Nucleic Acids Research, Vol. 20, No. 5 ***1093–1099***

Critical comparison of consensus methods for molecular sequences

William H.E.Day and F.R.McMorris
Department of Computer Science, Memorial University of Newfoundland, St John's, NF A1C 5S7, Canada and Department of Mathematics, University of Louisville, Louisville, KY 40292, USA

Received November 18, 1991; Revised and Accepted February 2, 1992

ABSTRACT

Consensus methods are recognized as valuable tools for data analysis, especially when some sort of data aggregation is desired. Although consensus methods for sequences play a vital role in molecular biology, researchers pay little heed to the features and limitations of such methods, and so there are risks that criteria for constructing consensus sequences will be misused or misunderstood. To understand better the issues involved, we conducted a critical comparison of nine consensus methods for sequences, of which eight were used in papers appearing in this journal. We report the results of that comparison, and we make recommendations which we hope will assist researchers when they must select particular consensus methods for particular applications.

INTRODUCTION

Consensus methods via voting procedures have been powerful tools for political or social change, and today they are recognized as valuable tools for data analysis (1,2). In particular, theories of social choice have nurtured the development of consensus criteria such as the majority rule (3), the median rule (4,5), and the plurality rule (6–8). Using voting paradigms, researchers have extended theories of consensus so that consensus methods can be applied in areas of classification and systematics (9–12). As well, the consensus concept plays a vital role in molecular biology where consensus sequences are used, for example, to identify mRNA initiation and termination sequences (13–15), to analyse the secondary structure of RNA (16,17), to align multiple DNA sequences (18–20), and to find molecular patterns occurring imperfectly above a preset frequency (21,22). Recently Choo *et al.* (23) stressed the importance of the consensus concept in evolutionary investigations: 'At the most basic level, the present consensus sequence therefore represents an 'evolutionary consensus' which reveals the degree of tolerance for the conservation (or divergence) of nucleotide in the different positions within the alphoid monomers of different subfamilies. Information for such a consensus sequence is important for the understanding of the evolution of this DNA and its potential biological roles.'

Although consensus sequences play an important role in molecular biology, researchers pay little heed to the features and limitations of consensus sequence methods: when they calculate consensus sequences, they often do not state the underlying consensus criterion (24,25); when they state it, they usually do not consider its appropriateness to the application. In recent issues of *Nucleic Acids Research*, eight authors (14,15,23,26–30) stated and used distinct criteria for generating consensus sequences: no author discussed alternative criteria, and no author defended the use of the particular criterion employed. Thus there are risks that criteria for constructing consensus sequences will be misused or misunderstood. The dilemma is that, although many researchers use consensus sequences as experimental tools, few have studied theories or methodologies of consensus sequences (but see 20–22,31,32).

In these circumstances, we think it desirable to compare and evaluate existing consensus methods for sequences. We consider these methods to be tools to summarize the distributions of symbols (e.g., bases, amino acids) at the positions of an aligned set of molecular sequences. Typically the methods make three simplifying assumptions: analysis of molecular sequences is a multi–stage process in which sequence alignment precedes the identification of consensus sequences, an alignment of the molecular sequences has already been obtained, and aligned positions within molecular sequences can be treated independently (31). Thus the problem to find a consensus of k aligned molecular sequences, in which n aligned positions have been identified, can be viewed as a set of n simpler problems, each to find a consensus of k symbols at an aligned position. To model this simpler problem, we consider a consensus method to be a function $cm: U \rightarrow V$ which maps each element of its domain U to a suitable element of its codomain V. To specify U and V, let S be a finite set of symbols of interest. Although S might be the set of amino acids, in this paper we take it to be the set {A,C,G,T} of nucleic acid bases. Each element of U represents a possible k–tuple of bases appearing at a given aligned position in each of k molecules. For any positive integer k, let a *profile* of S be a k–tuple of symbols of S, and let S^k denote the set of profiles of S of length k. We will use S^k as the domain of the consensus methods we investigate. The codomain V typically is a set of ambiguity codes such as those proposed by the Nomenclature Committee of the International Union of Biochemistry (33). However, in order to emphasize the constituent bases, and their number, we will

represent each such code by the subset of its bases. Thus (with a minor abuse of set−theoretic notation) we define the set of subsets of S to be

$$\Pi(S) = \{\phi,A,C,G,T,AC,AG,AT,CG,CT,GT,ACG,ACT,AGT,CGT,ACGT\},$$

where AG denotes a purine base, CT denotes a pyrimidine base, and so on. We will use P(S) as the codomain of the consensus methods we investigate. When $cm(P) = \phi$, the consensus method maps the profile P to a symbol denoting the empty set: no ambiguity code is associated with P. By adopting this model of consensus, we exclude from consideration the problem of displaying patterns in aligned sequences (34−36).

The remainder of this paper is organized as follows. In the section on methods, we describe nine consensus methods (see Table 1) which are the subjects of our study. In the section on results, we consider six properties (appropriateness, basis, conformity, consistency, rationality, robustness) of consensus methods and relate the nine methods to them. In the concluding sections, we discuss general features of the consensus methods, and we make recommendations which we hope will assist researchers when they must select particular consensus methods for particular applications.

