 5mer ($|B|^5$ = 1,024) is 128 and 8 respectively, an IVOM approach treats the two k-mers as equally reliable estimates (64 x 128 = 1,024 x 8) of the local sequence composition of a region. Having computed the weights for each k-mer, in a second step the IVOM frequency for each 8mer m_8, as well as all its interpolated k-mers $m_k^{m_8}$, in the sequence S is calculated as follows:

$$\mathrm{IVOM}(S, m_k^{m_8}) = \begin{cases} W_{m_k^{m_8}}(S) \cdot P_{m_k^{m_8}}(S) + [1 - W_{m_k^{m_8}}(S)] \cdot \mathrm{IVOM}(S, m_{k-1}^{m_8}) & \text{if } k \geq 2 \\ W_{m_k^{m_8}}(S) \cdot P_{m_k^{m_8}}(S) & \text{if } k = 1 \end{cases} \qquad (2.3)$$

The IVOM frequency of each interpolated k-mer is calculated step-wise, starting with the shortest interpolated k-mer (i.e. 1mer) and progressively moving towards longer k-mers all the way up to the 8mer itself. Using the above equation, it is possible for the observed frequencies of all the interpolated motifs to be combined linearly in such a way that if high order motifs are reliable (sufficient counts) estimates of the local sequence composition, then the corresponding weight will be high enough for the contribution of the lower motifs to be ignored and *vice versa*.

A similar equation is implemented by Salzberg (Salzberg *et al.*, 1998) in GLIMMER, a widely used gene prediction method. In GLIMMER however the above equation is used in a Markov model-based context i.e.

Interpolated Markov Models (IMMs). Moreover GLIMMER uses two different criteria in order to calculate the weight for each *k*-mer. The first is number of occurrences; if that number exceeds a pre-determined threshold value, then the weight is set to 1.0 (the default threshold value is 400). The second is a predictive value determined by a X^2 test comparing the observed base frequencies with the IMM probabilities derived from the immediately shorter context. In the IVOM algorithm however through equation 2.2 the weight for each *k*-mer is determined on-the-fly directly from the underlying local compositional landscape avoiding the incorporation of arbitrary threshold values (Table 2.1).

2.2.2 Relative entropy

In order to predict putatively horizontally transferred regions in microbial genomes, it is assumed that each genome exhibits a reasonably constant (although exceptions may apply – e.g. the rRNA operon) background sequence composition that is the result of the same mutational pressure applied throughout its sequence. Consequently regions of "atypical" composition within a genome are likely to have been horizontally acquired from a donor genome of different composition.

In order to detect compositionally deviating regions, a sliding window approach over raw genomic sequence is applied. In this framework the analysis of atypical regions can be applied both on annotated and newly sequenced genomes without any level of annotation (e.g. pre-existing gene prediction).

Obviously in a sliding window based approach, different window sizes and moving steps can be exploited. In order to converge over the optimal sliding window size *L*, I experimented on different *L* values, implementing a Receiver Operating Characteristic (ROC) curve analysis and the results (Appendix A) showed that the greatest Area Under the Curve (AUC), which is a measure of the accuracy of the classifier, for *k*-mers of $k \leq 8$ is achieved when the sliding window size and step is set to 5kb and 2.5kb respectively.

Table 2.1: Example of two different 8mers, present in the sequence of *Salmonella* Pathogenicity Island (SPI)-7 inserted in the chromosome of Typhi CT18. The interpolated, variable order motifs of each 8mer are shown along with their observed frequency A, calculated based on the compositional analysis of SPI-7. The weight W of each interpolated k-mer has been also calculated. In the first 8mer (GCCAGCGC), the interpolated k-mer with the highest compositional information is the 8mer itself whereas for the second 8mer (AAAACATG) the most informative interpolated k-mer is the di-nucleotide 'TG'.

8mer	Interpolated k-mer	A	$\|B\|^k$	A x $\|B\|^k$	W
GCCAGCGC	GCCAGCGC	24	4^8	1572864	**42.10**
	CCAGCGC	49	4^7	802816	21.51
	CAGCGC	116	4^6	475136	12.73
	AGCGC	238	4^5	243712	6.53
	GCGC	738	4^4	188928	5.06
	CGC	2452	4^3	156928	4.20
	GC	9784	4^2	156544	4.19
	C	33854	4^1	135416	3.63
AAAACATG	AAAACATG	1	4^8	65536	7.38
	AAACATG	6	4^7	98304	11.08
	AACATG	26	4^6	106496	12.00
	ACATG	81	4^5	82944	9.35
	CATG	474	4^4	121344	13.67
	ATG	2110	4^3	135040	15.21
	TG	9243	4^2	147888	**16.66**
	G	32499	4^1	129996	14.65

It should be noted that increasing the order of the utilized k-mers causes the optimal window size to increase too (Wu *et al.*, 2005). The same authors concluded that for symmetric Kullback-Leibler discrepancy as a similarity measure and $2550 \leq L \leq 4950$ the optimal word size k is 8, confirming the rationale behind the selection of a 5kb sliding window used in the current analysis. The step of the sliding window is set to 2.5kb; however, increasing the step size too much will increase the uncertainty about the real boundaries of the predicted "atypical" regions. This technical issue and how it can be handled efficiently will be discussed in the next section. Both for the sliding window w and the genome G a compositional vector, defined as:

$$\overrightarrow{\mathbf{IVOM}(\boldsymbol{S},\boldsymbol{m_8})} = \{\mathrm{IVOM}(S, m_8) \mid m_8 \in B^8\} \quad (2.4)$$

is built. This vector extends over all ($|B|^8$) the different possible 8mers m_8 in the sequence S. In order to compare the two vectors (of w and G) a distance similarity measure has to be applied. In the current methodology, the relative entropy (Kullback-Leibler – KL distance), defined as:

$$d_G(w) = \sum_{m_8 \in B^8} \mathrm{IVOM}(w, m_8) \log_2 \frac{\mathrm{IVOM}(w, m_8)}{\mathrm{IVOM}(G, m_8)} \quad (2.5)$$

is implemented. The KL distance is a reasonable similarity measure in this case, since the task at hand is to compare two probability distributions; moreover KL is always non-negative and equals zero only if the two distributions are identical. Implementing equation 2.5, a sequence region of "atypical" composition will have high relative entropy while native-typical regions will have relative entropy close to zero (compositional distribution closer to the genome); it should be noted that the compositional vector of the genome IVOM(G, m_8), extends over all 8mers present in the genome sequence, including those of the current sliding window w.

2.2.3 Score threshold

Given that the current implementation of the Alien_Hunter algorithm is unsupervised, the very specific compositional landscape of different query, previously unseen genomic sequences will determine the exact value of the score threshold; above this threshold value, regions that deviate from the backbone composition of the query chromosome will be reported as putative horizontally acquired candidates. Consequently a pre-determined value of a score threshold, based on a supervised training on a test dataset is not applicable in the case of chromosomal compositional analysis, given that some chromosomes may consist of almost zero (Tamas *et al.*, 2002) up to 24% of alien DNA (Nelson *et al.*, 1999); moreover some bacterial chromosomes, that contain several HGT events, show a fairly constant

backbone composition (e.g. *Salmonella*) while other genomes (e.g. *Staphylococcus*) display a highly mosaic composition.

Generally, for a typical microbial chromosome, the compositional distribution is a long-tail one of the form shown in Figure 2.3. The majority of the regions present in a bacterial genome will have a compositional distribution very close to the genome backbone composition (low IVOM score – blue coloured in Figure 2.3), a few will deviate (red colour) and very few will deviate strongly (green colour). A reasonable value for the score threshold is a value close to the point in the distribution where the transition from the "typical" (backbone – blue coloured) to the "atypical" (compositional deviating – red coloured) compositional score population occurs.

There are different approaches for capturing dynamically the optimal score threshold for any given, previously unseen microbial chromosome. In the current implementation of Alien_Hunter, I exploit two different methods. The first relies on a derivative-based approach, similar to the one exploited by Tsirigos and Rigoutsos (Tsirigos and Rigoutsos, 2005); in this approach, the transition, in the compositional score distribution f, from the "typical" to the "atypical "scores can be captured by calculating the derivative f'' of the distribution.

