Computers Chem. Vol. 17, No. 2, pp. 149–163, 1993
Printed in Great Britain

0097-8485/93 $6.00 + 0.00
Pergamon Press Ltd

STATISTICS OF LOCAL COMPLEXITY IN AMINO ACID SEQUENCES AND SEQUENCE DATABASES*

JOHN C. WOOTTON† and SCOTT FEDERHEN
National Center for Biotechnology Information, National Library of Medicine, Building 38A, 8N805, National Institutes of Health, Bethesda, MD 20894, U.S.A.

(Received 7 October 1992; in revised form 9 March 1993)

Abstract—Protein sequences contain surprisingly many local regions of low compositional complexity. These include different types of residue clusters, some of which contain homopolymers, short period repeats or aperiodic mosaics of a few residue types. Several different formal definitions of local complexity and probability are presented here and are compared for their utility in algorithms for localization of such regions in amino acid sequences and sequence databases. The definitions are:—(1) those derived from enumeration *a priori* by a treatment analogous to statistical mechanics, (2) a log likelihood definition of complexity analogous to informational entropy, (3) multinomial probabilities of observed compositions, (4) an approximation resembling the χ^2 statistic and (5) a modification of the coefficient of divergence. These measures, together with a method based on similarity scores of self-aligned sequences at different offsets, are shown to be broadly similar for first-pass, approximate localization of low-complexity regions in protein sequences, but they give significantly different results when applied in optimal segmentation algorithms. These comparisons underpin the choice of robust optimization heuristics in an algorithm, SEG, designed to segment amino acid sequences fully automatically into subsequences of contrasting complexity. After the abundant low-complexity segments have been partitioned from the Swissprot database, the remaining high-complexity sequence set is adequately approximated by a first-order random model.

1. INTRODUCTION

Natural protein sequences are very different from random strings of 20 amino acids. In recent years an increasing proportion of polypeptide sequences translated from cloned genes or cDNAs have revealed many highly non-random regions. These include clusters of glycine, proline, alanine, glutamine, serine, histidine, glutamate, aspartate, arginine, lysine, asparagine or threonine residues, commonly in homopolymeric tracts or in mosaic sequence arrangements, some of which contain regular or irregular short-period tandem repeats. A recent study that analyzed amino acid sequence databases globally (Wootton & Federhen, 1993) found that approx. 40% of sequence entries contain at least one such cluster and approx. 15% of the residues in the database occur in segments of improbably low compositional complexity. These clusters are very poorly understood at the molecular level and they were not anticipated from classical structural studies of globular proteins. Their compositional biases are very much greater than the relatively well-understood constraints observed in secondary structure elements and supersecondary structural motifs that are familiar from crystal and NMR structures.

The work reported here provides a foundation for algorithms that analyze the local complexity of protein sequences and make automated segmentation of sequences on the basis of defined complexity characteristics. Such algorithms are important because description and classification of the low-complexity sequences of proteins can provide a focus for further research. Why, for example, do such residue clusters occur so commonly? This turns out to be a profound question that bears on many aspects of protein structure and interactions, biological function, genome structure and mutational flux. As a typical illustration, a recently-determined human sequence, RING3 (Beck *et al.*, 1992), which contains several low-complexity segments of unknown function and shows an interesting distribution of homology to the *Drosophila* Fsh protein (Haynes *et al.*, 1989, 1992), is shown in Fig. 1. This example also illustrates the application of one automated segmentation algorithm that is based on some of the formalisms developed in this report.

The great majority of low-complexity clusters are relatively short subsequences, in the length range of 15–50 residues, that do not resemble the functionally well-understood, abundant structural proteins such as keratins, collagens and elastins. Although little is known about their molecular structures, dynamics and interactions, several distinct classes of low-complexity clusters are strikingly abundant

* The preliminary version of this work was presented during the *Second International Workshop on Open Problems of Computational Molecular Biology*, Telluride Summer Research Center, Telluride, Colo., 19 July–2 August 1992.

† Author for correspondence.

Low-complexity segments		High-complexity segments
	1-4	MASV
palqltpanppppevsnpkkp	5-25	
	26-428	GRVTNQLQYLHKVVMKALWKHQFAWPFRQPVDAVKLGLPDYHKIIKQPMDMGTIKRRLEN NYYWAASECMQDFNTMFTNCYIYNKPTDDIVLMAQTLEKIFLQKVASMPQEEQELVVTIP KNSHKKGAKLAALQGSVTSAHQVPAVSSVSHTALYTPPPEIPTTVLNIPHPSVISSPLLK SLHSAGPPLLAVTAAPPAQPLAKKKGVKRKADTTTPTPTAILAPGSPASPPGSLEPKAAR LPPMRRESGRPIKPPRKDLPDSQQQHQSSKKGKLSEQLKHCNGILKELLSKKHAAYAWPF YKPVDASALGLHDYHDIIKHPMDLSTVKRKMENRDYRDAQEFAADVRLMFSNCYKYNPPD HDVVAMARKLQDVFEFRYAKMPDEPLEPGPLPVSTAMPPGLAK
sssessseessssessseeeeeedeedeeeeesessdseee	429-468	
	469-496	RAHRLAELQEQLRAVHEQLAALSQGPIS
kpkrkrekkekkkkrkaek	497-515	
	516-536	HRGRAGADEDDKGPRAPRPPQ
pkkskkasgsgggsaalgpsgfgpsggsg	537-565	
tklpkkatktappalpt	566-582	
	583-677	GYDSEEEEESRPMSYDEKRQLSLDINKLPGEKLGRVVHIIQAREPSLRDSNPEEIEIDFE TLKPSTLRELERYVLSCLRKKPRKPYTIKKPVGKT
keelalekkrelekrl	678-693	
	694-715	QDVSGQLNSTKKPPKKANEKTE
sssaqqvavsrlsasssssdsssssssssssdtsdsds	716-753	
	754-754	G