METHODS

We analyse nine consensus methods for sequences (Table 1). Eight of the methods have been used recently by authors of papers in *Nucleic Acids Research*, and so we have identified each method by the first two letters of the corresponding first author's surname (column 1 of Table 1). Each consensus method is a function $cm: S^k \rightarrow \Pi(S)$ which maps a profile P of length k to at most one ambiguity code in $\Pi(S)$. When defining the function's value for P, it is convenient to refer to the frequency of occurrence of each base in P, and furthermore to relabel the bases of P in such a way that frequencies and bases have a standard order. Since the consensus methods to be discussed depend on these frequencies, but not on physical or chemical properties of particular bases, these assumptions cause no loss of generality. Therefore, the absolute frequencies with which bases occur in P are specified by a vector $g = g(P) = (g_1,g_2,g_3,g_4)$, with $k = g_1+g_2+g_3+g_4$, where: (1) the symbols in the profile are relabeled so that $g_1 \geq g_2 \geq g_3 \geq g_4$, and (2) g_1 is the frequency of A, g_2 of C, g_3 of G, and g_4 of T. In addition, the relative frequencies $f_i = g_i/k$ are specified by a vector $f = f(P) = (f_1,f_2,f_3,f_4)$. For example, when $k = 9$ consider the profile (G,G,G,C,T,G,G,A,G). To satisfy requirements (1) and (2), interchange the labels A and G to obtain P = (A,A,A,C,T,A,A,G,A), so that $g(P)$ = (6,1,1,1) and $f(P)$ = (0.667, 0.111, 0.111, 0.111). Column 3 of Table 1 gives the criteria each consensus method uses to calculate the consensus results from frequencies. When P = (A,A,A,C,T,A,A,G,A), since $0.667 = f_1 > 0.5$ and $0.667 = f_1 > 2f_2 = 0.222$, column 2 shows that Cavener's consensus method (13) returns ambiguity code A, that is, $ca(P)$ = A. If column 2 does not specify an ambiguity code of given length (e.g., ACG for method *ca*), then that method never returns codes of that length.

Table 1. Consensus Methods. Except for method *pl*, an ID in column 1 is based on the first author's surname in column 4. Columns 2 and 3 define consensus methods using notation described in the text. In column 4, asterisks mark references having original definitions of methods; otherwise references describe recent applications of methods.

ID	Result	Criterion	References
ca	A:	$f_1 > 0.5$ and $f_1 > 2f_2$	Cavener (13*)
	AC:	($f_1 \leq 0.5$ or $f_1 \leq f_2$) and $f_1+f_2 > 0.75$	Cavener and Ray (14)
	ACGT:	($f_1 \leq 0.5$ *or* $f_1 \leq 2f_2$) and $f_1+f_2 \leq 0.75$	
ch	A:	$f_1 \geq 3f_2$	Choo *et al.*(23*)
	AC:	$f_1 < 3f_2$	
da	A:	$f_1 \geq 0.25$	Daniels and Deininger (26) Jurka and Milosavljevic (17)
gi	A:	$f_1 > 0.25$ and $f_4 \leq f_3 \leq f_2 \leq 0.25$	Gilson *et al.*(27*)
	AC:	$f_1 \geq f_2 > 0.25$ and $f_4 \leq f_3 \leq 0.25$	
	ACG:	$f_1 \geq f_2 \geq f_3 > 0.25$ and $f_4 \leq 0.25$	
	ϕ:	$f_1 = f_2 = f_3 = f_4 = 0.25$	
gr	A:	$f_1 \geq 0.875$	Grasser and Feix (28)
	ϕ:	$f_1 < 0.875$	Sakumi *et al.*(41)
sa	A:	$f_1 \geq 0.75$	Sayers and Eckstein (29)
	ϕ:	$f_1 < 0.75$	Kolodrubetz (38)
sh	A:	$f_1 \geq 0.4$ and ($f_2 < 0.3$ or $f_2 < 2f_3$)	Shapiro and Senapathy (30*)
	AC:	$f_1 \geq 0.4$ and ($f_2 \geq 0.3$ and $f_2 \geq 2f_3$)	
	ϕ:	$f_1 < 0.4$	
ya	A:	$f_1 > 0.5$	Yamauchi (15*)
	AC:	($f_1 \leq 0.5$) and ($f_1+f_2 > 0.75$)	
	ϕ:	($f_1 \leq 0.5$) and ($f_1+f_2 \leq 0.75$)	
pl	A:	$d(P,\beta_1) = \min_{1 \leq j \leq 4} d(P,\beta^j)$	Day and McMorris (31*)
	AC:	$d(P,\beta^2) = \min_{1 \leq j \leq 4} d(P,\beta^j)$	
	ACG:	$d(P,\beta^3) = \min_{1 \leq j \leq 4} d(P,\beta^j)$	
	ACGT:	$d(P,\beta^4) = \min_{1 \leq j \leq 4} d(P,\beta^j)$	

The ninth method (*pl*) is the plurality–rule consensus method (31,32). It is based on concepts of distance between profiles, of the median rule (4,5), of the plurality rule (6–8), and of balanced profiles. Given any two profiles P and Q of length k, the *distance* $d(P,Q)$ between them is a count of the number of positions where the profiles differ, so that distance is an integer between 0 (for identical profiles) and k (for completely different profiles). A profile P of length k is called *balanced* if its bases essentially appear equally often, i.e., if the positive numbers in $g(P)$ differ from each other by at most one. For example, (A,A,A,A,A,A,A), (A,A,A,A,C,C,C), (A,A,A,C,C,G,G) are balanced, but (A,A,A,A,A,C,C), (A,A,A,A,C,C,G), (A,A,A,C,C,C,G) are not. We use the notation β^j to denote a balanced profile in which j bases appear with positive frequencies. Thus for balanced profiles of length 7 we have $g(\beta^1)$ = (7,0,0,0), $g(\beta^2)$ = (4,3,0,0), $g(\beta^3)$ = (3,2,2,0), and $g(\beta^4)$ = (2,2,2,1). Method *pl*'s definition also depends on an explicit connection between ambiguity codes and balanced profiles: an ambiguity code of length j is said to be *associated with* any balanced profile that contains all and only the j bases of the ambiguity code. For example, ambiguity code ACG is associated with many balanced profiles including (A,A,C,C,C,G,G) and (A,A,C,C,G,G,G).