Starting from the highest scoring regions moving (sliding window based) towards low-scoring ones, the point in the distribution, where the value of f'' (calculated through the current sliding window) starts to remain steady (after several iterations) represents a good score threshold value that discriminates compositional deviating from non-deviating regions within a chromosome; the score threshold can be dynamically determined on-the-fly for each query genome. However, this approach can be quite sensitive to data noise depending on the actual shape of the compositional distribution. Moreover a derivative-based threshold often over-predicts (i.e. very low threshold); for example in the distribution shown in Figure 2.3 the point in the score distribution where the derivative starts to remain steady results in a score threshold of 7.6 well

below the value of 13.2 where a much stronger transition from the "typical" to "atypical" scores occurs (see next paragraph).

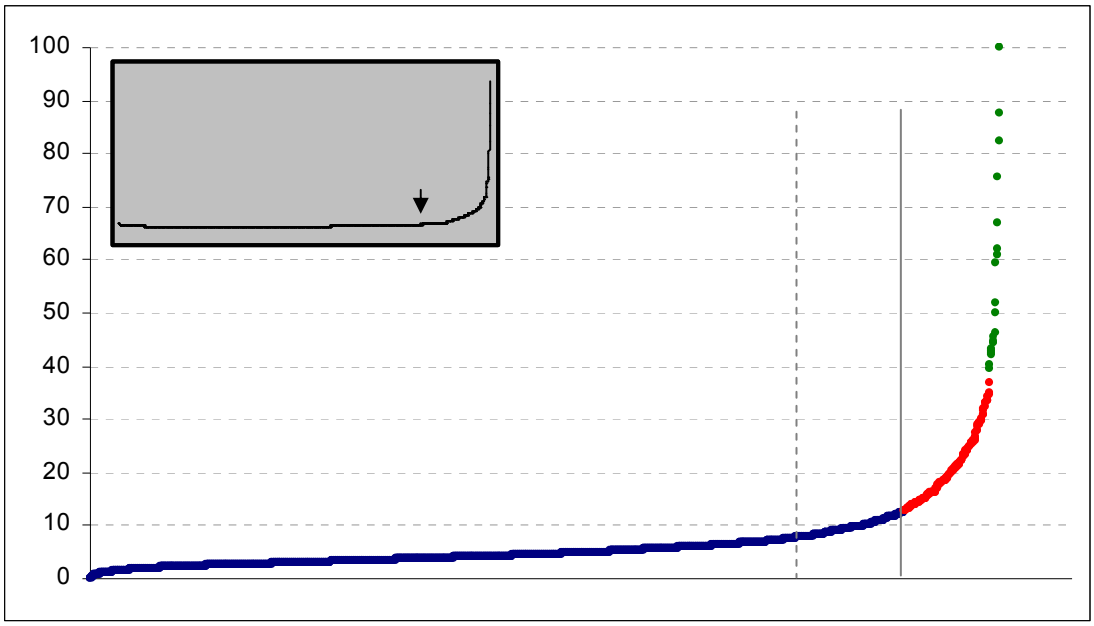

Figure 2.3: An example of the compositional score distribution of *E. coli* MG1655 chromosome. The IVOM score of all the sliding windows is plotted, sorted by increasing order. A three-colour scheme has been used to highlight the three distinct compositional populations, blue (backbone), red (intermediate compositional deviation), green (high compositional deviation). The dashed and solid, vertical grey line represents the score threshold determined by a derivative based (T=7.6) and a K-means clustering method (T=13.2), respectively. The derivative of the score distribution is plotted in the inset, with an arrow highlighting the value of the derivative used to determine the score threshold.

The second approach for determining dynamically the score threshold is based on the K-means clustering algorithm (MacQueen, 1967). K-means clustering is a non-hierarchical, supervised method, given that the initial number K of clusters is fixed and determined prior to the learning process. In the current implementation, in order to model properly the three distinct compositional populations (backbone, atypical and very atypical) a K-means clustering with three different clusters is exploited. The pseudocode describing the K-means clustering implementation is shown below.

Algorithm: K-means clustering.

C: number of re-initializations.
F: objective function.
i = 1.

1. Determine the number of clusters, K = 3.
2. Initialize the value of the 3 centroids.
3. Assign each point to the cluster with the nearest centroid value.
4. When all points have been assigned to one of the 3 clusters, update the new centroid values.
5. Re-iterate steps 3 and 4 until the 3 centroids do not change;
   convergence criteria: $Last\_F_i - Current\_F_i < 0.1$.
6. **If** $i < C$ **do**
   if $F_i > F_{i\_max}$ then $F_{i\_max} = F_i$
   i++
   goto step 2 re-initializing the 3 centroids with different values.
7. Set the score threshold to the value where the transition from cluster 1 → 2 occurs, for the iteration with F_{i_max}.

end

Although the K-means clustering algorithm always converges, the final clustering strongly depends on the values used to initialize the centroids of the three clusters. For those reasons different starting points are used to initialize the centroids and the iteration with the maximum (i.e. the one that separates the three clusters the most) objective function value is used to determine the score threshold. Through steps 1-5, the algorithm is trying to minimize the following objective function:

$$F = \sum_{j=1}^{K} \sum_{i=1}^{n} \left\| x_i - c_j \right\|^2 \tag{2.6}$$

where, K is the number of clusters (i.e. 3), n the total number of sliding windows, x_i the IVOM score of each sliding window and c_j the centroid value. As the clustering algorithm proceeds, the value of the F function decreases and converges over a minimum value; at this stage the clustering terminates.

Table 2.2: Performance benchmarking of the Alien_Hunter algorithm implementing a derivative and a K-means clustering based algorithm to determine the score threshold (6.1 and 11.3 respectively) for the genome of Typhi CT18.

Performance metric	Derivative	K-means
Specificity	0.653	0.746
Sensitivity	0.649	0.511
Accuracy	0.764	0.775
Matthews correlation coefficient	0.473	0.473

Overall, the K-means clustering determines more accurately the optimal score threshold (Figure 2.3) and is less affected by the noise in the compositional data and the shape of the distribution. The accuracy of the Alien_Hunter algorithm, implementing the derivative-based and the K-means clustering algorithm, was benchmarked using a manually curated (see below) dataset of 1560 putative horizontally acquired genes in Typhi CT18 (Table 2.2); the data confirm the higher accuracy of the second method compared to the first one, although a derivative-based threshold results in an overall more sensitive method, capable of detecting older HGT events with composition very close to the genome backbone.

2.2.4 Change-point detection

As mentioned in section 2.2.2 the choice of the step for the sliding window approach is crucial, given that the window slides over raw genomic sequence (consequently the gene boundaries are unknown), decreasing the window step will increase the computation required, and increasing the window step will reduce the accuracy of the localization of the predicted "atypical" regions. For these reasons, upon the completion of the first round of the window-based prediction, a second-order, two-state HMM is implemented in a change-point detection framework. HMM is a statistical model widely used in speech and music recognition (Rabiner, 1989; Raphael, 1999) as well as in several bioinformatics tasks, e.g. gene prediction (Burge and Karlin, 1997). HMMs can be thought as finite state

machines in which each state emits symbols governed by an emission probability distribution over a given alphabet of allowed symbols; at each stage, the model can either stay in the same state, or make the transition to a new one, a process governed by a distribution of transition probabilities; both the emission and transition probability distributions are state-specific.

HMMs can be described by two sequences (Durbin *et al.*, 1998). The hidden state sequence $\pi = (\pi_1,\ldots,\pi_L)$ also known as the "path" and the observed sequence $x = (x_1,\ldots,x_L)$ which corresponds to the observed symbols; in our case the bases of a DNA sequence. In an n-th order HMM each base x_i depends on the previous $(x_{i-n},\ldots, x_{i-1})$ bases as well as on the ith state π_i in the path. In the current study, two states are exploited: the "native" (N) state that corresponds to regions of typical (i.e. close to the genome backbone) composition and the "alien" (A) state that models compositionally deviating, "atypical" regions. Under this framework, a change-point corresponds to switching from one state to the other; in the current implementation the aim is to infer the boundaries of the predicted regions, where a state transition occurs. This change-point will represent the new optimized boundary of each prediction, offering higher predictive accuracy in terms of boundary localization (see results section). In order to detect the point where the transition from the native to the alien state occurs and *vice versa*, the following approach is pursued.

