 Nucleic Acids Research, 1999, Vol. 27, No. 19 **3899–3910**

Comparison of five methods for finding conserved sequences in multiple alignments of gene regulatory regions

Nikola Stojanovic[1], Liliana Florea[1], Cathy Riemer[1], Deborah Gumucio[4], Jerry Slightom[5], Morris Goodman[6], Webb Miller[1,3] and Ross Hardison[2,3,*]

[1]Department of Computer Science and Engineering and [2]Department of Biochemistry and Molecular Biology and [3]Center for Gene Regulation, The Pennsylvania State University, University Park, PA 16802, USA, [4]Department of Anatomy and Cell Biology, University of Michigan Medical School, Ann Arbor, MI 48109-0616, USA, [5]Molecular Biology Unit 7242, Pharmacia and Upjohn Inc., Kalamazoo, MI 49007, USA and [6]Department of Anatomy and Cell Biology, Wayne State School of Medicine, Detroit, MI 48201, USA

Received May 3, 1999; Revised and Accepted August 5, 1999

ABSTRACT

Conserved segments in DNA or protein sequences are strong candidates for functional elements and thus appropriate methods for computing them need to be developed and compared. We describe five methods and computer programs for finding highly conserved blocks within previously computed multiple alignments, primarily for DNA sequences. Two of the methods are already in common use; these are based on good column agreement and high information content. Three additional methods find blocks with minimal evolutionary change, blocks that differ in at most *k* positions per row from a known center sequence and blocks that differ in at most *k* positions per row from a center sequence that is unknown a priori. The center sequence in the latter two methods is a way to model potential binding sites for known or unknown proteins in DNA sequences. The efficacy of each method was evaluated by analysis of three extensively analyzed regulatory regions in mammalian β-globin gene clusters and the control region of bacterial arabinose operons. Although all five methods have quite different theoretical underpinnings, they produce rather similar results on these data sets when their parameters are adjusted to best approximate the experimental data. The optimal parameters for the method based on information content varied little for different regulatory regions of the β-globin gene cluster and hence may be extrapolated to many other regulatory regions. The programs based on maximum allowed mismatches per row have simple parameters whose values can be chosen a priori and thus they may be more useful than the other methods when calibration against known functional sites is not available.

INTRODUCTION

The rapid expansion in the amount of DNA and inferred protein sequence data resulting from the progress of genome initiatives and other projects has led to a compelling need for computational aids in identifying important, functional segments within these sequences (1). One successful approach has been to find sequences that are highly similar in phylogenetic comparisons; these slowly changing sequences have been reliable guides to functional elements both in protein coding (2,3) and regulatory (4,5) regions of genes. This paper presents and compares five methods, three of them novel, for identifying potential candidates for regions within homologous DNA sequences that have experienced natural selection. The applications discussed are for gene regulatory regions, although these methods can be applied to protein coding regions as well.

Some important terms are used in different ways in the literature, so the following paragraph defines them within the context of this study. A conserved character is one that was present in the common ancestral species and has been preserved in the contemporary species being examined. An alignment of the DNA sequences of homologous genes from two related species reveals positions with identical nucleotides. An identical nucleotide at a given position may have been preserved because of selection against change in the sequence, in which case it is important for some function. However, not all conserved characters are functional (6). For instance, orthologous genes are, by definition, descended from the same gene in the last common ancestral species and they will share common sequences even in unselected regions for some period of time. The rate of sequence change is considerably slower in selected regions than in non-selected regions (7)

*To whom correspondence should be addressed at: Department of Biochemistry and Molecular Biology, The Pennsylvania State University, 206 Althouse Laboratory, University Park, PA 16802, USA. Tel: +1 814 863 0113; Fax: +1 814 863 7024; Email: rch8@psu.edu
Present address:
Nikola Stojanovic, The Whitehead Institute, Massachusetts Institute of Technology, Cambridge, MA, USA

and thus after the species have been separated for a sufficient period of time, DNA segments under selection (i.e. functional sequences) will have significantly higher similarity scores than non-selected regions. Various measures for sequence similarity have been used to construct optimal pairwise alignments (8) and robust (but not mathematically optimal) alignments of three or more sequences (9).

Given a set of conserved sequences, one would like to distinguish functional (selected) regions from those whose similarity reflects the residual common ancestral sequence that has not yet changed via evolutionary drift. One approach is to use pairwise alignments of homologous genes from species that separated so long ago that drift has changed all unselected regions. Many studies have used conservation of amino acid sequence in proteins from species as distantly related as yeast and human as one guide to functional assignments. Furthermore, nucleotide sequences conserved in non-coding segments of homologous genes from mice and humans are frequently informative guides to regulatory regions (10). Substantial resolving power is added by including more than two sequences in a multiple sequence alignment, since the likelihood of random column identities in such a multiple alignment is enormously lower than in a pairwise alignment. Since each lineage diverged independently after separation from a common ancestor, the phylogenetic distances covered are effectively additive and thus comparisons among a group of eutherian mammals can show the effects of a much longer period of divergence than the time since they separated from the last common ancestor. Hence, multiple alignments are less likely to show residual similarities in non-selected regions. Of course the true test of functionality must be experimental, so in order to gain the most benefit from computational tools, it would be prudent to try to establish a set of approaches and criteria that are successful in identifying known functional regions within an alignment.

The problem of identifying conserved sequence blocks in multiple alignments is, therefore, critical and the application of computational tools to their detection in long sequences is imperative. A multiple alignment generates a matrix with each DNA sequence occupying a row so that each nucleotide is placed in an appropriate column. A consecutive group of columns, or block, can be identified as conserved based on a number of approaches. The simplest is to compute the level of similarity in each column and find blocks that fit user-defined criteria for the degree of similarity per column and the length of the block. This column agreement approach does not take into account the effects of nucleotide frequency in the genes under consideration and thus Schneider *et al*. (11) developed a metric called information content that incorporates both nucleotide similarities and overall nucleotide composition as a measure of column similarity. These methods are not influenced by the shape of the phylogenetic tree deduced from the contemporary sequences (except to the extent that the multiple alignment itself depends on the order in which the sequences are added) (12), but a method based on minimal evolutionary change uses phylogenetic information to identify conserved blocks. The last two approaches are aimed at finding protein binding sites on DNA. Such sites are usually a series of consecutive positions, one or more of which can vary somewhat without measurably changing the binding affinity. Thus it is desirable to examine a series of neighboring positions in each row when finding blocks. Each row-based method allows up to *k* mismatches per row; in one method the mismatches are relative to a specified 'center' sequence (e.g. the human sequence) and in the other the mismatches are relative to an unknown 'center' sequence. In this latter approach, the unknown center sequence could represent a consensus binding site for some unknown transcription factor. We evaluate the efficacy of each of these methods for finding experimentally determined functional regions within three regulatory regions of mammalian β-globin gene clusters and one bacterial regulatory region. However, they can be applied to any multiple alignment. Moreover, the methods used here could also be used to evaluate and improve the program that generates the alignments.

MATERIALS AND METHODS

DNA sequences and alignments

DNA sequences from the β-globin locus control regions (LCRs) of human (combined GenBank loci HUMHBB and HUMBGLOBC), galago (OCU60902), rabbit (combined GenBank loci OCU63091 and RABBGLOB), goat (GOTGLOBE), cow (BOVBG) and mouse (a collation of loci AF071080, MMMLCRHS4, MMMLCRHS3, MMCONREG and MMBGCXD provided by M. Bender) were aligned using the program *yama2* (12). Sequences and full alignments are available at our Globin Gene Server (13,14) at: http://globin.cse.psu.edu/ . The *Escherichia coli* K-12 sequence is from Blattner *et al*. (15). The sequences of related bacteria were obtained from the following sites: *Salmonella typhimurium*, ftp://genome.wustl.edu/pub/gsc1/sequence/st.louis/bacterial/salmonella/B_STM/B_STM.full.seq; *Salmonella typhi*, ftp://ftp.sanger.ac.uk/pub/pathogens/st/ST.dbs; *Salmonella paratyphi* A, ftp://genome.wustl.edu/pub/gsc1/sequence/st.louis/bacterial/salmonella/B_SPA/BEFORE_MELD/B_SPA.full.seq; *Klebsiella pneumonia*, ftp://genome.wustl.edu/pub/gsc1/sequence/st.louis/bacterial/salmonella/B_KPN/B_KPN.full.seq.