Informally, the plurality–rule method *pl* returns as consensus results all ambiguity codes associated with balanced profiles at a minimum distance from the given profile P (see (31) for *pl*'s formal definition). Thus *pl* differs from the other methods being studied because it may return more than one ambiguity code as a consensus result. In that case, since *pl* always returns at least one of the standard ambiguity codes in{A,AC,ACG,ACGT}, we judge *pl* to agree with another consensus method whenever the intersection of their results is nonempty. Table 1 gives the conditions under which *pl* returns the standard ambiguity codes. To illustrate the plurality–rule concepts, suppose P = (A,A,A,A,A,C,C,G,G) so that $g(P)$ = (5,2,2,0). Since P can be transformed into the balanced profile of nine As by changing four bases in P, it follows that $d(P,\beta^1) = 4$. Since (by changing two bases) P can be transformed into either a balanced profile with five As and four Cs or one with five As and four Gs, we have $d(P,\beta^2) = 2$. The reader can verify that $d(P,\beta^3) = 2$ so that $pl(P)$ = {AC,AG,ACG} (as at line 10 of Table 4), and thus we judge *pl* to agree with any consensus method returning AC or ACG.

The profiles described in Table 2 demonstrate that the nine consensus methods are distinct with respect to ambiguity codes of length one. For example, if a profile P has frequencies $g(P)$ = (5,1,0,0), then $gr(P) = \phi$ and $sa(P)$ = A so that $gr \neq sa$. In the same way, the profiles of Table 2 demonstrate that the six consensus methods returning ambiguity codes of length two are distinct with respect to ambiguity codes of that length. For example, if a profile P has frequencies $g(P)$ = (7,3,0,0), then $gi(P)$ = AC and $ca(P)$ = A so that $gi \neq ca$.

RESULTS

Table 3 and Figure 1 show how the consensus methods of Table 1 relate to the following properties.

Basis

Call a consensus method *frequency–based* if it is defined solely in terms of the relative frequencies $f = (f_1,f_2,f_3,f_4)$ of a given profile. Column 3 of Table 1 demonstrates that eight of the consensus methods are frequency–based. By contrast, method *pl* is defined in terms of distances, and an associated median

Table 2. Examples for Comparing Consensus Methods. Each row represents a profile P with frequencies $g(P)$ and $f(P)$ as shown in columns 1 and 2. The remaining columns give the consensus results obtained by the consensus methods.

g_1	g_2	g_3	g_4	f_1	f_2	f_3	f_4	*gr*	*sa*	*pl*	*ca*	*ya*	*ch*	*sh*	*gi*	*da*
8	1			0.875	0.125			A	A	A	A	A	A	A	A	A
5	1			0.833	0.167			ϕ	A	A	A	A	A	A	A	A
7	1	1	1	0.7	0.1	0.1	0.1	ϕ	ϕ	A	A	A	A	A	A	A
7	2	1	1	0.636	0.182	0.091	0.091	ϕ	ϕ	AC	A	A	A	A	A	A
7	3			0.7	0.3			ϕ	ϕ	AC	A	A	AC	AC	AC	A
4	2	1		0.571	0.286	0.143		ϕ	ϕ	AC ACG	ACG	AC	A	AC	A	AC
20	11	9		0.5	0.275	0.225		ϕ	ϕ	ACG	AC	AC	AC	A	AC	A
16	12	6		0.471	0.353	0.176		ϕ	ϕ	ACG	AC	AC	AC	AC	AC	A
2	1	1		0.500	0.250	0.250		ϕ	ϕ	ACG	ACGT	ϕ	AC	A	A	A
3	2	2	1	0.275	0.250	0.250	0.125	ϕ	ϕ	ACG ACGT	ACGT	ϕ	AC	ϕ	A	A
3	1	1	1	0.500	0.167	0.167	0.167	ϕ	ϕ	ACGT	ACGT	ϕ	A	A	A	A

Table 3. Properties of Consensus Methods. The section on results defines these properties.

Property	*ca*	*ch*	*da*	*gi*	*gr*	*sa*	*sh*	*ya*	*pl*
Basis (frequency or median):	freq.	freq.	freq.	freq.	freq.	freq.	freq.	freq.	med.
Rational?	yes	yes	yes	yes	yes	yes	yes	yes	yes
Robust at lengths:	1,2,4	1	–	1–3	1	1	1,2	1,2	1–4
Appropriate at lengths:	1,2	1	1	1–3	1	1	1,2	1,2	1–4
Consistent?	yes	yes	yes	yes	yes	yes	yes	yes	no

concept (4,5,9), and so we call it *median−based*. Although the distinction between these concepts seems obvious, in fact it is subtle since method *pl* can be defined solely in terms of the absolute frequencies $g = (g_1,g_2,g_3,g_4)$ of a profile. Put another way, the difference between median−based and frequency−based methods concerns whether their specifications require a profile's length, k, as well as its relative frequencies. In the subsection about consistency (below), we will show that the consensus results returned by *pl* depend on k.

Rationality

Call a consensus method *rational* if, when the method returns an ambiguity code of length m for a profile with frequencies $f_1 \geq f_2 \geq f_3 \geq f_4$, the m bases of the ambiguity code have the frequencies $f_1,\ldots f_m$. Thus, for a profile with frequencies $f_1 > f_2 \geq f_3 \geq f_4$, a rational consensus method must return ambiguity codes from {A,AC,AG,AT,ACG,ACT,AGT,ACGT} but not from{C,G,T,CG,CT,GT,CGT}. It's easy to verify that the consensus methods of Table 1 are rational; however, they need not satisfy a property of robustness.