Each predicted "atypical" region is extended further upstream in order to incorporate sequence of typical composition. This hybrid sequence of one typical and one atypical subsequence, is used to train the HMM on-the-fly (the same approach is also applied on the downstream boundary – Figure 2.4). Implementing the Baum-Welch (BW) algorithm (Baum, 1972), the parameters (transition and emission probabilities) of the model are trained, in an iterative fashion until some convergence criteria are met. The BW algorithm is an Expectation Maximization (EM) technique that estimates transition and emission probabilities calculating the expected number of times each transition and emission is used, given the training

sequence; this is done iteratively, until convergence, by considering probable paths within the sequence exploiting each time the current/updated parameters of the model. However different starting parameter values strongly affect the local maxima which the BW will converge over. One straightforward solution to this problem is to start multiple times from different initial model parameters, an approach that is implemented in this analysis.

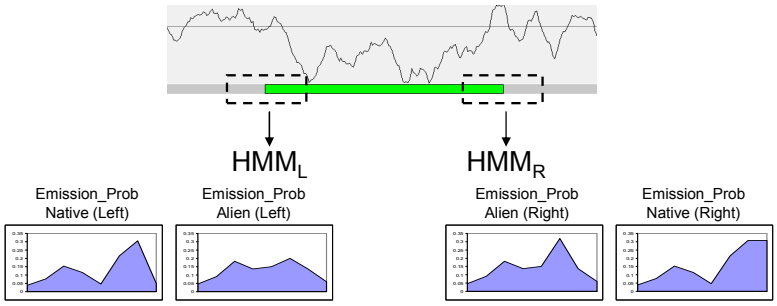

Figure 2.4: Two side-specific HMMs trained for the left (HMM$_L$) and for the right (HMM$_R$) boundary of each predicted GI.

Given that we do not know beforehand for how long the system remains in the native state before it makes the transition to the alien state (and *vice versa* for the downstream boundary) the algorithm starts with multiple starting points (*prior* expectations) over the transition probability:

$$\alpha_{NA} = P(\pi_i = A \mid \pi_{i-1} = N) \tag{2.7}$$

where α_{NA} denotes the transition probability from the native (N) to the alien (A) state; for each starting point, the model is trained using the BW algorithm until convergence.

In a change-point detection framework with a single change-point, once the α_{NA} transition occurs, the model persists at the alien state until the end. For this reason only the α_{NA} transition probability is trained, while the transition probability from the alien to the native state is set to be zero ($\alpha_{AN} = 0$, un-trainable). For the emission probabilities, given that the composition of the native and the alien DNA sequence is not known *a*

priori, two trainable, uniform, second-order compositional distributions are exploited (Figure 2.5).

In a second step for each starting point, upon BW training, the Viterbi algorithm (Viterbi, 1967) is implemented with the updated-trained parameters. The Viterbi algorithm is a dynamic programming algorithm widely used in inferring the most probable state path Π^* (in our case the most probable sequence of native/alien hidden states) given the observations (DNA bases) and the model parameters (emission and transition probabilities):

Algorithm: Viterbi.

1. Initialisation ($i = 0$): $v_0(0) = 1,\ v_N(0) = 0$ for $N > 0$.

2. Recursion ($i = 1 \ldots L$): $v_A(i) = e_A(x_i)\max_N(v_N(i-1)a_{NA})$; (2.8)

$ptr_i(A) = \text{argmax}_N(v_N(i-1)a_{NA})$.

3. Termination: $P(x,\Pi^*) = \max_N(v_N(L)a_{N0})$;

$\Pi_L^* = \text{argmax}_N(v_N(L)a_{N0})$.

4. Traceback ($i = L \ldots 1$): $\Pi_{i-1}^* = ptr_i(\Pi_i^*)$.

Source: (Durbin *et al.*, 1998).

Briefly a score $v_A(i)$ for each DNA base x_i in the state A (with the previous base x_{i-1} being in state N) is calculated (equation 2.8). The first part of this equation consists of the emission probability $e_A(x_i)$ of x_i in state A; the second part consists of the maximum value (over all values of N) of the product of the maximal score at the previous (i-1) base position and the transition probability from state N to A. The optimal path can be found by backtracking over an array of pointers ($ptr_i(A)$) that keep track, at each x_i in state A, of the maximum score of the previous state thus revealing the most probable sequence of hidden states that gave "birth" to the observed sequence.

In the Alien_Hunter algorithm, keeping track of the probability of the most probable path predicted by the Viterbi algorithm, the iteration

(over different starting points of a_{NA}) with the highest probable path, among all the most probable state paths, will be the one which best describes the data (the true transition point). An example is given in (Table 2.3) and the algorithm is summarized in the following pseudocode:

Algorithm: Change-point detection.

C: number of iterations
Init: i = 1;
a'_{NA}: initial starting point for a_{NA}

1. extend the predictions upstream and downstream
2. set initial model:
  2.1. *prior* distribution for the emission probabilities:
    2.1.1. *N* state: trainable second order uniform (e_N) distribution
    2.1.2. *A* state: trainable second order uniform (e_A) distribution
  2.2. *prior* transition probabilities:
    2.2.1. $a_{NA} = a'_{NA}$ (multiple starting points - trainable)
    2.2.2. $a_{AN} = 0$ (untrainable)
3. BW training until convergence:
  3.1. stopping criteria: LastScore - CurrentScore < 0.001
  3.2. updated-trained emission, transition probabilities
4. Viterbi: most probable path π^*, with score S_i
    4.1.1. **if** $S_i > S_{imax}$ then $S_{imax} = S_i$
5. **if** i < C **do**
    5.1.1. i++;
    5.1.2. new starting point a'_{NA}
    5.1.3. **goto** step 2
6. report the path π^* with S_{imax}
7. set predicted boundary = transition point in the path π^* with S_{imax}
end

Table 2.3: An example of multiple starting points for the transition probability a_{NA} and the corresponding score of the most probable path π^* predicted by the Viterbi algorithm for a test hybrid sequence.

iteration	score S_i of path π^*	*prior* over a_{NA}	change-point (bp)
1	-9643.868804	500^{-1}	1720
2	-9643.868873	1000^{-1}	1720
3	-9627.033373	2000^{-1}	4870
4	-9627.033077	2500^{-1}	4870
5	-9627.033131	3000^{-1}	4870

In the example described in Table 2.3 the best model (highest scored π^*) for estimating the position in the sequence where the transition from the N to the A state occurs, is the one in which the prior expectation over the a_{NA} value is 2500^{-1}. In the first two starting points, the predicted change-point occurs at 1720bp (starting from the 5' end of the test hybrid sequence) whereas in the remaining cases the change-point is predicted at 4870bp. In the current version of the Alien_Hunter software (http://www.sanger.ac.uk/Software/analysis/alien_hunter/) the BW and the Viterbi algorithms are implemented using the relevant Biojava libraries (http://www.biojava.org).