The regions selected for the calibrations of parameters were 7188–7487, 11240–11510 and 64561–64826 (–263 to +3) for HS3, HS2 and the *HBB* promoter, respectively, in the combined human sequence. A given nucleotide position in this sequence is 2687 larger than in GenBank locus HUMHBB. The list of nucleotide positions assigned as functional is at the web site, along with references.

The region selected for calibration against the bacterial *araBAD–araC* regulatory region begins just before the ATG start codon of *araB* (oriented to the left) and ends just before the ATG start codon of *araC* (oriented to the right). This corresponds to positions 70049–70386 in the *E.coli* sequence. To align the five bacterial sequences, the sequences that matched the *araBAD–araC* intergenic region in pairwise comparisons with the *E.coli* sequence were extracted and then aligned simultaneously.

Programs for finding conserved sequence blocks

Certain parameters are common to all of the tools. The minimum length of the regions to be reported and the minimum number of sequences which must be active (i.e. present in that region of the alignment) are selectable by the user. Also, the search can be conducted in the entire alignment or it can be restricted to a portion specified by a given range in

any of the sequences. The results can either be reported as a list of the selected regions' end-points together with their associated consensus/ancestral/center sequences (explained below) or displayed as boxed regions in an alignment diagram. A server running these programs on the alignment of mammalian β-globin gene clusters is available at the Globin Gene Server.

Each utility has at least two ways of dealing with gaps. In the first, columns containing a gap symbol will be rejected, so the reported regions do not contain any gaps. Alternatively, gaps can be treated just like ordinary characters. Ambiguity codes (e.g. W representing A or T) can be permitted in columns. Full details about treatment of gaps and ambiguity codes are available at our web site.

agree. This utility locates regions in a given alignment that have good column agreement. The columns are examined individually to determine whether or not they meet a user-specified threshold for letter agreement, and runs of columns passing this test are reported.

infocon. When searching for conserved regions in alignments, the region's length is often a reliable indicator that some functionality was preserved across the species. However, as the conservation need not be perfect, such regions might be fragmented into conserved pieces too small to be detected, and a systematic way to link the smaller regions is needed. The next two utilities we describe, *infocon* and *phylogen*, attempt to solve this problem. The idea is to assign a numerical score to each column and then look for runs of columns meeting the following two conditions: (i) their cumulative score (obtained by adding together the individual column scores) is no smaller than the score of any of their sub-runs; and (ii) they are maximal with this property, i.e. they are not contained in any longer run having the property (i). We refer to such regions as 'full runs'. Two full runs cannot partially overlap, i.e. if they have a position in common, they must be identical (16).

The *infocon* tool finds full runs of columns with high information content in the given alignment. To do this, each column is assigned an intermediate score that measures its information content, based on the frequencies of the letters both within the column and within the alignment as a whole (11,17). The exact value of this score is the fraction $1/L$ of the logarithm of the likelihood ratio obtained for the frequency of letters within the alignment and within the column under examination, where L is the number of active sequences in the alignment column. As a numerical example, consider the alignment in Figure 1A, which is part of a longer alignment. The overall letter frequencies in the longer alignment (not shown) are f_A = 58 525/192 535, f_C = 36 937/192 535, f_G = 38 963/192 535 and f_T = 58 110/192 535, since the counts of the A, C, G and T letters in the alignment are 58 525, 36 937, 38 963 and 58 110, totaling 192 535. Similarly, the letter frequencies within column 1 of the alignment (C,C,T) are $fc_A = 0$, $fc_C = 2/3$, $fc_G = 0$ and $fc_T = 1/3$. L is 3.

The information content for column 1, which will serve as its intermediate score, can then be computed as:

$$\text{Information content} = 1/3 \log_2(P_{fc}/P_f)$$

where

$$P_{fc}/P_f = (fc_A/f_A)^{nc_A}(fc_C/f_C)^{nc_C}(fc_G/f_G)^{nc_G}(fc_T/f_T)^{nc_T}$$

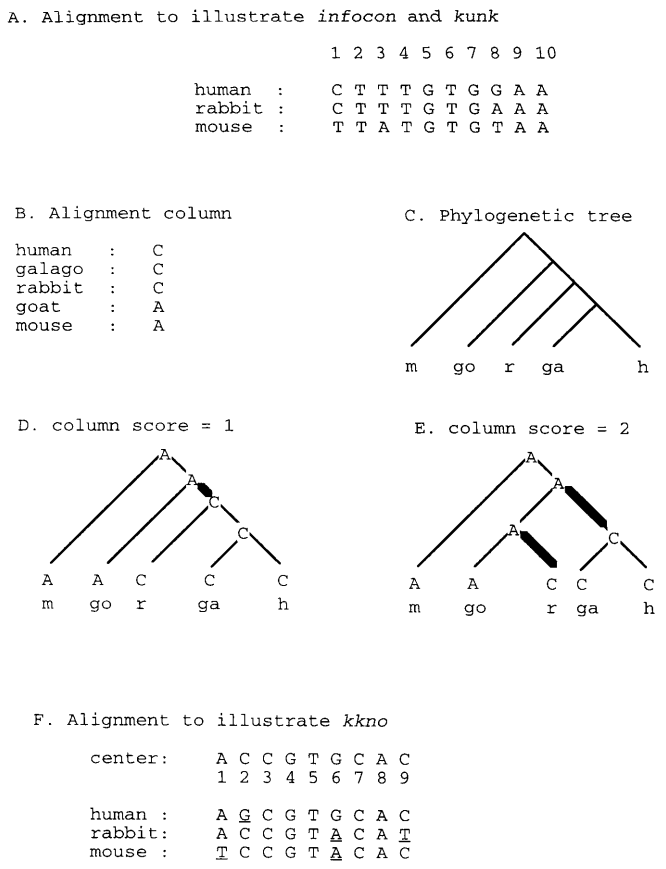

Figure 1. Alignments and trees that illustrate the different methods for finding conserved sequences. (**A**) A hypothetical alignment illustrating *infocon* and *kunk*. (**B–E**) Illustrations of features of *phylogen*: (B) one column of a hypothetical alignment; (C) diagram of the phylogenetic tree used with our *phylogen* tool; (D) illustration of assigning scores in *phylogen*; (E) illustration of the change in column score with a different tree. (**F**) A hypothetical alignment to illustrate *kkno*.

and $nc_A = 0$, $nc_C = 2$, $nc_G = 0$ and $nc_T = 1$ are the letter counts for column 1. The resulting value is 1.2457797 for the information content.

It is imperative that these intermediate scores be adjusted for the results to be relevant. Indeed, as the raw information content is always a positive value, each examined column would increase the cumulative score and be included in the current region, so the entire alignment would be reported erroneously as the result. Consequently, some negative column scores are necessary to separate the regions of interest, i.e. those of high information content. Accordingly, the score is adjusted by subtracting the average per-column information content of the alignment, which is a constant for the alignment under consideration, and/or a user-specified constant, called an anchor value.

phylogen. In the *phylogen* program, columns are scored following the schemes outlined in Fitch (18) and Sankoff and Rousseau (19), based on the evolutionary relationships among the sequences of the given alignment implied by a supplied phylogenetic tree. The phylogenetic tree has a leaf node for each species and each internal node represents a putative

common ancestor for the species in its sub-tree. For each column, *phylogen* assigns to each leaf node the letter from the alignment row of the corresponding species, and labels the internal nodes so as to minimize the total number of changes in the tree. This number is the initial score associated with the column and it is computed as the total edge weight of the labeled tree, where an edge has weight 1 if it corresponds to a letter change, and 0 if it connects two nodes labeled with the same character. The root label is named the 'ancestral' character for the column.

To illustrate the approach used by this program, an optimal assignment of letters to internal nodes for the aligned column in Figure 1B, given the phylogenetic tree in Figure 1C, is presented in Figure 1D. The initial column score is 1 in this case. The optimal assignment and the corresponding score may change if a different tree is used. For instance, one could make rabbit and goat a monophyletic group, as shown in Figure 1E, which results in an increase in the initial column score to 2.