Robustness

Call a consensus method *robust* at length m if, when the method returns an ambiguity code of length m for a profile with frequencies $f_1 \geq f_2 \geq f_3 \geq f_4$, it returns <u>all</u> ambiguity codes whose bases have the frequencies $f_1,\ldots f_m$. The median−based method *pl*, which has the ability to return more than one ambiguity code, is defined so as to be robust at all lengths. However, two frequency−based methods in Table 1 violate the robustness criterion. Method *da* is not robust at length one since, for example, if a profile has $f = (x,x,1-2x,0)$ and $1/2 \geq x \geq 1/3$, then *da* returns only code A even though A and C both have frequency x. Method *ch* is not robust at length two since, for example, if a profile has $f = (1-2x,x,x,0)$ and $1/3 \geq x > 1/5$, then *ch* returns only code AC even though AC and AG both have frequencies $1-2x$ and x. Table 3 shows the lengths at which the consensus methods of Table 1 are robust; details of the analyses are in Appendix 1 of (37).

Appropriateness

Researchers use consensus methods for diverse purposes. Some wish to identify positions at which variation in the distribution of bases can be summarized by ambiguity codes of length one. Others are content to describe positions by ambiguity codes of lengths one or two, and so on. For any integer j, with $1 \leq j \leq 4$, call a consensus method *appropriate* at length j if it has a reasonable criterion for returning ambiguity codes of length j. Table 3 specifies the values of j at which the consensus methods of Table 1 seem appropriate. This classification is based on analyses of the criteria in column 3 of Table 1. For example, not only does method *ca* return ACGT when it is unable to return meaningful codes of lengths one or two, it also returns no ambiguity codes of length three; thus *ca*'s use should be restricted to cases when codes of lengths one or two are appropriate. Not only does method *ch* return AC when it is unable to return meaningful codes of length one, it also returns no ambiguity codes of lengths three or four; thus *ch*'s use should be restricted to cases when codes of length one are appropriate. The other methods seem appropriate at those lengths for which they return ambiguity codes.

Consistency

For profiles P and Q, let $P+Q$ denote the concatenation of P and Q; then any consensus method *cm* having the codomain P(S) is called *consistent* if $cm(P+Q) = cm(P)$ whenever $cm(P) = cm(Q)$. When a method *cm* returns sets of ambiguity codes, it is called *consistent* if $cm(P+Q) = cm(P) \cap cm(Q)$ whenever $cm(P) \cap cm(Q) \neq \phi$ (31). It's easy to see that the frequency−based methods of Table 1 are consistent, and so their consensus results are invariant under changes in the scale of the profile being analysed. If $ca(P)$ = A when P = (A,A,A,A,C), one might take comfort in knowing that *ca* returns the same result when the profile is $P+P$ = (A,A,A,A,C,A,A,A,A,C). Informally, any insights obtained by analysing the behaviour of a consistent consensus method for profiles of small size also hold for comparable profiles of larger size. Method *pl* is not always consistent. For the previous example, $pl(P)$ = {A,AC} since one change in P = (A,A,A,A,C) yields the balanced profiles (A,A,A,A,A) or (A,A,A,C,C); yet $pl(P+P)$ = {A} since two changes in $P+P$ yield the balanced profile (A,A,A,A,A,A,A,A,A,A), while three changes yield (A,A,A,C,C,A,A,C,C,C). Informally, what has happened is that *pl* used the longer profile's absolute frequencies to differentiate between optimal solutions for the original profile.

Conformity

Since researchers seem preoccupied with using ambiguity codes of length one, we analysed the consistency with which consensus methods return ambiguity codes of each possible length. Let m

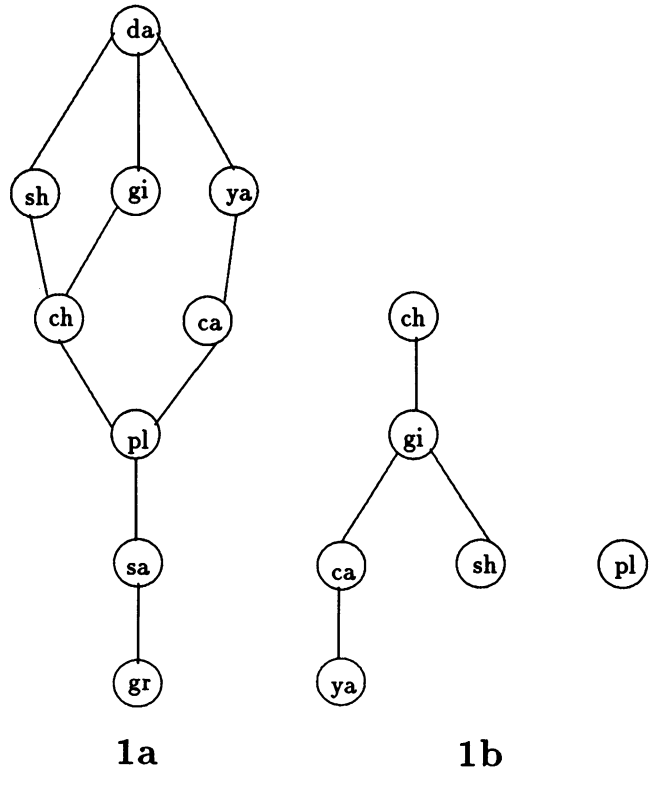

Figure 1. Diagrams showing how consensus methods conform to each other at length one (Figure 1*a*) or length two (Figure 1*b*). If two methods are connected by an edge, the upper method covers the lower one with respect to relation $<_1$ (Figure 1*a*) or $<_2$ (Figure 1*b*).