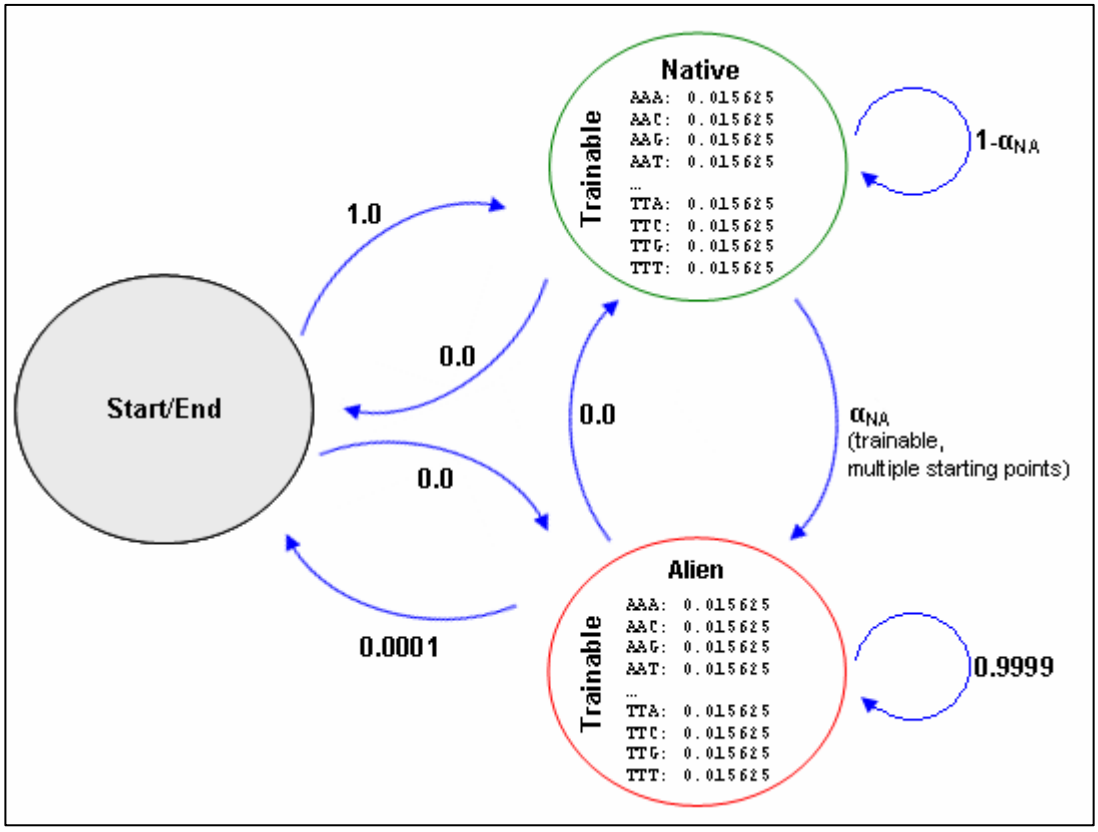

Figure 2.5: The architecture of the two-state (Native, Alien), second order HMM, used in a change-point detection framework.

2.2.5 Reciprocal FASTA

In order to evaluate the performance of Alien_Hunter, a test dataset of putative horizontally transferred genes was built. Previous approaches

(Azad and Lawrence, 2005; Tsirigos and Rigoutsos, 2005) involved simulation of HGT events by inserting genes from various donor genomes into the genome under study; such approaches simulate only very recent HGT events, thus they do not take into account the amelioration (Lawrence and Ochman, 1997) of horizontally acquired DNA, a time-dependent process. For this reason I chose to build a test dataset of putative HGT events, based on real data.

The genome of *S. typhi* CT18, a well-studied prokaryote in terms of HGT events, was used as the reference genome. *S. typhimurium* LT2 was selected as a sister lineage to Typhi while the genome of *E. coli* MG1655 was chosen as an outgroup of Typhi and Typhimurium. The main idea is that genes that are present in all the three genomes form a set of core genes, while the rest of the genes represent either species or strain specific genes thus are considered putative candidates for HGT. The choice of two sister lineages and one outgroup increases the chances of capturing older HGT events, which otherwise might be indistinguishable; for example SPI-1 and SPI-2 are species-specific, but not strain-specific islands. Moreover a comparative analysis between two sister taxa and one outgroup, enables a more reliable discrimination between gene loss and gene gain, two events that can equally explain a limited phylogenetic distribution of a gene, within a lineage. *E. coli* seems to form a good outgroup organism, given that the estimated divergence of *E. coli* and *S. enterica* from the common ancestor occurred approximately 100-140 Myr ago (Doolittle *et al.*, 1996; Ochman and Wilson, 1987). In order to extract all the putative horizontally transferred genes in Typhi, the following approach was pursued.

Each CDS (a) from the genome (A) was searched, with FASTA, against the CDSs of the other genome (B). If the top hit covered at least 80% of the length of both sequences with at least 30% identity, a reciprocal FASTA search of the top hit sequence (b) was launched against the CDSs of the first genome. If the reciprocal top hit is the same as the original query CDS then (a) and (b) are considered orthologous genes of (A) and

(B). Genes that are unique in, or are orthologs between Typhi and Typhimurium but do not have an ortholog in *E. coli* form the initial dataset of putative HGT events. In a second step, in order to validate the results, a BLASTN and TBLASTX comparison between the three genomes was carried out, to check for a syntenic relationship among the putative orthologs and the results were visualized using ACT (Carver *et al.*, 2005). It should be recognized that this procedure will also identify genes that have been uniquely deleted in *E. coli* as putative HGT events (see results section).

2.3 Results

2.3.1 Manually curated HGT dataset

Implementing the reciprocal FASTA approach described above, four different groups of genes present in Typhi were identified: The first group involves 725 genes that are unique in Typhi. The second and third group includes orthologous genes between Typhi and *E. coli* (52) and Typhi and Typhimurium (903). In the last group are 2920 core genes that are shared between all the three genomes (Figure 2.6).

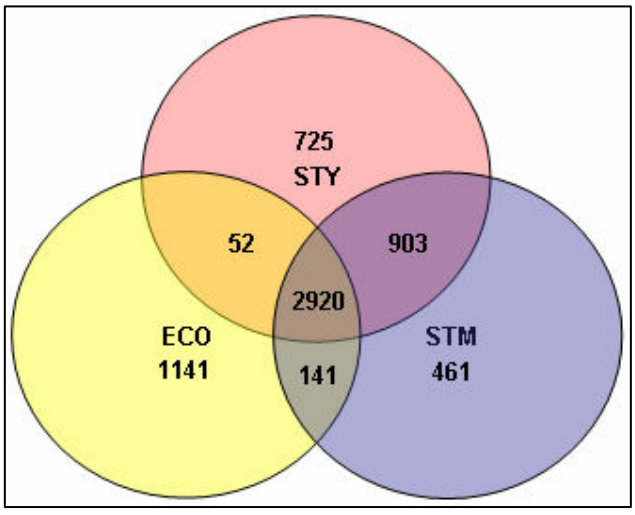

Figure 2.6: Venn diagram illustrating the unique and the orthologous genes present in the genome of *E. coli* (ECO), *S. typhi* (STY) and *S. typhimurium* (STM).

Excluding the 2920 predicted core genes and the 52 Typhi and *E. coli* unique orthologs, the remaining gene set (1628 genes) forms the initial dataset of putatively horizontally transferred genes in Typhi. In a second step, the above dataset was manually curated for gene position consistency using ACT, and the initial number was reduced to 1560 manually curated putative horizontally transferred genes which form the basis of the analysis described in the following sections.

It should be noted that this analysis yields a significantly high number of putative HGT events in the genome of Typhi CT18. The reliable estimation of true HGT events strongly depends on the evolutionary sample at hand; going well back in the evolutionary history of an organism offers more reliable detection of sequences that have been transferred horizontally from other sources. For example, some of the *Salmonella* lineage-specific genes might not necessarily represent HGT events (gene loss in *E. coli*). However this analysis provides a more reliable estimation of putative HGT events (taking into account the amelioration process), given that it is based on real data rather on simulated events. A more robust approach of discriminating putative gene gain from gene loss events will be described and discussed in chapter 3.

2.3.2 Three novel SPIs

Running the Alien_Hunter algorithm on the genome of Typhi CT18, all the previously annotated SPIs (SPI-1 to SPI-10) and bacteriophages were successfully predicted. Moreover this analysis revealed three novel putative SPIs, SPI-15, SPI-16 and SPI-17 (Table 2.4); SPI-11, 12 and SPI-13, 14 have been previously described is other *Salmonella* serovars (Chiu *et al.*, 2005; Shah *et al.*, 2005). SPI-15 represents an insertion of approximately 6.3kb, inserted in the 3' end of a $tRNA^{Gly}$; the insertion has duplicated a 22bp tRNA fragment, which forms the downstream boundary of SPI-15. Adjacent to the tRNA, there is an integrase gene of putative phage origin and further downstream four hypothetical protein-coding genes. Among the eight *Salmonella* genomes analyzed, SPI-15 is only

present in Typhi CT18 (Figure 2.7); in Typhi TY2, there is a similar insertion of different gene content, at the same position, which is also flanked by two DRs, 22bp long.