Since well-conserved columns will have low scores, but the selection algorithm is geared toward maximization, the column scores are adjusted by subtracting them from a suitable 'anchor value'. However, as with the *infocon* program, it is essential that both positive and negative scores occur, so the anchor value must be chosen carefully. It can be calculated by the program, as either the current number of active rows for a column or the current number of active rows not containing a gap, or it can be set to an arbitrary non-negative number. Combinations of these values are also possible.

kkno. The *kkno* program scans the alignment to determine, starting at each position, the longest region in which no row differs from a specified, known 'center' sequence in more than *k* positions. The parameter *k*, denoting the number of permitted mismatches, is user-selectable. The known center can be an existing alignment sequence or specified separately.

As an example of applying the *kkno* program allowing 1 mismatch per row ($k = 1$), consider the hypothetical alignment in Figure 1F. Given the center ACCGTGCAC, the longest qualifying regions starting at positions 1 and 2 span the ranges of columns 1–5 and 2–8, respectively. The mismatches in every row are underlined.

kunk. The *kunk* utility is similar to *kkno* except that the center sequence is not known a priori; instead, the program computes the 'best' center sequence for each conserved region it finds. This center sequence can be thought of as belonging to a common ancestor of the species represented in the alignment or as a potential binding site for known or unidentified proteins.

For each column in the alignment, the algorithm recursively examines all possible center sequences starting at that position to see how far the region can be extended and back-tracks when the extension becomes impossible. The quality measure for assessing a potential center sequence is the sum of the squares of the number of mismatches between it and the alignment sequences within the region. A lower value indicates a better candidate for the center sequence. Only characters within a column can be used in the center sequence.

As a consequence, this method is more flexible than *kkno,* in that it allows consecutive letters in the center sequence to be drawn from possibly different alignment rows. Moreover, it allows the letter inhabiting a certain position in the center sequence to vary between applications of the procedure for different starting columns. For instance, for $k = 1$, the best center sequence for the region starting at position 1 (columns 1–7) in the alignment of Figure 1A is CTATGTG, rendering 'A' as the letter representing the alignment column 3 (i.e. occupying the position corresponding to the alignment column 3). In contrast, the best center sequence for the region starting at position 2 (columns 2–10) is TTTGTGTAA, rendering 'T' for the same column. Notice that CTATGTG does not correspond to any of the alignment sequences in the region. If *kkno* were used instead, with the human sequence as center, the regions detected at positions 1 and 2 would extend only up to columns 2 and 7, respectively.

Calibration of parameters for each method against known functional sequences

Underlying our parameter calibration scheme are some remarkable monotonicity properties, formalized in Stojanovic (20). It is intuitive that a larger percentage threshold for column similarity in *agree* or a lower number of permitted mismatches in *kunk* and *kkno* lead to a smaller number and shorter length of the reported regions. However, it is less obvious how the results change when varying the parameter values for *phylogen* and *infocon*. For a fixed required minimum region length, regions obtained by *phylogen* with a larger anchor value always include those obtained with smaller ones (20). Similarly, regions produced by *infocon* decrease in number and extent as the value of the score adjustment parameter increases. The analysis becomes more complex when other parameters are considered, such as the minimum length required for reporting a region or the choice for the flexible anchors in *phylogen*.

To determine good settings for these adjustable parameters, we conducted a series of tests on our multiple alignment of the β-globin gene cluster (5) using the five utilities described and varying the values of the relevant parameters for each method. The goal was to determine the sets of parameter values that would minimize a chosen cost function. Specifically, the cost function was the total count of false positives and false negatives with respect to a set of experimentally determined functional sites. A false positive is a position in the human sequence that does not belong to any of the known functional sites but was reported by the program under examination. A false negative is a position in the human sequence that belongs to a known functional site but was not reported by the program.

Two types of assessments were performed: per region assessments, targeted towards HS2, HS3 and the *HBB* promoter individually, and overall assessments (examining all three regions in the same test). In the latter case, the goal was to find the set of parameter values that would produce a lowest aggregate total cost for the regions considered.

The optimal sets of parameter values for each utility differed for each region examined and are listed in Table 1. The optimal sets were determined as described in the following.

For the *agree* utility, values of the parameter *l* (required minimum region length) over the range 3–25 were tested for values of *p* (percent identity threshold) ranging from 10 to 100% in increments of 1%. The number of false positives and false negatives varied monotonically with *p*, as the method achieved smaller coverage with increasing *p* values. Tests were

Table 1. Parameter values that produce lowest costs in the *HBB* locus

Region	Program	*l*	Column agreement	Anchor	*k*	FP	FN	Cost	FP max	FN max
HS2	*agreeG*	16	60%	na	na	67	7	74	138	133
HS2	*agreeX*	13	60%	na	na	51	21	72	138	133
HS2	*infocon*	16	na	0.900	na	49	2	51	138	133
HS2	*phylogen*	9	na	1.300	na	61	0	61	138	133
HS2	*kkno*	5	na	na	1	31	38	69	138	133
HS2	*kunk*	7	na	na	1	19	51	70	138	133
HS3	*agreeG*	4	100%	na	na	15	35	50	227	73
HS3	*agreeX*	4	100%	na	na	15	35	50	227	73
HS3	*infocon*	7	na	1.191	na	12	37	49	227	73
HS3	*phylogen*	5	na	0.300	na	17	32	49	227	73
HS3	*kkno*	9	na	na	1	25	26	51	227	73
HS3	*kunk*	9	na	na	1	22	33	55	227	73
*HBB*_pr	*agreeG*	8	80%	na	na	43	30	73	173	93
*HBB*_pr	*agreeX*	6	80%	na	na	40	13	53	173	93
*HBB*_pr	*infocon*	6	na	1.101	na	8	31	39	173	93
*HBB*_pr	*phylogen*	6	na	0.740	na	14	23	37	173	93
*HBB*_pr	*kkno*	7	na	na	1	10	56	66	173	93
*HBB*_pr	*kunk*	7	na	na	1	13	35	48	173	93
Combined	*agreeG*	3	100%	na	na	51	176	227	538	299
Combined	*agreeX*	11	80%	na	na	31	184	215	538	299
Combined	*infocon*	6	na	1.056	na	89	73	162	538	299
Combined	*phylogen*	6	na	0.633	na	56	106	162	538	299
Combined	*kkno*	8	na	na	1	43	164	207	538	299
Combined	*kunk*	7	na	na	1	64	113	177	538	299

l, minimum block length; *k*, number of mismatches allowed per row; *HBB*_pr is the promoter for the β-globin gene. The program *agree* was run in the gap-inclusive (*agreeG*) or gap-exclusive (*agreeX*) modes; all other programs were run in the gap-exclusive mode.

run separately for the gap-inclusive (mode = G) and gap-exclusive (mode = X) cases.

The *infocon* utility was tested with values of the parameter *l* over the range 3–25 and values of *a* (anchor value or score adjustment parameter) ranging from 0 to 2.0 in increments of 0.001. The maximum information content for a column in the alignment of β-globin gene clusters is 1.65, and thus 2.0 is a reasonable value for the maximum anchor. As *a* became larger, the number of false positives decreased and the number of false negatives increased, as the regions obtained for larger *a* values were included in those obtained for smaller values. For each value of *l*, we partitioned the range [0,2.0] of possible score adjustment values into intervals so that within each interval the number of false negatives and the number of false positives did not vary. Then we selected the best *a* interval for every length *l* and the best overall pair of values for *a* and *l*.

The *phylogen* utility was tested for values of the parameter *l* over the range 3–25 and for a range of values of *a* (a user-specified fixed anchor value). The value *a* was varied over the range 0–4, which is the maximum phylogenetic distance for an alignment of five sequences, in increments of 0.001. The number of false positives increased and the number of false negatives decreased as *a* became larger. As before, a partition of the range of *a* values was produced for each value of *l* and the best *a* intervals and best overall (*a*,*l*) pair were determined according to the cost criterion.

For the utilities *kkno* and *kunk*, the number of mismatches allowed, *k*, was fixed at 1 and the parameter *l* was varied from 3 to 25.