be the ambiguity−code length of interest. For consensus methods *cm* and *cm'*, define the relation $<_m$ such that $cm <_m cm'$ if for all profiles P, $cm(P) = s_1 \ldots s_m$ only if $cm'(P) = s_1 \ldots s_m$, and for some profile P, $cm(P) \neq s_1 \ldots s_m$ and $cm'(P) = s_1 \ldots s_m$. When $cm <_m cm'$ we say that *cm* conforms to *cm'* at length m. Clearly $<_m$ is irreflexive and transitive, and so it is a strict partial order. If $cm <_m cm'$, then *cm'* extends the domain of profiles with which *cm* associated ambiguity codes of length m: *cm'* gives the same results as *cm* when *cm* returns ambiguity codes of length m, but *cm'* returns ambiguity codes of length m for some profiles where *cm* returned no ambiguity codes or ambiguity codes of lengths not equal to m.

Figure 1 depicts the main results concerning conformity of consensus methods at lengths one and two; details of the analyses are in Appendix 2 of (37). Figure 1*a* shows the ordering of the consensus methods with respect to $<_1$. Since *gr* conforms to all other methods in the study, it has the smallest domain of profiles with which are associated ambiguity codes of length one, while *da* has the largest domain. Notice in Figure 1*a* the position of method *pl*. Since it often returns ambiguity codes of lengths greater than one, *pl* has a relatively small domain of profiles with which are associated ambiguity codes of length one; thus it appears toward the bottom of the diagram. Figure 1*a* suggests, and examples in Table 2 confirm, that seven pairs of consensus methods are incomparable with respect to $<_1$. For example, consider *ca* and *ch*. Since any profile P having frequencies $g(P) = (7,3,0,0)$ is assigned ambiguity code A by *ca*, but AC by *ch*, it is false that $ca <_1 ch$. Since any profile P having frequencies $g(P) = (3,1,1,1)$ is assigned ambiguity code A by *ch*, but ACGT by *ca*, it is false that $ch <_1 ca$. Thus *ca* and *ch* are incomparable with respect to $<_1$.

Figure 1*b* shows the ordering of six consensus methods with respect to $<_2$. Figure 1*b* suggests, and examples in Table 2 confirm, that seven pairs of these methods are incomparable with respect to $<_2$. That Figure 1*b* has less structure than Figure 1*a* is confirmed by the occurrences of two maximal elements (*ch* and *pl*) and three minimal elements (*sh*, *ya*, *pl*). The peculiar (indeed, inappropriate) way in which *ch* is defined for ambiguity codes of length two makes it the maximum of the frequency−based methods. An absence of conformity at length two between method *pl* and any of the frequency−based methods suggests the uniqueness of its median−based definition.

The lack of conformity between plurality rule and the frequency−based methods continues to occur at lengths three and four. Only *gi* and *pl* return ambiguity codes of length three, and neither conforms to the other at this length. For example, since any profile P having frequencies $g(P) = (2,2,2,1)$ is assigned code ACG by *gi*, but ACGT by *pl*, it is false that $gi <_3 pl$; since any profile having frequencies (4,4,3,1) is assigned code ACG by *pl*, but AC by *gi*, it is false that $pl <_3 gi$. Only *ca* and *pl* return ambiguity codes of length four, and neither conforms to the other at this length. For example, since any profile P having frequencies $g(P) = (4,3,3,0)$ is assigned code ACGT by *ca*, but ACG by *pl*, it is false that $ca <_4 pl$; since any profile having frequencies (7,2,2,2) is assigned code ACGT by *pl*, but A by *ca*, it is false that $pl <_4 ca$.

Table 4. Consensus Methods Applied to Actual Data. Each row represents a set of profiles which yield the same pattern of consensus results. Column 1 is a line number. Column 2 gives the number of profiles in the set. Columns 3–10 describe two representative profiles in the set. The remaining columns give for these profiles the consensus results obtained by the consensus methods. Lines 1–5 describe 74 profiles in Figure 4 of Grasser and Feix (28). Lines 6–11 describe 100 profiles in Figure 1 of Crowther *et al.* (24). Lines 12–41 describe 1140 profiles in Figure 2 of Irwin *et al.* (40).