Figure 2.7: ACT screenshot: BLASTN comparison between *E. coli* and 8 *Salmonella* genomes (from top to bottom): *E. coli* MG1655, *S. typhi* CT18, *S. typhi* TY2, *S. paratyphi* A, *S. typhimurium* LT2, *S. gallinarum* 287/91, *S. enteritidis* PT4, *S. arizonae* RSK2980, *S. bongori* 12419. Regions within the nine genomes with sequence similarity are joined by red coloured bands that represent the matching regions. The three novel SPIs are illustrated as white coloured features (from left to right: SPI-16, SPI-17, SPI-15). The above screenshot is a mosaic picture of three individual screenshots at different locations along the genomes that have been concatenated for ease of visualization.

Although SPI-15 is a 6.3kb island, Alien_Hunter predicts a much larger (~18kb) region overlapping the SPI-15 locus. The predicted region starts at the exact 5' end of SPI-15, at the insertion point within the tRNA locus, but extends the 3' end 12kb further to the left (Figure 2.8), questioning the accuracy of the predicted boundaries. SPI-15 represents a very recent insertion, present only in the genome of Typhi CT18; however the comparison between Typhi CT18 and *E. coli* MG1655 suggests that the entire (~18kb) predicted region is absent from the genome of the latter. Possibly, it represents a mosaic region of more than one independent HGT events; a fairly old insertion (~12kb, G+C content = 50.1%) upstream of SPI-15 and a very recent insertion (SPI-15, G+C content = 48.9%). This observation suggests that Alien_Hunter shows increased sensitivity,

predicting correctly even old HGT events or regions of mosaic compositional profile. A closer look at the 5' end of the entire 18kb region reveals that the predicted boundary starts immediately downstream of CDS STY3169 (encoding a histidine rich hypothetical protein); STY3169 is a pseudogene with an in-frame stop codon at position 110.

Table 2.4: Characteristics of the three novel predicted SPIs (SPI-15, SPI-16 and SPI-17) in the genome of Typhi CT18.

SPI	Location	Insertion site	Repeats	Integrase	Score	Size (bp)	Potential virulence determinants
SPI-15	3053654..3060017	$tRNA^{Gly}$	22nt (DR)	phage integrase	18.893	6364	unknown
SPI-16	605515..609992	$tRNA^{Arg}$	43nt (DR)	phage integrase	20.949	4478	serotype conversion by O-antigen glucosylation
SPI-17	2460793..2465914	$tRNA^{Arg}$	-	-	23.953	5122	serotype conversion by O-antigen glucosylation

The overall G+C content of STY3169 is 52.6% (genome average 52.09%), while the G+C contents from the 5' end up to the in-frame stop codon, and from the stop codon to the 3' end of this CDS are 49.8% and 54.3% respectively. Based on the comparison between Typhi CT18 and *E. coli* MG1655 the true 5' boundary of the 18kb locus is upstream of STY3169, suggesting perhaps that STY3169 is expected to be part of the 18kb locus. Perhaps, STY3169 as a non functional CDS, carrying an internal in-frame stop codon, has been subject to accelerated amelioration, a likely scenario, taking into account its mosaic composition (upstream and downstream of the stop codon) and the nonetheless very similar overall composition to the genome average (52.6% and 52.09% respectively). This further explains why the boundary predicted by the Alien_Hunter algorithm does not encompass the STY3169 CDS.

The second novel SPI, SPI-16 is a 4.5kb long island, inserted in a $tRNA^{Arg}$ gene. Two DRs of 43bp form the boundaries of SPI-16 while a phage integrase (pseudogene) is located near the tRNA gene. Encoded within this island are two bactoprenol-linked glucose translocases (*gtrA*

and *gtrB*) that along with the integrase pseudogene show high percentage identity (93%, 97% and 78% respectively) to homologous genes in the genome of bacteriophage P22 (Figure 2.9). *gtrA* and *gtrB* have been previously described to be involved in serotype conversion through O-antigen glycosylation mediated by bacteriophages (Guan *et al.*, 1999; Mavris *et al.*, 1997).

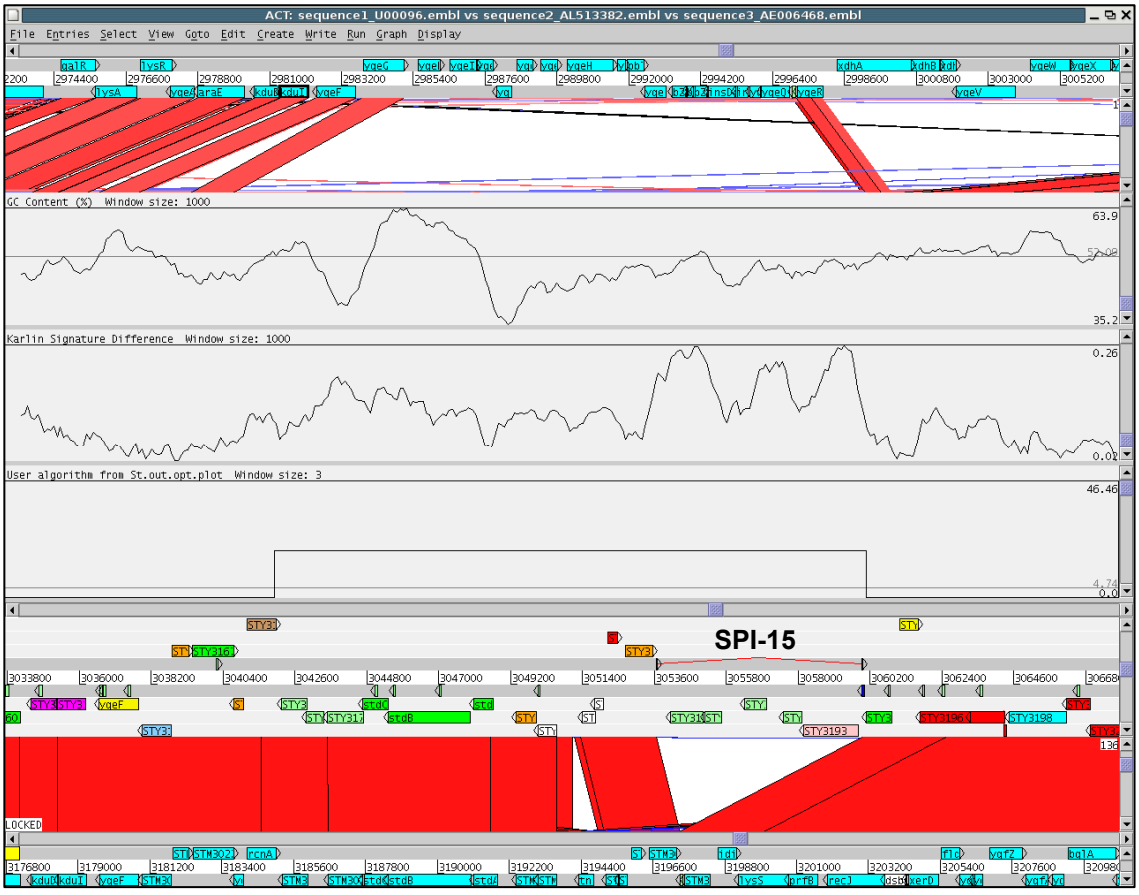

Figure 2.8: ACT screenshot: BLASTN between *E. coli* MG1655, Typhi CT18 and Typhimurium LT2 (from top to bottom) at the genomic locus encompassing SPI-15 (flanked by DRs – red-coloured joined features). Plots (from top to bottom): G+C% content, di-nucleotide bias (δ^* difference) (window size = 1kb) and IVOM score. Regions within the three genomes with sequence similarity are joined by red coloured bands that represent the matching regions. The brown coloured CDS (STY3169), encoding for a histidine rich hypothetical protein, is a pseudogene with an in-frame stop codon at position 110.