RESULTS

Rationale for the five methods

Column agreement. Detection of conserved blocks is straightforward if no sequence variations are allowed in the criterion for ‘conserved’. One simply finds all blocks composed of a string of invariant columns of a designated minimum length. Although useful in some cases, this approach can miss some important protein-binding segments (Fig. 2A). For example, consider the underlined sequence AGATAG at position 7405 in this part of HS3 in the human β-globin LCR: the protein GATA1 can bind at this site (21), it is occupied by a protein *in vivo* (22) and this region contributes to the function of HS3

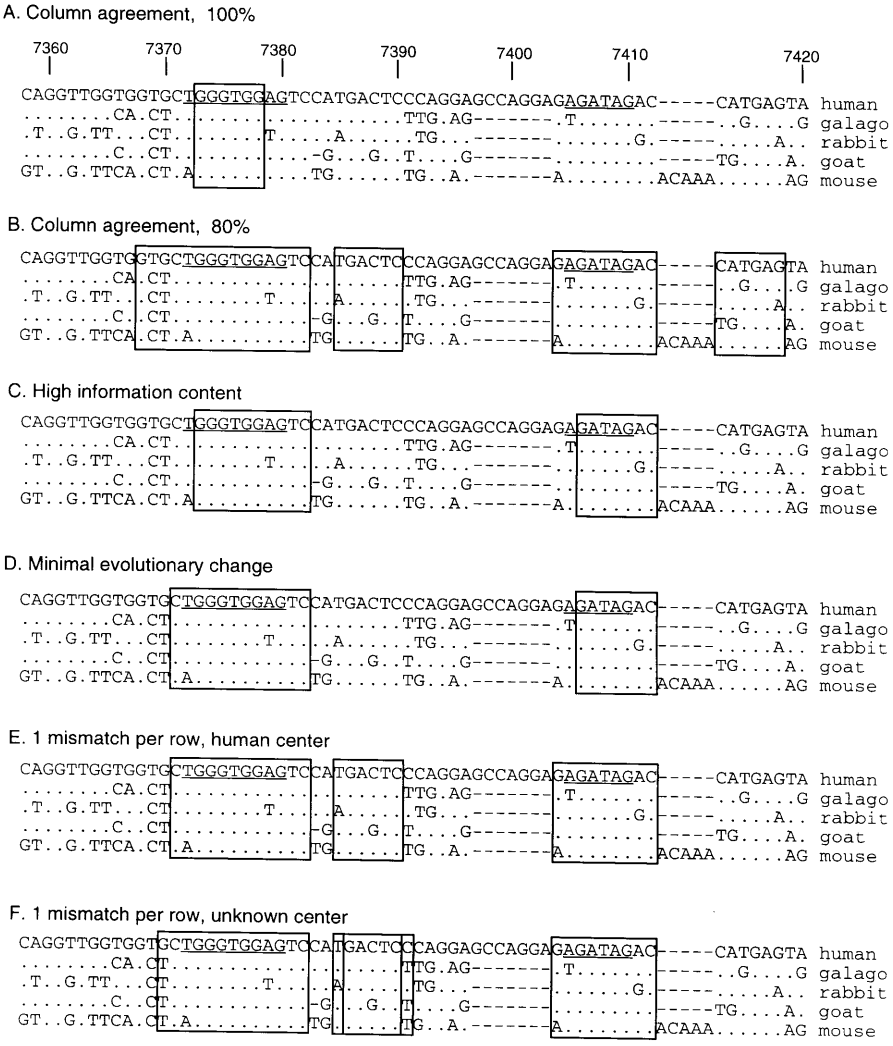

Figure 2. Sample results from the five methods for finding conserved blocks. An alignment of the human β-globin LCR sequence and a few of its eutherian homologs is shown for positions 7358–7420 (part of HS3), with boxes drawn around the conserved blocks determined by each method. Variation in the parameters was minimized; all blocks have a minimum length of 6 and are gap free. Hence, these parameters are not optimal for matching known functional sequences (see Fig. 4). (**A**) *agree*, column agreement 100%; (**B**) *agree*, column agreement 80%; (**C**) *infocon*, anchor value 1.174 (the average information content for the entire alignment); (**D**) *phylogen*, anchor value 0.5; (**E**) *kkno*, $k = 1$; (**F**) *kunk*, $k = 1$.

(23). However, this site is not detected as conserved if one searches for invariant blocks of length greater than 5. The preferred binding sites for GATA1 allow for an A or T at the first position of the WGATAR consensus sequence (24,25) and, indeed, the galago sequence has an A→T transversion at this position. This, plus another substitution just 3′ to the consensus binding site in rabbit, restricts the number of consecutive invariant columns to 5. The fact that some transcription factors have comparable binding affinities for different sequences means that one should allow limited nucleotide substitutions in the algorithm for detecting conserved blocks. Although this GATA1 site is detected by restricting the length of the block to five or less, this is sufficiently short that the likelihood of false positives may become unacceptable. Also, other transcription factors, such as basic helix–loop–helix proteins, have ambiguities in the center of their preferred binding site CANNTG (26), which reduces the string of invariant columns to an unacceptably short length. Thus we developed a program, called *agree*, for finding strings of columns that meet an adjustable level of agreement. For instance, out of the five sequences used in the alignment in Figure 2B, we allow one mismatch in each column, i.e. require at least 80% agreement. This detects the highly conserved block containing GGGTGG and the GATA1 site, along with two other blocks. However, the block containing CATGAG in the human sequence has a substitution in four of its six columns in the non-human species. This illustrates the concern that this simple adjustment to allow substitutions in every column may be too lenient and allow the detection of an excessive number of false positives. Indeed, allowing a single mismatch per column enormously increases the number of

potentially spurious blocks detected in the 17 kb of the β-globin LCR (5,27).

High information content. Finding blocks with high information content (11,17) has the advantage of using a more finely graded metric than the substantial difference between allowing none versus one mismatch per column, and it also takes into account the overall nucleotide composition of the sequences being examined. Our program, called *infocon*, for detecting blocks with high information content finds blocks of a designated minimum length whose average information content per column exceeds a user-adjustable value or anchor value. As shown in Figure 2C, this method finds the blocks containing GGGTGG, which is likely a binding site for EKLF (28), and GATA, without the additional blocks detected by 80% column agreement.

Minimal evolutionary change. When the same substitution is present in more than one sequence from different species in an alignment, it could result from a mutation in the common ancestor to those species, in which case it should be counted only as a single alteration, or it could result from independent mutations after the species diverged, in which case it should be counted as multiple alterations. This argument can be incorporated into the analysis if the phylogenetic relationships among the species being examined are known with considerable certainty. We used a tree that groups human with galago, since both are primates, then has lagomorphs (rabbit) diverging from them fairly recently, preceded by the divergence of artiodactyls (goat), with rodents (mouse) as the earliest order to diverge from other eutherians (Fig. 1C). This phylogeny is supported in a number of studies (29,30), although the relative order of divergence of the lagomorph and artiodactyl lineages is still an open issue.

Our program for finding blocks of minimal evolutionary change based on a given phylogenetic tree, called *phylogen*, computes the minimum number of changes required to account for the contemporary sequences and subtracts that value from a user-specified 'anchor value' (see Materials and Methods for details). Our utility then reports blocks of maximal extent whose scores are larger than or equal to the scores of any of their sub-blocks. In the example shown in Figure 2D, *phylogen* identifies two blocks, one encompassing the GGGTGG motif and the other capturing most of the GATA motif.

Blocks that conform to a known center. The three previous methods compute some score for each column with no regard for the entries in nearby columns (except for the value of overall base composition used by *infocon*). However, one would expect the binding site for a particular protein to vary in a limited number of positions between species, since proteins will often bind to several similar sequences. Thus one would like to find blocks in which each row differs from the preferred binding site in no more than *k* positions per row, regardless of the columns at which these differences occur. For this and the next method, we will examine the cases for $k = 1$, but this is an adjustable parameter. The preferred binding site may or may not be known and thus we have developed tools to find blocks of sequences that conform within *k* mismatches to either a known or unknown (see next section) comparator sequence. The sequence to which the individual rows are compared is called the center sequence. The application of the program for finding blocks conforming to a known center (called *kkno*) to search for blocks that differ in only one mismatch per row from the human sequence is illustrated in Figure 2E. The blocks containing GGGTGG and GATA motifs are captured. One of the 'extra' blocks found by the 80% agreement approach in Figure 2B is also found here, but the block just 3′ to the GATA motif is not detected since it contains more than one difference in the goat sequence.

Blocks that conform to an unknown center. Often the actual proteins binding to a particular site or even the preferred binding site for characterized proteins are not known, i.e. the desired center sequence is unknown prior to the analysis. The *kunk* program will identify blocks that differ by no more than *k* mismatches from an a priori unknown center sequence (31). Effectively, it tries to find a sequence of designated minimum length such that each row of the block differs at no more than *k* positions from it. Of course, after the analysis the center is known and can be reported to the user. For the aligned sequences analyzed in Figure 2F, this approach includes one additional column in the block containing GGGTGG. All the species except human have a T in this initial column of the block and thus the center sequence will be chosen with a T at this position.