Line	Total	g_1	g_2	g_3	g_4	g_1	g_2	g_3	g_4	*ca*	*ch*	*da*	*gi*	*gr*	*sa*	*sh*	*ya*	*pl*
1	57	8	0	0	0	7	1	0	0	A	A	A	A	A	A	A	A	A
2	10	6	1	1	0					A	A	A	A	Ø	A	A	A	A
3	2	6	2	0	0					A	A	A	A	Ø	A	A	A	A,AC
4	2	5	1	1	1					A	A	A	A	Ø	Ø	A	A	A,AC,AG,AT,ACG,...
5	3	5	2	1	0					A	AC	A	A	Ø	Ø	A	A	AC,ACG
6	86	9	0	0	0	8	1	0	0	A	A	A	A	A	A	A	A	A
7	3	7	1	1	0					A	A	A	A	Ø	A	A	A	A
8	6	6	2	1	0					A	A	A	A	Ø	Ø	A	A	AC
9	1	6	1	1	1					A	A	A	A	Ø	Ø	A	A	A,AC,AG,AT,ACGT
10	1	5	2	2	0					A	AC	A	A	Ø	Ø	A	A	AC,AG,ACG
11	3	6	3	0	0	5	3	1	0	AC	AC	A	AC	Ø	Ø	AC	A	AC
12	701	20	0	0	0	18	1	1	0	A	A	A	A	A	A	A	A	A
13	136	17	3	0	0	15	2	2	1	A	A	A	A	Ø	A	A	A	A
14	25	15	5	0	0					A	A	A	A	Ø	A	A	A	A,AC
15	12	14	3	3	0	14	3	2	1	A	A	A	A	Ø	Ø	A	A	A
16	12	14	4	2	0	13	3	2	2	A	A	A	A	Ø	Ø	A	A	A,AC
17	2	13	3	3	1					A	A	A	A	Ø	Ø	A	A	A,AC,AG,ACG
18	3	13	4	2	1	12	4	2	2	A	A	A	A	Ø	Ø	A	A	AC
19	7	13	4	3	0	12	4	3	1	A	A	A	A	Ø	Ø	A	A	AC,ACG
20	2	12	4	4	0					A	A	A	A	Ø	Ø	A	A	ACG
21	22	14	5	1	0	11	4	3	2	A	AC	A	A	Ø	Ø	A	A	AC
22	10	12	5	3	0	11	5	3	1	A	AC	A	A	Ø	Ø	A	A	AC,ACG
23	7	11	5	4	0	11	4	4	1	A	AC	A	A	Ø	Ø	A	A	ACG
24	25	14	6	0	0	13	6	1	0	A	AC	A	AC	Ø	Ø	AC	A	AC
25	2	9	7	4	0					AC	AC	A	AC	Ø	Ø	A	AC	ACG
26	91	13	7	0	0	11	6	2	1	AC	AC	A	AC	Ø	Ø	AC	A	AC
27	4	11	6	3	0					AC	AC	A	AC	Ø	Ø	AC	A	AC,ACG
28	26	10	10	0	0	9	8	2	1	AC	AC	A	AC	Ø	Ø	AC	AC	AC
29	14	10	7	3	0	9	8	3	0	AC	AC	A	AC	Ø	Ø	AC	AC	AC,ACG
30	2	9	7	3	1					AC	AC	A	AC	Ø	Ø	AC	AC	ACG
31	2	10	5	3	2					ACGT	AC	A	A	Ø	Ø	A	Ø	AC,ACG,ACGT
32	4	10	5	5	0	10	5	4	1	ACGT	AC	A	A	Ø	Ø	A	Ø	ACG
33	3	9	5	4	2					ACGT	AC	A	A	Ø	Ø	A	Ø	ACG,ACGT
34	5	10	4	3	3	8	5	4	3	ACGT	AC	A	A	Ø	Ø	A	Ø	ACGT
35	1	7	5	4	4					ACGT	AC	A	A	Ø	Ø	Ø	Ø	ACGT
36	11	9	6	5	0	8	6	5	1	ACGT	AC	A	AC	Ø	Ø	A	Ø	ACG
37	4	9	6	3	2	8	7	3	2	ACGT	AC	A	AC	Ø	Ø	AC	Ø	ACG
38	1	7	6	5	2					ACGT	AC	A	AC	Ø	Ø	Ø	Ø	ACG
39	1	7	6	4	3					ACGT	AC	A	AC	Ø	Ø	Ø	Ø	ACG,ACGT
40	3	8	6	6	0					ACGT	AC	A	ACG	Ø	Ø	A	Ø	ACG
41	2	7	7	6	0	7	6	6	1	ACGT	AC	A	ACG	Ø	Ø	Ø	Ø	ACG

DISCUSSION

The frequency criteria in column 3 of Table 1 contribute in two ways to the identification of ambiguity codes. Some (with form $f_i \geq c$ or $f_i > c$, where c is a constant) establish a threshold below which a code cannot be returned. Others (e.g., $f_1 > 2f_2$, $f_1 \geq 3f_2$, $f_1 > 0.4$ and $f_2 < 0.3$) connect a code's return to the existence of a gap between consecutive frequencies. Methods *da*, *gr*, *sa*, and *ya* are based on threshold criteria; *ch*, *gi*, and *sh*, on gap criteria; while *ca* employs criteria of both types. Threshold and gap criteria need not be independent since, for example, any threshold of the form $f_1 \geq x > 0.5$ establishes as well a gap of at least $2f_1 - 1$ between f_1 and f_2.

Although simplicity of concept makes consensus methods based on thresholds popular (15,28,29,38,39), their conformity with other methods is difficult to predict. For any c with $1.0 \geq c > 0.5$, let th_c denote a *threshold method* for which $th_c(P)$ = A when $f_1 \geq c$, and $th_c(P) = \phi$ otherwise. With this terminology we have $gr = th_{0.875}$, $sa = th_{0.75}$, and *ya* is effectively $th_{0.5+\epsilon}$ for very small positive ϵ, but see (39) for uses of five other threshold methods. When $1.0 \geq c \geq 0.75$, the results in Appendix 2 of (37) show that th_c conforms to *pl*, *ch*, and *ca* at length one. When $0.75 > c > 2/3$, however, similar arguments show that th_c conforms to *ca* at length one but is incomparable with *pl* and *ch*. When $2/3 \geq c > 0.625$, th_c is incomparable at length one with *pl*, *ch*, and *ca*. Finally, when $0.625 \geq c > 0.5$, *pl* conforms to th_c at length one although th_c is incomparable with *ch* and *ca*.