This observation leaves open the possibility that SPI-16 and SPI-17 (see next paragraph) are GIs probably involved in driving the variation of the cell surface structure of Typhi and perhaps the way this bacterium

interacts with its host (i.e. humans) or "parasitic" mobile elements, e.g. bacteriophages.

Also present in SPI-16 is STY0605 that encodes a putative membrane protein with nine predicted transmembrane segments (TMs). Although there is no sequence similarity to the *gtrC* gene in P22 bacteriophage (data not shown), both genes encode proteins with TMs in equivalent positions (the same applies for STY2629 of SPI-17 – see next paragraph). It seems possible that those proteins have similarity on the structural rather on the sequence level which might indicate similar function. Moreover the DR at the 5' end of SPI-16 has significant sequence similarity (74% in 23nt) with the 23bp P22 bacteriophage attP attachment site (alignment in Figure 2.9).

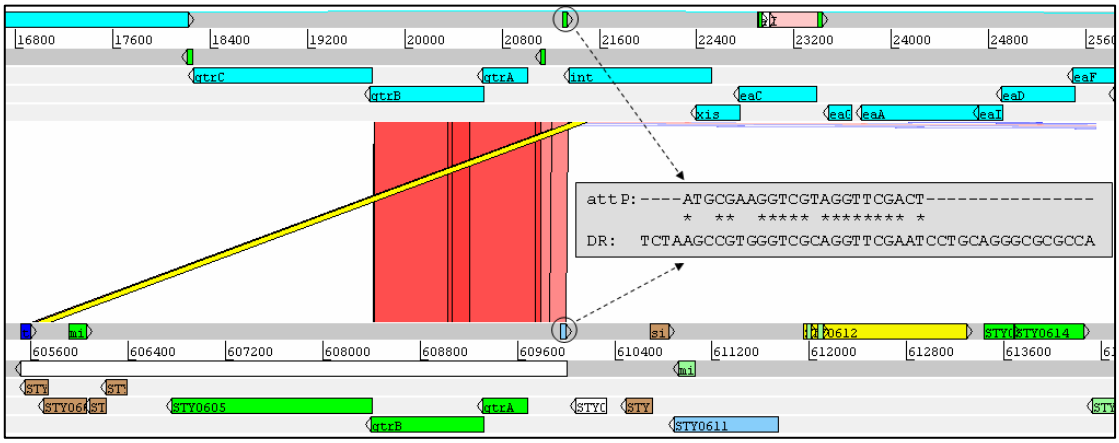

Figure 2.9: ACT screenshot: BLASTN comparison between bacteriophage P22 and Typhi CT18 (from top to bottom). The highlighted yellow band represents the sequence similarity between the P22 phage integrase and the integrase pseudogene in SPI-16. Within the grey text box the sequence alignment between the DR of SPI-16 and the P22 bacteriophage attP is provided; identical bases are indicated with an asterisk.

These data support the phage origin of SPI-16 and indicate that this island seems to have been originated from a phage that shares similarities with P22 bacteriophage family. SPI-16 is absent from *E. coli*, *S. bongori* and *S. arizonae* while it is present in the rest of the *Salmonella* lineage (Figure 2.7). Interestingly in *S. bongori* at the same tRNA location, there is a different insertion (8155bp) with a phage integrase gene, suggesting

that this tRNA locus might represent a hotspot for integration of different GIs in the *Salmonella* lineage.

The third novel island, SPI-17 is 5.1kb long, inserted in a $tRNA^{Arg}$ gene. An integrase gene and DRs/IRs seem to be absent from this island, which is present in all the *Salmonella* genomes used in this study, apart from *S. bongori*, *S. arizonae*, and *S. typhimurium*; this observation may indicate a possible recent deletion event that took place in the genome of *S. typhimurium* (Figure 2.7). SPI-17 seems to belong to the same phage family as SPI-16 given that the two serotype converting genes (*gtrA* and *gtrB*) are also present in the former island and both show high similarity with homologous genes in P22 bacteriophage; moreover in SPI-17 there is a pseudogene (STY2631a) with sequence similarity to the P22 phage bifunctional tail protein coding gene (TSPE_BPP22), suggesting an island of phage origin with two well defined boundaries (*gtrA* and the phage tail protein coding gene).

2.3.3 Predicted boundary optimization

As mentioned earlier, given that the current method is sliding window-based, the step of the window significantly affects the accuracy of the localization of the predicted boundaries.

The implementation of a HMM model in a change-point detection framework seems to provide an effective way of dealing with this problem (Table 2.5). Indeed the average absolute error δx for the predicted boundaries with the implementation of the HMMs is much lower (3830bp) than that without the boundary optimization (4936bp). Interestingly the HMM-based approach gives an average δx quite close to the W8 method (Tsirigos and Rigoutsos, 2005) (3543bp); W8 is a gene-based method, thus it is expected to provide quite accurate predicted boundaries of HGT events.

Overall this indicates that the implementation of HMMs in a change-point detection framework significantly improves (22%) the localization of the predicted boundaries; an example is illustrated in

Figure 2.10. This region is absent from the genome of *E. coli* and *S. typhimurium* and the BLASTN comparison indicates a well defined putative horizontally transferred region, 5223bp long, consisting of four genes (STY3343, STY3344, STY3345, STY3347: putative membrane and putative hypothetical genes of no significant database hits). As illustrated in the score plot in Figure 2.10, the unoptimized boundaries (green coloured plot) were predicted in the middle of STY3343 and STY3349 genes.

Table 2.5: Absolute error of the Alien_Hunter algorithm for the predicted boundaries with (optimized) and without (unoptimized) the implementation of HMMs in a change point detection framework. In addition the absolute error of the W8 (gene-based) method is provided as a control set. The absolute error is defined as $\delta x = | x - x_0 |$, where x is the annotated boundary and x_0 is the predicted one.

Annotated HGT	**Boundaries(bp)**								**Absolute Error (bp)**					
	Annotated		**Optimized (HMM)**		**Unoptimized**		**W8**		**Optimized**		**Unoptimized**		**W8**	
	left	right	left	right	left	right	left	right	left	right	left	right	left	right
SPI-6	302172	361067	302445	358919	302500	362500	306935	360757	273	2148	328	1433	4763	310
Prophage10	1008747	1051266	999914	1053088	1000000	1055000	1001995	1055793	8833	1822	8747	3734	6752	4527
SPI-5	1085156	1092735	1081688	1091828	1082500	1095000	1085337	1094839	3468	907	2656	2265	181	2104
Bacteriophage	1538899	1572919	1539019	1572916	1537500	1577500	1538899	1574581	120	3	1399	4581	0	1662
SPI-2	1625084	1664823	1624923	1650692	1622500	1652500	1622537	1667392	161	14131	2584	12323	2547	2569
Bacteriophage	1887450	1933558	1872930	1933953	1870000	1937500	1870173	1939495	14520	395	17450	3942	17277	5937
SPI-9	2743495	2759190	2743818	2754300	2742500	2755000	none	2759190	323	4890	995	4190	none	0
Bacteriophage 27	2759733	2782364	2759506	2787702	2757500	2787500	2759733	2783554	227	5338	2233	5136	0	1190
SPI-1	2859262	2899034	2862660	2900872	2860000	2902500	2861845	2900586	3398	1838	738	3466	2583	1552
SPI-8	3132606	3139414	3133940	3151951	3130000	3152500	3134156	3149714	1334	12537	2606	13086	1550	10300
Bacteriophage	3515397	3549055	3514572	3558310	3512500	3562500	3512700	3552416	825	9255	2897	13445	2697	3361
SPI-3	3883111	3900458	3888383	3904602	3887500	3907500	3888370	3902214	5272	4144	4389	7042	5259	1756
SPI-4	4321943	4346614	4321935	4348906	4320000	4350000	4321410	4349963	8	2292	1943	3386	533	3349
SPI-7	4409511	4543072	4402961	4541642	4402500	4545000	4401582	4542913	6550	1430	7011	1928	7929	159
SPI-10	4683690	4716539	4685054	4723629	4682500	4727500	4683853	4728101	1364	7090	1190	10961	163	11562
ALL (left/right)									3112	4548	3811	6061	3731	3356
ALL (left+right)									**3830**		**4936**		**3543**	

Applying the HMM approach, the true transition points were successfully identified (red plot), predicting the exact downstream and upstream boundaries of this region, diminishing the uncertainty of the localization of the predicted regions caused by the sliding window approach. The reason why I chose not to apply a purely HMM-based approach in the first place was the fact that a significant number of GIs (e.g. SPI-2) show a very mosaic structure, a result of several individual acquisitions, perhaps of different origin. Given that a HMM

implementation requires the properties of the regions modeled to remain constant throughout their whole length, such an approach is not readily applicable to the prediction of GIs in microbial genomes.