Comparative evaluation of the five methods

The actual results of analysis by each method are determined by the choice of parameters. Some adjustable parameters are common to all methods, such as the minimum length of the block (*l*), the number of sequences that must be active and whether gaps can be included in the block (gap-inclusive versus gap-free blocks). Other parameters apply solely to a particular method, such as the level of column agreement in *agree*. The user can adjust each of these parameters, so that each method can return a wide spectrum of results for any given alignment, ranging from very few columns to nearly all columns. Consequently, the sets of possible results from the five methods show considerable overlap.

Thus the choice of parameter values is a key determinant of the efficacy of each method. To 'calibrate' the programs, we initially compared their output with a set of known functional sequences from three intensively studied regulatory regions in the β-globin gene cluster: HS2 and HS3 in the LCR and the *HBB* promoter. Using this set of experimentally identified sites as a standard, we adjusted each program's parameters to make its output match the desired set as closely as possible by minimizing the cost, which is the sum of the false positives and false negatives it reported.

For example, the results of the optimization for *infocon*'s anchor value are shown in Figure 3. The regions encompassing HS2, HS3 and the *HBB* promoter were examined by increasing the anchor value in small increments over a wide range, holding the minimum length constant at the best value for a particular region. The resulting columns for each anchor value were compared to the reference set of known functional sequences (see below). A clear minimum cost can be seen at a certain anchor value for each of the three regions. Interestingly, a slightly different anchor value and a different minimal cost is obtained for each region.

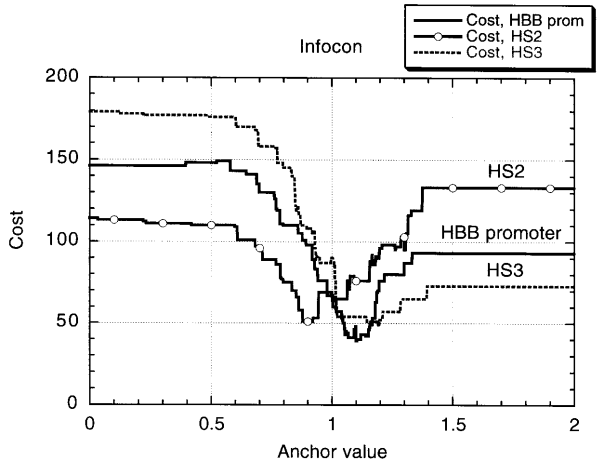

Figure 3. A typical calibration curve, showing the costs of results returned by *infocon* while varying the anchor value. The anchor value was varied over the range 0–2 in increments of 0.001, holding the minimum length *l* constant at the best value for each region. Each line has 2000 data points.

Parameter calibration using HS2. The core of HS2 has been analyzed by *in vivo* footprints (32–35), effects of mutations (36–38) and *in vitro* protein binding (39,40). Many other papers have been published on this subject, but the cited ones cover all the demonstrated functional regions within the core of HS2.

The outputs for each utility, at parameter values that produce the closest match to the set of functional sites (Table 1), are plotted in Figure 4A. All of the programs return the best-characterized functional sites, including the MAREs (binding sites for NFE2 and related proteins), one of the GATA motifs, an invariant E box (including position 11390) and a GGGTG motif. Four of the five tools return at least part of every functional region. The most comprehensive coverage was obtained by *infocon* and *phylogen*, which produced almost identical output with these optimized parameters. These two programs produced the lowest cost results as well (Table 1). The *kunk* program did not identify two of the functional regions (one of the GATA motifs and the E box at 11450). Optimized results from the program *agree* had the highest costs, in both the gap-inclusive (*agreeG*) and gap-exclusive (*agreeX*) modes.

The two methods based on comparisons of rows to a center sequence, *kkno* (with human center sequence) and *kunk*, returned conserved blocks of shorter length than the other methods. For instance, the upstream functional region extends from 11276 to 11304, but only a short segment of that is identified by either *kkno* or *kunk*. Perhaps the rest of the functional region, which is found by *agree*, *infocon* and *phylogen*, is involved in some aspect of regulation that is not well modeled by our current expectations for protein-binding sites.

Some regions were selected as conserved by all of the methods but have not been characterized functionally to date. For example, part or all of a 23 bp fragment located at positions 11424–11446 in the alignment is returned by all the methods. The fact that several tools select this region independently makes it an excellent candidate for experimental analysis to determine whether it is indeed functional.

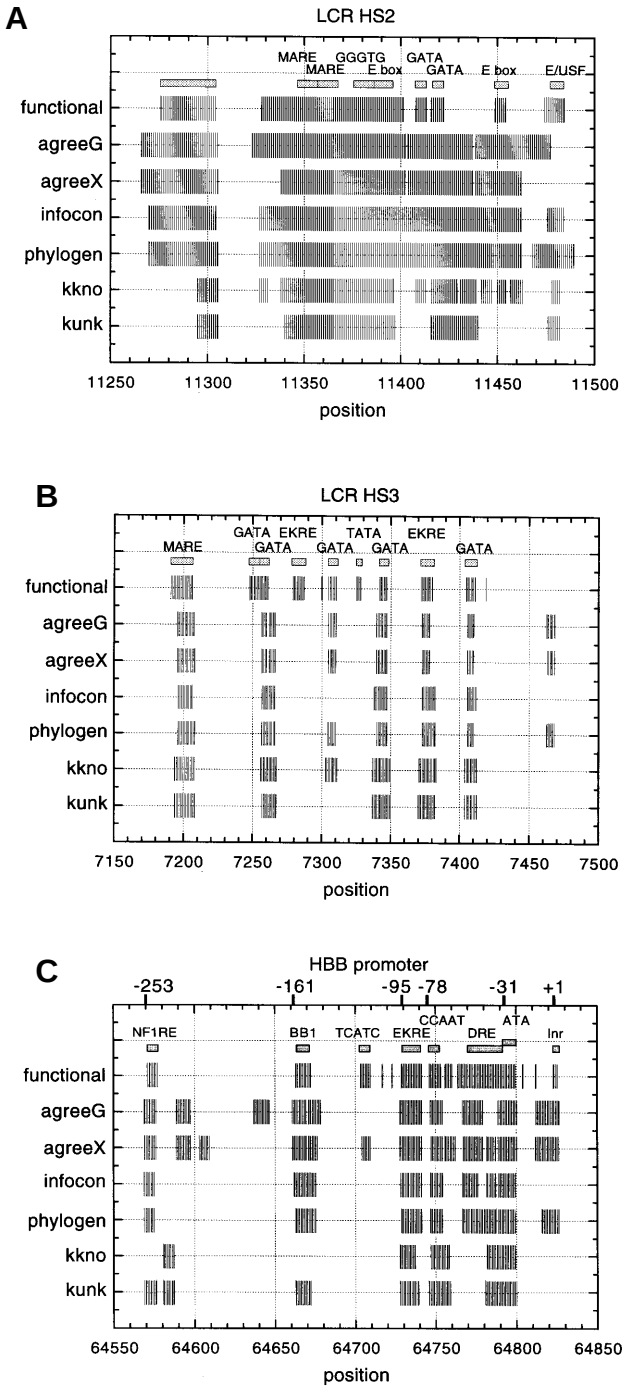

Figure 4. Graphs of the positions of blocks identified by the five programs after calibration of parameters against known functional regions in the β-globin gene cluster. The positions shown experimentally to be functional are marked on the line so indicated; boxes are labeled with sequence motifs or other identifiers. Additional lines show the positions of blocks found by each method. The parameter values are listed in Table 1. The program *agree* was run in the gap-inclusive (*agreeG*) or gap-exclusive (*agreeX*) modes; all other programs were run in the gap-exclusive mode.

Parameter calibration using HS3. HS3 is associated more with opening a chromatin domain than with enhancement (21,41–44) and thus it may show a different pattern of conser-

vation than the HS2 enhancer. We calibrated the results of the five methods against the *in vivo* footprints for HS3 (22), adjusted to include the consensus binding sites for proteins implicated as acting at these sites. Other functional data on HS3, such as mutation results and *in vitro* footprints, covered such extensive regions of the HS3 core that they were not effective in defining useful parameters for our tools.