Although frequencies are at the heart of method *pl*'s definition, the ambiguity codes it returns are best understood as solutions of an optimization problem. If P is a profile and X is an ambiguity code returned by *pl* for P, then X has an associated balanced profile Q which is closer to P than is any other balanced profile. The measure of closeness (in fact, of distance) counts the number of bases that must be changed to transform P into Q. No ambiguity code other than X provides a better (i.e., closer) description of P. Since ambiguity codes other than X may provide equally good (i.e., equally close) descriptions of P, *pl* returns them all as the consensus result for P.

When a consensus method returns more than one consensus result, the display of the several results is awkward. For example, when S = {A,C,G,T} method *pl* can return up to eight ambiguity codes as equally good descriptions of a profile (32). Since such results identify positions where the distribution of bases has many equally valid descriptions, researchers should be cautious when selecting just a single ambiguity code to summarize that distribution.

Since ambiguity codes returned by method *pl* are solutions of an optimization problem, they share a single defining criterion even when their lengths are different. By contrast, the frequency−based methods of Table 1 apply different defining criteria to return codes of different lengths.

For all the consensus methods of Table 1, the problem of calculating the consensus result for profiles of length k can be solved efficiently by algorithms requiring order at most k time and space. As far as algorithm design and complexity are concerned, all these methods could be used to analyse amino acid, as well as DNA, sequences. Of course, users must verify that a method's properties (as summarized, for example, in Table 3) make its use appropriate in their applications.

Table 4 summarizes applications of the consensus methods of Table 1 to actual data: lines 1−5 have to do with analysing the amino acid sequences of eight HMG−box regions (28); lines 6−11 concern DNA sequences of L1Hs elements from nine λ clones (24); lines 12−41 pertain to the DNA sequences of 20 mammalian cytochrome *b* genes (40). The percentage of profiles on which all nine consensus methods agree varies from 89% when k=9 (line 6) to 61% when k=20 (line 12). When compared with those of frequency−based methods, *pl*'s results often seem conservative since *pl* returns ambiguity codes of lengths greater than one for 24% of the profiles. In line 8, for example, six frequency−based methods return A as a consensus while *pl* returns AC (since only two changes transform (6,2,1,0) to (5,4,0,0), while three changes transform it to (9,0,0,0)). Method *pl* exhibits conservatism of another type where (for 7% of the profiles) it returns more than one consensus result. In line 3, for example, seven frequency−based methods return A as the consensus while *pl* returns both A and AC (since two changes transform (6,2,0,0) to (8,0,0,0) or (4,4,0,0)). Line 4 illustrates an extreme case in which *pl* returns the eight ambiguity codes A,AC,AG,AT,ACG,ACT,AGT,ACGT when S ={A,C,G,T}. In all these cases the user may apply criteria based on philosophical or biological considerations to reduce the size of *pl*'s consensus set, but such considerations have nothing to do with the *pl* method and their appropriateness may depend on the application. Notice that in each application summarized in Table 4, *pl* returns ambiguity codes of each of the four possible lengths. This fact highlights the inability of the frequency−based methods of Table 1 to return consensus results of particular lengths. For example, seven of those methods cannot return ambiguity codes of length three (perhaps because such results are expected to be rare or of little biological interest). By contrast, for 7% of the profiles *pl* returns ambiguity codes of length three because they are best (closest) descriptions of the given profiles.

RECOMMENDATIONS

Of the consensus methods in Table 1, only *pl* returns ambiguity codes of the four possible lengths, and so only *pl* would serve adequately as a completely general method to summarize the distribution of bases at a position in a set of aligned molecular sequences.

Although method *gi* is robust and appropriate at lengths one through three, it permits great variation among profiles to which it assigns the same consensus result. As examples, *gi* returns code A for profiles having absolute frequencies g = (100,0,0,0) and g = (26,25,25,24), AC for profiles with g = (74,26,0,0) and g = (26,26,24,24), and ACG for profiles with g = (48,26,26,0) and g = (26,26,26,22). Most researchers will find it unsatisfactory that *gi* does not return ACGT for profiles with g = (26,25,25,24), g = (26,26,24,24), or g = (26,26,26,22). For these reasons, we do not recommend using method *gi*.

The voting strategy on which method *da* is based is so simplistic that *da* is not robust and is not sensitive to the presence of gaps. Consequently, *da* returns code A for profiles with no gaps as when g = (25,25,25,25), with small gaps as when g = (26,25,25,24), and with large gaps as when g = (100,0,0,0). For these reasons,we do not recommend using method *da*.

The remaining seven consensus methods of Table 1 are at least appropriate and robust at length one. If researchers wish to use them to obtain codes of length one, Figure 1*a* exhibits two chains of methods which are ordered by conformity at length one: $gr <_1 sa <_1 pl <_1 ch <_1 sh$, and $gr <_1 sa <_1 pl <_1 ca <_1 ya$. Within either chain, methods to the left apply strict criteria to

obtain codes of length one, while those to the right apply lenient criteria. Of the seven methods, four (i.e., *ca*, *pl*, *sh*, *ya*) are appropriate and robust at length two. If researchers wish to use them to obtain codes of length two, Figure 1*b* shows that they cluster into three groups with *ya* and *ca* in one group and with *pl* and *sh* in separate groups. Although *ca* and *ya* are clearly related by their definitions, the sense of their conformity reverses between length one (Figure 1*a*) and length two (Figure 1*b*). Of the seven methods, only *pl* is appropriate and robust at lengths three and four.

ACKNOWLEDGEMENTS

We acknowledge with pleasure the correspondence or criticisms of S. Carr, D. R. Cavener, G. R. Estabrook, M. B. Shapiro, D. M. Woodcock, K. Yamauchi, and two anonymous referees. This work was supported in part by the Canadian Institute for Advanced Research, the Natural Sciences and Engineering Research Council of Canada under grant A-4142, and by the US Office of Naval Research under grant N00014-89-J-1643. The first author is an Associate in the Program in Evolutionary Biology of the Canadian Institute for Advanced Research.