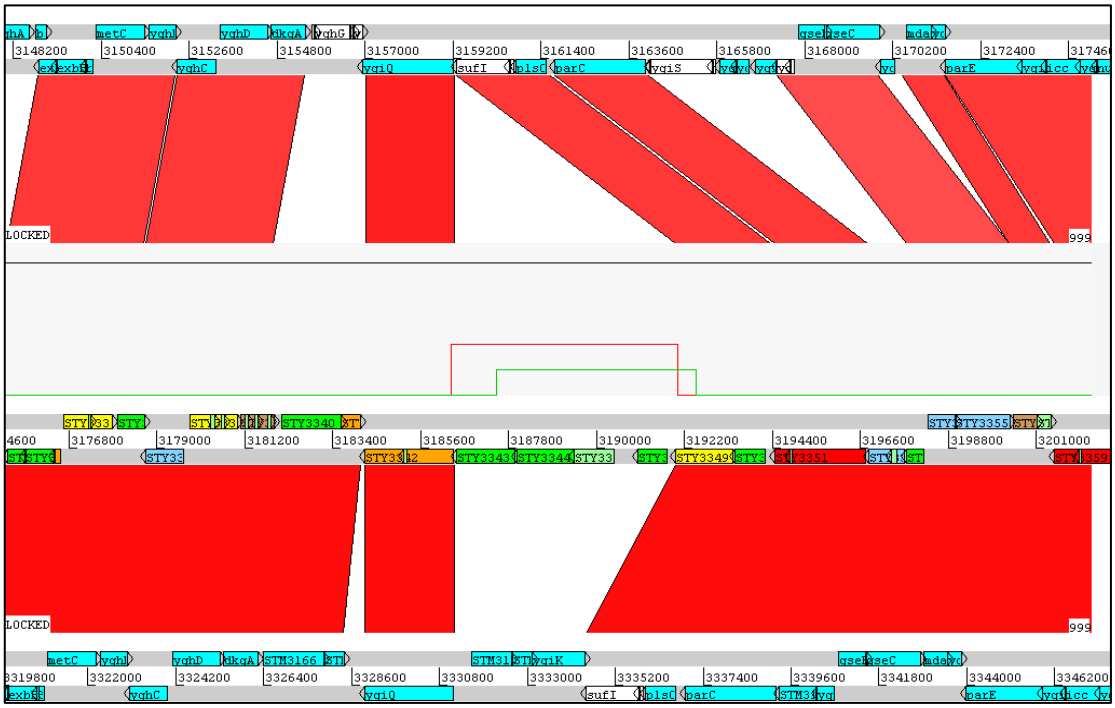

Figure 2.10: ACT screenshot: BLASTN comparison between (from top to bottom): *E. coli* MG1655, *S. typhi* CT18 and *S. typhimurium* LT2. An example of a predicted putative horizontally transferred region in the genome of *S. typhi* is indicated with two peaks in the IVOM score plot (above *S. typhi*). This region seems to be absent in the other two genomes compared. The red and the green coloured IVOM score plots represent the predictions of Alien_Hunter with optimized (HMM) and unoptimized boundaries respectively.

2.3.4 Performance benchmarking

In order to test the performance of the Alien_Hunter algorithm, a dataset of 1560 manually curated putative horizontally transferred genes in the genome of Typhi was used. Alien_Hunter was compared against four other published methods for the prediction of putative HGT events (Table 2.6): Islander (Mantri and Williams, 2004), IslandPath (Hsiao *et al.*, 2003), HGT-DB (Garcia-Vallve *et al.*, 2003), and the W8 method (Tsirigos and

Rigoutsos, 2005). Furthermore the above methods and the method for the prediction of PAIs introduced by Yoon *et al.* (Yoon *et al.*, 2005) were tested in terms of percentage coverage of the 10 previously described SPIs (SPI-1 to SPI-10) and the five annotated bacteriophages (Table 2.7).

Overall, Alien_Hunter shows the highest predictive accuracy (AC=0.764) compared with the other four methods (Table 2.6). Interestingly, the second most accurate method is W8, which utilizes higher order motifs (i.e. 8mers). These data suggest that the utilization of interpolated variable order motifs, improves both the sensitivity (SN) (Alien_Hunter: 0.649, W8: 0.62) and the specificity (SP) (Alien_Hunter: 0.653, W8: 0.643) compared with fixed-order methods; similarly this analysis confirms the superiority of higher order motif methods, discussed in the introduction.

Table 2.6: Performance comparison of the Alien_Hunter algorithm with other prediction methods. The comparison was based on the manually curated dataset of 1560 putative horizontally transferred genes, described in the text. TP: true positives, FP: false positives, TN: true negatives, FN: false negatives, SN: sensitivity, SP: specificity, AC: accuracy, CC: Matthews correlation coefficient. The performance of IslandPath was evaluated based on two compositional indices: G+C% content and di-nucleotide bias (δ^* difference).

Method	TP	FP	TN	FN	Number of Predictions	SN	SP	AC	CC
Alien_Hunter	1013	539	2501	547	1552	0.649	0.653	0.764	0.473
W8	968	538	2502	592	1506	0.620	0.643	0.754	0.447
HGT-DB	435	116	2924	1125	551	0.279	0.789	0.730	0.351
Islander	275	89	2951	1285	364	0.176	0.755	0.701	0.258
IslandPath (GC)	611	467	2573	949	1078	0.392	0.567	0.692	0.266
IslandPath (δ^*)	301	492	2548	1259	793	0.193	0.380	0.619	0.039

The sensitivity of Alien_Hunter is much higher compared to the other four methods which in turn reflects an increased ability to predict novel, putative horizontally transferred regions as well as already known examples. In terms of specificity Alien_Hunter is third from the top, following the Islander and the HGT-DB. Perhaps this can be attributed to

the increased number of predictions provided by Alien_Hunter (1552) compared to the Islander (364) and HGT-DB (551) as well as to the fact that Alien_Hunter runs on raw genomic sequence without gene position information. Compared to the W8 method, although Alien_Hunter provides higher number of predictions, both its sensitivity and specificity are higher. In the second performance analysis, based on the percentage coverage of previously described HGT events, the Alien_Hunter predictions overlap with 91.2% of the CDSs present in SPIs and bacteriophages giving the highest number of complete GIs in Typhi, followed by the W8 method with 80.7% coverage.

Table 2.7: Performance comparison of the Alien_Hunter algorithm with other prediction methods based on a dataset of 10 previously described SPIs (SPI-1 to SPI-10) and five annotated bacteriophages (SopE and P4 bacteriophages were ignored because they overlap with SPI-7 and SPI-10 respectively). For each annotated island or phage the % CDS coverage by each method has been calculated. The genomic locations of annotated bacteriophages (from top to bottom) are: 1008747..1051266, 1538899..1572919, 1887450..1933558, 2759733..2782364 and 3515397..3549055.