The results summarized in Figure 4B show remarkably consistent results for all five methods. Likewise, the associated costs of the optimized results of all methods were very close (Table 1). All of the methods detected the MARE, three GATA motifs and one CACC motif (labeled EKRE) that is likely a response element for EKLF (28). In addition, *agree*, *phylogen* and *kkno* detected a fourth GATA motif. None of the methods returned the GATA motif centered at 7250, a putative EKRE centered at 7284, the TATA motif or the two isolated nucleotides detected by *in vivo* footprinting. All of these latter functional sites are found only in the human sequence (see alignments at http://globin.cse.psu.edu or in ref. 5) and thus would not be expected to be identified by tools seeking conserved sequences.

The minimal evolutionary change approach, *phylogen*, performed very similarly to *agree* and *kkno* in this example (Fig. 4B), whereas *phylogen* produced results very similar to *infocon* in the analysis of HS2 (Fig. 4A).

Both *agree* and *phylogen* detected a block between 7462 and 7469 (GCATTTTT in the human sequence) that was not examined in the *in vivo* footprinting or mutagenesis studies. In fact, it lies just 3′ to the restriction endonuclease cleavage site used in defining the minimal core for HS3 (21). It is clearly conserved, with the CATTTT being invariant in the five species analyzed (hence it would be detected by *kkno* and *kunk* with $l = 6$). DNA fragments containing this sequence bind *in vitro* to YY1, GATA1, NFE2, Oct1 and an unidentified protein (45). This is an example of a conserved block warranting further functional study.

Parameter calibration using the HBB promoter. It is possible that promoters might show different patterns of sequence conservation than enhancers or other regulatory elements, so the five methods were also calibrated against the promoter for *HBB*. This promoter is among the most intensively studied for any mammalian gene, including considerable data from naturally occurring thalassemia mutations (46), close to saturation mutagenesis *in vitro* (47–51), *in vitro* footprints (52), *in vivo* footprints (34) and analysis of specific activator proteins such as EKLF (53). A summary of the functional sites reported in these studies is shown in Figure 4C, along with the positions detected as conserved by each of the methods at their optimal parameter settings.

In the *HBB* promoter, each of the methods produces a distinctive set of results, in contrast to the rather homogeneous results seen for HS2 and HS3. All methods detect four of the functional regions, i.e. EKRE, CCAAT, part of the direct repeat element (DRE) and the ATA motif (recognized by TBP/TFIID). Also, other important motifs, such as the response element for NF1, BB1 and the initiator, are detected by most but not all methods. The utility *agree* in the gap-exclusive mode detected at least part of all functional regions, but it also detected two additional regions not implicated in function (centered around 64595 and 64605). All the other methods missed the TCATC motif, which is conserved in most species but has a 3 nt substitution in the galago sequence. Thus in order for other methods to detect it, the parameters would have to be relaxed from the optimal settings. The programs *infocon* and *phylogen* produced results with the lowest costs (Table 1).

The utility *kunk* performed better than *kkno* at the *HBB* promoter, whereas *kkno* produced better results than *kunk* at HS2 and HS3. We had expected that *kunk*'s flexibility to choose the center sequence would make it the better tool, thereby justifying its added complexity. It appears that this will be true for some but not all regions.

Neither *kunk*, *kkno* nor *infocon* detected the initiator region, encompassing the nucleotide encoding the capped nucleotide of the mRNA. Since the initiator is recognized by components of TFIID (54) that may be directed to this segment of the DNA template by binding to a different site, i.e. the TATA box, this region may not be easily detectable by methods based on expectations for direct protein binding.

Compared to the results for HS2 and HS3, the pattern of conserved sequences makes a less compelling case for additional functional regions in the *HBB* promoter. This doubtless reflects the very intensive experimental analysis of this promoter over the course of 20 years and the variety of techniques used. Despite this, both *kkno* and *kunk* reveal an additional conserved block centered around 64585, suggesting that even at this promoter the identification of functional regions may not be complete.

Application of the methods to a control region in eubacteria. We also wished to test the efficacy of these tools in a different gene system and set of organisms. The complete sequence of *E.coli* K-12 has been determined (15) and recently the genomic sequences of four related eubacteria, i.e. three *Salmonella* spp. and *K.pneumonia*, have been determined at ~2-fold shot-gun coverage (see Materials and Methods for ftp sites). This presents an opportunity to explore the efficacy of these tools for different genes. The estimated time of divergence of these eubacteria is ~100 million years ago (55), close to the estimates for the divergence of eutherian mammals (29). The intergenic region between *araBAD* and *araC* was chosen as a well-studied regulatory region, with two oppositely oriented σ70 promoters and several experimentally defined binding sites for AraC and CRP (56–58). Protein coding regions were excluded from the analysis.

A reference set of functional sequences was selected based on information in RegulonDB (59,60) and the Genomes section of Entrez at NCBI (61), which have been underlined in Figure 5. We then applied the five methods for finding highly conserved sequences to this region, optimizing the parameters to find the closest matches to the reference set of sites. All the methods worked well after optimization (Table 2). Indeed, the percentage of errors (or cost) to total length of the region (FPmax+FNmax) is only ~10%, whereas it was higher for the analysis of the *HBB* locus (Table 1). As expected, *phylogen* has one of the best scores. The three species from the same genus are more likely to share a sequence distinctive from that of other genera, and *phylogen* would count any *Salmonella*-specific nucleotides as a single change. As illustrated in Figure 5, all the functional regions except the –35 box for *araBAD* are captured and all the false positives are adjacent to known functional regions.

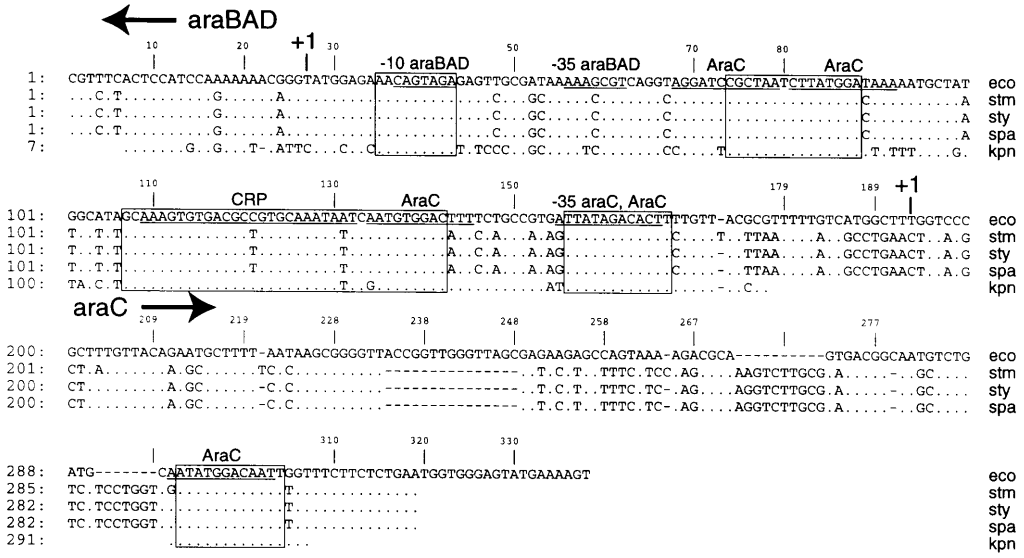

Figure 5. Results of using *phylogen* with optimized parameters to find highly conserved blocks in the control region of the bacterial *araBAD* and *araC* operons. The known functional sequences listed in RegulonDB and in Entrez Genomes are underlined and labeled above the set of aligned sequences. AraC and CRP refer to binding sites for these proteins, and the –10 motif of the *araBAD* promoter and the −35 motifs of both promoters are underlined. Boxes are drawn around the blocks identified by *phylogen*. Species names are abbreviated as follows: eco, *E.coli*; stm, *S.typhimurium*; sty, *S.typhi*; spa, *S.paratyphi* A; kpn, *K.pneumonia*.