REFERENCES

1. Day, W. H. E. (1986) *J. Classification*, **3**, 183–185.
2. Day, W. H. E. (1988) In Bock, H. H. (ed.), *Classification and Related Methods of Data Analysis*. North–Holland, Amsterdam, pp. 317–324.
3. May, K. O. (1952) *Econometrica*, **20**, 680–684.
4. Kemeny, J. G. (1959) *Daedalus (Boston)*, **88**, 577–591.
5. Kemeny, J. G., and Snell, J. L. (1962) *Mathematical Models in the Social Sciences*. Ginn, New York, pp. 9–23.
6. Richelson, J. (1978) *J. Econom. Theory*, **19**, 548–550.
7. Roberts, F. S. (1991) *Math. Soc. Sci.*, **21**, 101–127.
8. Roberts, F. S. (1991) *Math. Soc. Sci.*, to appear.
9. Barthélemy, J.–P., and Monjardet, B. (1981) *Math. Soc. Sci.*, **1**, 235–267.
10. Margush, T., and McMorris, F. R. (1981) *Bull. Math. Biol.*, **43**, 239–244.
11. McMorris, F. R., and Neumann, D. (1983) *Math. Soc. Sci.*, **4**, 131–136.
12. Barthélemy, J.–P., and McMorris, F. R. (1986) *J. Classification*, **3**, 329–334.
13. Cavener, D. R. (1987) *Nucleic Acids Res.*, **15**, 1353–1361.
14. Cavener, D. R., and Ray, S. C. (1991) *Nucleic Acids Res.*, **19**, 3185–3192.
15. Yamauchi, K. (1991) *Nucleic Acids Res.*, **19**, 2715–2720.
16. Piller, K. J., Baerson, S. R., Polans, N. O., and Kaufman, L. S. (1990) *Nucleic Acids Res.*, **18**, 3135–3145.
17. Jurka, J., and Milosavljevic, A. (1991) *J. Mol. Evol.*, **32**, 105–121.
18. Bains, W. (1986) *Nucleic Acids Res.*, **14**, 159–177.
19. Bains, W. (1989) *CABIOS*, **5**, 51–52.
20. Waterman, M. S. (1986) *Nucleic Acids Res.*, **14**, 9095–9102.
21. Waterman, M. S., Arratia, R., and Galas, D. J. (1984) *Bull. Math. Biol.*, **46**, 515–527.
22. Waterman, M. S. (1989) In Waterman, M. S. (ed.), *Mathematical Methods for DNA Sequences*. CRC Press, Boca Raton, pp. 93–115.
23. Choo, K. H., Vissel, B., Nagy, A., Earle, E., and Kalitsis, P. (1991) *Nucleic Acids Res.*, **19**, 1179–1182.
24. Crowther, P. J., Doherty, J. P., Linsenmeyer, M. E., Williamson, M. R., and Woodcock, D. M. (1991) *Nucleic Acids Res.*, **19**, 2395–2401.
25. Krinke, L., and Wulff, D. L. (1990) *Nucleic Acids Res.*, **18**, 4809–4815.
26. Daniels, G. R., and Deininger, P. L. (1991) *Nucleic Acids Res.*, **19**, 1649–1656.
27. Gilson, E., Saurin, W., Perrin, D., Bachellier, S., and Hofnung, M. (1991) *Nucleic Acids Res.*, **19**, 1375–1383.
28. Grasser, K. D., and Feix, G. (1991) *Nucleic Acids Res.*, **19**, 2573–2577.
29. Sayers, J. R., and Eckstein, F. (1991) *Nucleic Acids Res.*, **19**, 4127–4132.
30. Shapiro, M. B., and Senapathy, P. (1987) *Nucleic Acids Res.*, **15**, 7155–7174.
31. Day, W. H. E., and McMorris, F. R. (1991) *Bull. Math. Biol.*, to appear.
32. Day, W. H. E., and McMorris, F. R. (1991) *Math. Biosci.*, submitted.
33. Nomenclature Committee Of The International Union Of Biochemistry (NC–IUB) (1985) *Eur. J. Biochem.*, **150**, 1–5. Also (1986) *J. Biol. Chem.*, **261**, 13–17.
34. Reznikoff, W. S., and McClure, W. R. (1986) In Reznikoff, W. S., and Gold, L. (eds.), *Maximizing Gene Expression*. Butterworths, Boston, 1–33.
35. Card, C. O., Wilson, G. G., Weule, K., Hasapes, J., Kiss, A., and Roberts, R. J. (1990) *Nucleic Acids Res.*, **18**, 1377–1383.
36. Schneider, T. D., and Stephens, R. M. (1990) *Nucleic Acids Res.*, **18**, 6097–6100.
37. Day, W. H. E., and McMorris, F. R. (1991) Tech. Rept. 9114, Dept. Computer Sci., Memorial Univ. Newfoundland.
38. Kolodrubetz, D. (1990) *Nucleic Acids Res.*, **18**, 5565.
39. McGeoch, D. J. (1990) *Nucleic Acids Res.*, **18**, 4105–4110.
40. Irwin, D. M., Kocher, T. D., and Wilson, A. C. (1991) *J. Mol. Evol.*, **32**, 128–144.
41. Sakumi, K., Shiraishi, A., Hayakawa, H., and Sekiguchi, M. (1991) *Nucleic Acids Res.*, **19**, 5597–5601.