Annotated HGT	# CDS	Alien_Hunter	Islander	IslandPath		HGT-DB	W8	Yoon *et al.*
				GC	δ^*			
SPI-6	60	81.7	0.0	51.7	41.7	40.0	70.0	0.0
Prophage10	63	81.0	100.0	23.8	39.7	25.4	96.8	0.0
SPI-5	8	100.0	100.0	75.0	100.0	100.0	100.0	100.0
Bacteriophage	53	100.0	0.0	39.6	5.7	34.0	86.8	0.0
SPI-2	44	77.3	0.0	61.4	18.2	68.2	77.3	100.0
Bacteriophage	71	88.7	0.0	33.8	8.5	35.2	94.4	0.0
SPI-9	4	25.0	0.0	25.0	50.0	0.0	25.0	0.0
Bacteriophage	19	89.5	0.0	36.8	0.0	5.3	73.7	26.3
SPI-1	44	95.5	0.0	54.5	25.0	77.3	88.6	40.9
SPI-8	16	100.0	0.0	68.8	0.0	68.8	68.8	0.0
Bacteriophage	46	89.1	0.0	37.0	6.5	23.9	60.9	0.0
SPI-3	14	85.7	0.0	28.6	0.0	14.3	42.9	100.0
SPI-4	7	100.0	0.0	85.7	0.0	100.0	100.0	100.0
SPI-7	149	100.0	31.5	31.5	32.2	28.2	81.2	10.1
SPI-10	29	100.0	44.8	44.8	62.1	6.9	72.4	0.0
ALL	627	91.2	20.9	40.5	25.0	36.8	80.7	17.7

These data suggest that Alien_Hunter is capable of detecting not only novel GIs but also of identifying the majority of the already known regions of "alien" origin. Overall Alien_Hunter predicts six complete structures (SPI-5, the bacteriophage at 1538899..1572919, SPI-8, SPI-4, SPI-7 and SPI-10), while in the case of SPI-2 it predicts 34 out of 44 genes; it has been shown previously (Hensel *et al.*, 1999) that SPI-2 is a mosaic island of at least two independent acquisitions (see chapter 3). The mosaic nature of this SPI is also apparent in the G+C content (44.08% and 52.85% for the two parts of the island). This observation might explain the fragmented prediction for this SPI by all the methods except for the method of Yoon *et al.* (Yoon *et al.*, 2005). The latter combines a method for capturing sequence deviation and similarity matches to already known PAIs to predict PAIs instead of GIs in general. Such methods can be powerful approaches in the detection of complete PAI structures of similar gene content with previously annotated ones, but are not directly applicable in the detection of novel PAIs or GIs.

Overall the W8 method only outperforms the Alien_Hunter algorithm twice: in the first case it predicts 96.8% (Alien_Hunter: 81%) of the complete structure of prophage10 and in the second case 94.4% (Alien_Hunter: 88.7%) of the bacteriophage located at position 1887450..1933558. The Islander provides the lowest number of predictions (364) perhaps due to the fact that it is restricted to predict only complete GI structures. In the case of known Typhi islands, Islander predicts three SPIs (SPI-5, SPI-7, SPI-10) and one bacteriophage (prophage 10); the rest of the already known SPIs were not predicted by this method although some of them (e.g. SPI-8) have identifiable tRNA and integrase genes.

2.4 Discussion

In this chapter, I introduced and described a novel computational method for the prediction of putative horizontally transferred regions. This method, IVOM, exploits compositional biases at various levels (e.g. codon, di-nucleotide and amino acid bias, structural constraints) by implementing

variable order motif distributions. Under this framework, the local sequence composition can be captured more reliably, compared to fixed-order methods. The IVOM approach relies more on higher order motifs to make more accurate predictions, but when the underlying information is insufficient for high order motifs, it takes into account information obtained from lower order motifs. Moreover, an IVOM approach can be applied even on newly sequenced genomes, given that it does not require any level of pre-existing annotation or gene position information.

I also discussed the implementation of a HMM-based approach in a change-point detection framework for the optimization of the boundaries of the predicted regions and showed that the uncertainty of the localization of the predictions caused by a sliding window method can be sufficiently handled by such an approach enabling more accurate localization of putative HGT events. Applying the IVOM method on the genome of Typhi, all the previously annotated SPIs and bacteriophages were successfully predicted; moreover, the analysis of Typhi revealed the presence of three novel SPIs, SPI-15 to SPI-17, that have not been previously described. SPI-16 and SPI-17 represent islands of putative phage origin that may be implicated in serotype conversion by O-antigen glycosylation.

The performance benchmark of the Alien_Hunter algorithm against four published methods indicates that this method is more sensitive in detecting compositionally deviating, putative HGT regions. On the other hand Alien_Hunter shows fairly poor specificity compared with HGT-DB and Islander. This observation seems to indicate that the last two methods are more reliable in terms of SP compared to Alien_Hunter. One obvious reason behind the lower SP of Alien_Hunter is the increased number of predictions (1552). HGT-DB and Islander show the highest SP due to the low number of predictions (551 and 364 respectively); in other words they sacrifice SN for SP, predicting only a small fraction of the already annotated HGT regions (Table 2.7).

However if both SP and number of predictions are taken into account, Alien_Hunter provides the highest number of predictions and at the same time its SP is even higher than W8's, although the latter provides a lower number of predictions (1506). Overall this indicates that Alien_Hunter can be more sensitive and accurate compared to other methods that provide equally high number of predictions. It should be noted that this performance benchmark is based on a reciprocal FASTA approach that might penalize older HGT regions that were inserted prior to the divergence of *E. coli* and *Salmonella* lineages and were predicted by the Alien_Hunter algorithm. Such cases are considered False Positives based on this analysis, although they might represent true HGT events, and significantly affect the assigned SP of this algorithm.

Furthermore, the approach for the identification of orthologous genes, exploiting a reciprocal FASTA methodology, would in theory fail to correctly predict true orthologs in the following cases: A. One or both orthologs are pseudogenes, B. one of the orthologs has been deleted in one of the two genomes, C. a gene duplication event has created extra copies of the corresponding ortholog(s), D. one of the orthologs has not been annotated in one of the two genomes – although being present, E. one of the orthologs has been mis-annotated (truncated or extended) to such an extend that the condition of the minimum length of the region being similar in the two sequences is violated, and F. one or both orthologs are fast evolving, to such an extend that it is impossible, relying purely on sequence information, to predict them as true orthologs. With the exception of case F, all the other cases can be manually inspected and corrected, exploiting the genome annotation and gene position information; therefore, the results discussed in this chapter as well as in chapters 3 and 4, relative to the number of horizontally acquired genes, should be treated as an upper bound to the true number of HGT events, since fast evolving orthologs, could in theory be incorrectly classified as horizontal acquired genes.

The prediction of the three novel SPIs in Typhi CT18, raises the following question: *What is the minimum size of PAIs or GIs that still maintain their ability to mobilize (integrate-excise)?* Usually GIs are expected to be large (≥ 10kb), distinct chromosomal regions (Schmidt and Hensel, 2004). The three novel SPIs described in this analysis seem to represent exceptions to this rule, with a size of 4-6kb. For example SPI-17 is a minute PAI, and is absent from the genome of Typhimurium LT2, possibly indicating a recent deletion or recombination event. The size of these regions may be the reason why they have not been previously reported.

SPI-15 encodes four hypothetical protein-coding genes with unknown function. Moreover while SPI-15 is only present in Typhi CT18 and TY2, it can also be found in *Shigella flexneri* serovar 2a, strains 301 and 2457T. Given that SPI-15 or similar structures are present in *S. flexneri* and *S. typhi* but not in *E. coli* (MG1655, EDL933, O157:H7 and CFT073) or other *Salmonella*, it would be interesting to further investigate the functionality of SPI-15 with respect to the biology of *S. typhi* and *S. flexneri*, given that both organisms are human-restricted enteric pathogens.

The annotation of horizontally transferred regions (e.g. GIs, phages) is a key task in annotation pipelines, especially in the case of pathogens since it can reveal pathogenic aspects and characteristics of newly sequenced genomes. Prediction methods that reliably detect regions of "alien" origin, requiring a minimum level of annotation, can form a powerful tool for the understanding and analysis of the biology for the genome at hand, revealing key evolutionary steps in becoming a "successful" pathogen (see chapter 3).