Table 2. Parameter values that produce lowest costs in the bacterial *araBAD–araC* regulatory region

Program	*l*	Column agreement	Anchor	*k*	FP	FN	Cost	FP max	FN max
agreeG	8	100%	na	na	7	26	33	240	98
agreeX	8	100%	na	na	7	26	33	240	98
infocon	11	na	1.230	na	11	20	31	240	98
phylogen	8	na	0.200	na	8	22	30	240	98
kkno	12	na	na	1	21	9	30	240	98
kunk	16	na	na	1	27	8	35	240	98

The optimal parameters for this region differ from those for the β-globin LCR or the *HBB* promoter (compare Tables 1 and 2). The reasons for these differences have not been elucidated, but they could have more to do with the particular region than any differences between mammalian versus bacterial control sequences. For instance, the 5′-untranslated region of *araBAD* is considerably more conserved than is the comparable region of *araC* (Fig. 5), but those invariant positions with no assigned function in the databases are assigned as false positives in the cost function. Further analysis could reveal function in these regions. In general, distinct optimal parameters are found for different regulatory regions.

Effects of allowing gaps

While the gap-free mode is useful for providing high resolution views, e.g. of potential binding sites for proteins, it cannot find regions where variations among the sequences are due to insertions or deletions rather than nucleotide substitutions. This becomes a significant concern when one acknowledges that sequencing errors do occur, including misreading the number of nucleotides in a string (e.g. GG instead of GGG). This of course produces gaps in the alignment. Allowing gaps to appear in conserved blocks thus makes our tools more tolerant of sequencing errors. When compared to the results with the gap-exclusive mode while maintaining other parameters the same, the use of the gap-inclusive mode will fuse clusters of neighboring gap-free blocks, which may make the potential functional regions more obvious. Even when using optimal parameters for the *agree* utility in the gap-inclusive (*agreeG*) and gap-exclusive (*agreeX*) modes, longer runs of conserved columns were detected in the gap-inclusive mode for HS2 and the BB1 site of the *HBB* promoter (Fig. 4A and C).

DISCUSSION

One goal of bioinformatics is to build tools that can identify regions with a high level of similarity among homologous sequences and thereby find strong candidates for functional sites. Since the level of similarity depends on the species examined and can vary among different genes and even different regions within genes and genomic segments, these tools must be flexible enough to support varying cut-offs along the similarity continuum. Ideally, the tools would be calibrated separately for each region of interest to find settings that produce good results when compared with the set of sites known to be functional. However, in most cases one will search for conserved blocks in previously unstudied regions to find strong candidates for functional sequences. Thus calibration of the computer tools is impossible in such regions, but the results obtained here for four regulatory elements in both mammals and bacteria could be a useful guide for initial studies. Of course, computational methods will never provide definite proof of evolutionary selection: that is a biological question. The challenge on the experimental end of functional genomics is to hypothesize reasonable functions for sequence regions that can be tested in the laboratory.

Our analysis of different methods was prompted by the realization that no single definition of conservation is adequate to cover all possible purposes. Thus we explored a set of approaches, each based on a different rationale. Full column agreement can be used to find the most highly conserved segments, but it is too stringent to find all binding sites. Blocks with minimal evolutionary change or high information content can detect known functional regions effectively by allowing some mismatches in the alignment. The two row-based utilities search for close matches to 'center' sequences that are either specified or unknown a priori. All five of the methods can return a set of blocks that is close to the set of experimentally determined functional sequences in the four regions that we investigated, provided one uses optimal parameters. In general, highly conserved motifs are detected by each of the methods, albeit with slightly differing end-points. For HS2 and HS3, the methods revealed some consistently conserved blocks that did not match any of the known functional sites and therefore may be deserving of further functional study. Conversely, for all four regulatory regions, some segments that are known to interact with proteins are not strongly conserved among the species we investigated.

No one method appeared clearly superior to the others and, indeed, the fact that these independent approaches produce such similar results strengthens the case for their validity. The goals and viewpoint of the investigator can dictate choice among the various methods. For example, users studying protein binding may choose the row-based tools, those interested in entropy may wish to use information content and those with a phylogenetic perspective may prefer the approach based on evolutionary change. Easy availability of these utilities should encourage use of and comparison among multiple approaches.

However, it would be premature to conclude that the five approaches do not differ significantly in their effectiveness. The alignments of the four regulatory regions chosen for the calibration study are clearly well conserved and have been recognized as such by a number of approaches, including visual inspection. The four regions examined in this study were chosen because of the substantial body of experimental results against which we could calibrate the parameters for our programs. However, the advantages for each individual program may become clearer as they are applied to additional functional regions.

Obtaining good results with *agree*, *infocon* and *phylogen* required calibration against the data set of known functional regions, since it is very difficult a priori to predict the best values for the relevant parameters, such as anchor values. The optimal parameter values for *agree* differed considerably among the regions used for calibration. The optimal values for the minimum length l ranged from 3 to 16 for the different regions and the column agreement ranged from 60 to 100% (Tables 1 and 2). There is no obvious rationale for these changes in the optimal parameter values. The programs *infocon* and *phylogen* invariably returned results with the best scores (lowest costs in Table 1), but again it is difficult to predict a priori the optimal anchor value. The optimal anchor value varied considerably for different regions analyzed by *phylogen*, but it is more consistent for *infocon*, ranging only from 0.9 to 1.2. Thus one may expect, based on our calibrations, that using *infocon* with $l = 6$ and $a = 1$ will return good results in many cases. Alternatively, one can choose the two parameters for both *kkno* and *kunk* based on objective expectations. For instance, an investigator may be interested in potential binding sites of minimum length 10, but may be willing to accept only 1 mismatch per row. Thus the parameters $l = 10$ and $k = 1$ can be chosen without calibration. The sequences of many genomic regions will soon be available, but with no previously determined functional regions available for calibration. In these cases, the predictability of parameter values for the programs *kkno* and *kunk* will be advantageous.

Further development of these approaches could improve their power or applicability. For instance, any of the utilities could be linked to a transcription factor database to allow one to search for all blocks whose consensus/ancestral/center sequence matches a known binding site. This would be most effective for the set of transcription factors with well-known binding sites. The *phylogen* utility could be made more sophisticated by providing a scoring scheme that discriminated among transitions, transversions and insertions/deletions. Our approach of first making an alignment and then searching for highly conserved sequences has some limitations. Obviously, the efficacy of the tools depends on the quality of the alignment, but the multiple alignment program does not guarantee an optimal solution. Hence, it is possible that some important motifs could be missed. Naturally, the same ideas could be used to evaluate the procedure that generates the alignments. An alternative approach would be to combine the identification of interesting motifs with the alignment procedure.

While a careful analysis has characterized pairwise alignments of protein coding regions between human and rodent sequences (62), alignments of functional non-coding genomic regions are less well understood. Here we have taken a step in that direction by studying experimentally confirmed regulatory elements in the context of a fixed multiple alignment of genomic sequence data, both from several orders of mammals and several genera of bacteria. Our study suggests that a wide variety of approaches effectively identify conserved regions

and, when optimally calibrated, their results are similar in practice.

AVAILABILITY

An interface for using these tools to find conserved blocks within the aligned mammalian β-globin gene cluster, as well as additional material including source code for the programs, is located at the Globin Gene Server (http://globin.cse.psu.edu/) under the section on Multiple Alignments.

ACKNOWLEDGEMENTS

This work was supported by the National Library of Medicine, grants RO1LM05110 and RO1LM05773, and National Institutes of Health grant RO1DK27635.

REFERENCES

1. Lander,E.S. (1996) *Science*, **274**, 536–539.
2. Boguski,M.S., Hardison,R.C., Schwartz,S. and Miller,W. (1992) *New Biol.*, **4**, 247–260.
3. Pietrokovski,S., Henikoff,J.G. and Henikoff,S. (1998) *Trends Genet.*, **14**, 162–163.
4. Gumucio,D., Shelton,D., Zhu,W., Millinoff,D., Gray,T., Bock,J., Slightom,J. and Goodman,M. (1996) *Mol. Phylogenet. Evol.*, **5**, 18–32.
5. Hardison,R., Slightom,J.L., Gumucio,D.L., Goodman,M., Stojanovic,N. and Miller,W. (1997) *Gene*, **205**, 73–94.
6. Mayr,E. (1982) In Eldredge,N. and Gould,S.J. (eds), *Systematics and the Origin of Species*, Columbia Classics in Evolution Series. Columbia University Press, New York, NY.
7. Nei,M. (1987) *Molecular Evolutionary Genetics*. Columbia University Press, New York, NY.
8. Needleman,S.B. and Wunsch,C.D. (1970) *J. Mol. Biol.*, **48**, 443–453.
9. Higgins,D.G., Bleasby,A.J. and Fuchs,R. (1992) *Comput. Appl. Biosci.*, **8**, 189–191.
10. Hardison,R., Oeltjen,J. and Miller,W. (1997) *Genome Res.*, **7**, 959–966.
11. Schneider,T., Stormo,G., Gold,L. and Ehrenfeucht,A. (1986) *J. Mol. Biol.*, **188**, 415–431.
12. Chao,K.-M., Hardison,R. and Miller,W. (1994) *J. Comput. Biol.*, **1**, 271–291.
13. Hardison,R., Chao,K.-M., Schwartz,S., Stojanovic,N., Ganetsky,M. and Miller,W. (1994) *Genomics*, **21**, 344–353.
14. Hardison,R., Riemer,C., Chui,D.H.K., Huisman,T.H.J. and Miller,W. (1998) *Genomics*, **47**, 429–437.
15. Blattner,F.R., Plunkett,G., Bloch,C.A., Perna,N.T., Burland,V., Riley,M., Collado-Vides,J., Glasner,J.D., Rode,C.K., Mayhew,G.F., Gregor,J., Davis,N.W., Kirkpatrick,H.A., Goeden,M.A., Rose,D.J., Mau,B. and Shao,Y. (1997) *Science*, **277**, 1453–1474.
16. Zhang,Z., Burman,P., Wiehe,T. and Miller,W. (1999) *Bioinformatics*, in press.
17. Stormo,G. (1990) *Methods Enzymol.*, **183**, 211–221.
18. Fitch,W. (1971) *Syst. Zool.*, **20**, 406–416.
19. Sankoff,D. and Rousseau,P. (1975) *Math. Program.*, **9**, 240–246.
20. Stojanovic,N. (1998) Ph.D. thesis, The Pennsylvania State University.
21. Philipsen,S., Talbot,D., Fraser,P. and Grosveld,F. (1990) *EMBO J.*, **9**, 2159–2167.
22. Strauss,E.C. and Orkin,S.H. (1992) *Proc. Natl Acad. Sci. USA*, **89**, 5809–5813.
23. Philipsen,S., Pruzina,S. and Grosveld,F. (1993) *EMBO J.*, **12**, 1077–1085.
24. Ko,L.J. and Engel,J.D. (1993) *Mol. Cell. Biol.*, **13**, 4011–4022.
25. Merika,M. and Orkin,S.H. (1993) *Mol. Cell. Biol.*, **13**, 3999–4010.
26. Blackwell,T.K. and Weintraub,H. (1990) *Science*, **250**, 1104–1110.
27. Slightom,J., Bock,J., Tagle,D., Gumucio,D., Goodman,M., Stojanovic,N., Jackson,J., Miller,W. and Hardison,R. (1997) *Genomics*, **39**, 90–94.
28. Tewari,R., Gillemans,N., Wijgerde,M., Nuez,B., von Lindern,M., Grosveld,F. and Philipsen,S. (1998) *EMBO J.*, **17**, 2334–2341.
29. Li,W.-H., Gouy,M., Sharp,P., O'Huigin,C. and Yang,Y.-W. (1990) *Proc. Natl Acad. Sci. USA*, **87**, 6703–6707.
30. Hedges,S.B., Parker,P.H., Sibley,C.G. and Kumar,S. (1996) *Nature*, **381**, 226–229.
31. Stojanovic,N., Berman,P., Gumucio,D.L., Hardison,R.C. and Miller,W. (1997) In Dehne,F., Rau-Chaplin,A., Sack,J.-R. and Tamassia,R. (eds), *Algorithms and Data Structures*. Springer, New York, NY, Vol. 1272, pp. 126–135.
32. Ikuta,T. and Kan,Y.W. (1991) *Proc. Natl Acad. Sci. USA*, **88**, 10188–10192.
33. Reddy,P.M.S. and Shen,C.-K.J. (1991) *Proc. Natl Acad. Sci. USA*, **88**, 8676–8680.
34. Reddy,P.M.S. and Shen,C.-K.J. (1993) *Mol. Cell. Biol.*, **13**, 1093–1103.
35. Reddy,P.M.S., Stamatoyannopoulos,G., Papayannopoulou,T. and Shen,C.-K.J. (1994) *J. Biol. Chem.*, **269**, 8287–8295.
36. Caterina,J.J., Ciavatta,D.J., Donze,D., Behringer,R.R. and Townes,T.M. (1994) *Nucleic Acids Res.*, **22**, 1006–1011.
37. Lam,L. and Bresnick,E.H. (1996) *J. Biol. Chem.*, **271**, 32421–32429.
38. Elnitski,L., Miller,W. and Hardison,R. (1997) *J. Biol. Chem.*, **272**, 369–378.
39. Talbot,D., Philipsen,S., Fraser,P. and Grosveld,F. (1990) *EMBO J.*, **9**, 2169–2178.
40. Caterina,J.J., Ryan,T.M., Pawlik,K.M., Palmiter,R.D., Brinster,R.L., Behringer,R.R. and Townes,T.M. (1991) *Proc. Natl Acad. Sci. USA*, **88**, 1626–1630.
41. Hug,B.A., Moon,A.M. and Ley,T.J. (1992) *Nucleic Acids Res.*, **21**, 5771–5778.
42. Pruzina,S., Antoniou,M., Hurst,J., Grosveld,F. and Philipsen,S. (1994) *Biochim. Biophys. Acta*, **1219**, 351–360.
43. Ellis,J., Tan-Un,K.C., Harper,A., Michalovich,D., Yannoutsos,N., Philipsen,S. and Grosveld,F. (1996) *EMBO J.*, **15**, 562–568.
44. Jackson,J.D., Miller,W. and Hardison,R.C. (1996) *Nucleic Acids Res.*, **24**, 4327–4335.
45. Shelton,D.A., Stegman,L., Hardison,R., Miller,W., Slightom,J.L., Goodman,M. and Gumucio,D.L. (1997) *Blood*, **89**, 3457–3469.
46. Huisman,T.H.J., Carver,M.F.H. and Efremov,G.D. (1996) *A Syllabus of Human Hemoglobin Variants*. The Sickle Cell Anemia Foundation, Augusta, GA.
47. Myers,R.M., Tilly,K. and Maniatis,T. (1986) *Science*, **232**, 613–618.
48. Cowie,A. and Myers,R.M. (1988) *Mol. Cell. Biol.*, **8**, 3122–3128.
49. Macleod,K. and Plumb,M. (1991) *Mol. Cell. Biol.*, **11**, 4324–4332.
50. Stuve,L.L. and Myers,R.M. (1990) *Mol. Cell. Biol.*, **10**, 972–981.
51. Taxman,D.J. and Wojchowski,D.M. (1995) *J. Biol. Chem.*, **270**, 6619–6627.
52. deBoer,E., Antoniou,M., Mignotte,V., Wall,L. and Grosveld,F. (1988) *EMBO J.*, **7**, 4203–4212.
53. Miller,I.J. and Bieker,J.J. (1993) *Mol. Cell. Biol.*, **13**, 2776–2786.
54. Purnell,B.A., Emanuel,P.A. and Gilmour,D.S. (1994) *Genes Dev.*, **8**, 830–842.
55. Lawrence,J.G. and Ochman,H. (1998) *Proc. Natl Acad. Sci. USA*, **95**, 9413–9417.
56. Lee,N. (1980) In Miller,J.H. and Reznikoff,W.S. (eds), *The Operon*, 2nd Edn. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, NY, pp. 389–409.
57. Lee,N.L., Gielow,W.O. and Wallace,R.G. (1981) *Proc. Natl Acad. Sci. USA*, **78**, 752–756.
58. Miyada,C.G., Stoltzfus,L. and Wilcox,G. (1984) *Proc. Natl Acad. Sci. USA*, **81**, 4120–4124.
59. Huerta,A.M., Salgado,H., Thieffry,D. and Collado-Vides,J. (1998) *Nucleic Acids Res.*, **26**, 55–59.
60. Salgado,H., Santos,A., Garza-Ramos,U., van Helden,J., Diaz,E. and Collado-Vides,J. (1999) *Nucleic Acids Res.*, **27**, 59–60.
61. Schuler,G., Epstein,J., Ohkawa,H. and Kans,J. (1996) *Methods Enzymol.*, **266**, 141–162.
62. Makalowski,W. and Boguski,M.S. (1998) *Proc. Natl Acad. Sci. USA*, **95**, 9407–9412.