Fig. 1. The result of automated segmentation of the human RING3 protein by sequence complexity. The protein is a product of the MHC Class II region and is of unknown function (Beck *et al.*, 1992). The sequence segments read from left to right and their order in the polypeptide runs from top to bottom, as shown by the central column of residue numbers. Most of the sequences in the high-complexity blocks 26–428 and 583–677 are strongly homologous to the Fsh protein of *Drosophila melanogaster* (Haynes *et al.*, 1989, 1992). Dashed underlines denote parts of the low-complexity segments that resemble corresponding segments of *Drosophila* Fsh in compositional bias but differ in detailed sequence and length. Other low-complexity regions show no similarity to *Drosophila* Fsh. The *Drosophila* homolog has additional low-complexity segments that have no counterparts in this human sequence.

in many eukaryotic proteins that are known from genetic experiments to be crucial in morphogenesis, embryonic development, transcriptional regulation, binding to chromatin and nuclear RNA, signal transduction, aspects of cellular structural integrity or extracellular structure and interactions. These are typically large, multidomain proteins, in many cases containing other modules whose likely structure and function can be identified by means of sequence homology. In some cases, domain homology provides part of a rationale for defining the boundaries of neighboring low-complexity segments, as deduced from the human and *Drosophila* homologs in the example shown in Fig. 1. DNA sequences encoding low-complexity segments provide evidence for a high frequency of fixation of mutational changes such as recombinational repeat expansion and DNA replication slippage in addition to nucleotide substitutions, deletion and insertions. Major open questions include the range of phenotypic consequences of these mutational events, the extent to which the observed spectrum of residue clusters in proteins is generated by mutational drive, and the magnitude of the genetic load imposed by this type of genome/phenotype flux.

Given this range of important aspects of research on low-complexity segments of protein sequences, it is necessary to base the computational analysis of them on precise formalisms of local complexity and compositional probability. In this report, we compare several such definitions for their utility in analyses of protein sequence databases and in optimal partitioning algorithms. These studies underpin the choice of definitions for the SEG algorithm for automated segmentation of amino acid sequences into regions of high and low complexity, details and applications of which are described separately (Wootton & Federhen, 1993).

2. LOCAL COMPLEXITY AND PROBABILITY

In this section, we develop definitions of the local "complexity" and "probability" of the possible subsequences of a linear biopolymer. Some of these definitions are based on a technique analogous to the enumeration of microstates in classical statistical mechanics. A more general technique that is rooted in statistics and information theory and achieves similar goals is the method of types (Csiszar & Korner, 1981; Cover & Thomas, 1991). In part, the theory presented here is an extension of the treatment used for short oligonucleotides by Konopka & Owens (1990a, b) and Salamon & Konopka (1992).

Let the biopolymer have N types of residues (an N-letter alphabet, usually $N = 4$ or 20) and consider a subsequence or *window* of length L residues. Statistical properties of each theoretically-possible or observed window may be defined at three levels: (1) *complexity state* or *numerical partition*, (2) *composition* or "*coloring*" or (3) *sequence*.

Complexity state

Each window has a number of occurrences of each of the N letters or residues. The complexity state of the window is defined by the sorted vector of these N numbers, irrespective of which specific letter or residue is assigned to each number. Thus each window of length L has a complexity state vector S_j whose N elements, n_i have the properties:—

$$0 \geqslant n_i \geqslant L, \quad \sum_{i=1}^{N} n_i = L,$$

and, in order to make a unique sorted vector that defines the state, $n_i \geqslant n_{i+1}$. These complexity state vectors, S_j, were named "repetition vectors" by Salamon & Konopka (1992). Each S_j represents a different partition of the integer L into N integers that sum to L. To generate and enumerate the S_j vectors is a known problem of restricted partitioning in number theory (Hardy & Wright, 1938). For the computations described here, data structures based on trees were implemented to generate the complete set of vectors for any values of N and L. The numbers of complexity state vectors for different window lengths L is shown for the $N = 20$ amino acid alphabet in Fig. 2.

The importance of representing sequence windows as numerical partitions is that these vectors have the property "complexity" that depends only on the numbers N, L and n_i, irrespective of the probabilities of occurrence of the states and their particular residue compositions. This may be illustrated by the following example of the 20-letter amino acid alphabet and window length 20, for which there are 627 possible states. These include the "least complex" vector

(20 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0)

and the "most complex"

(1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1).

These are both expected to be very improbable in typical random amino acid sequences. In contrast, some of the states of intermediate complexity, e.g.

(4 2 2 2 2 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0)

(3 2 2 2 2 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0)

typically occur relatively frequently. The second of these has slightly greater numerical complexity than the first. Two different measures that correspond to this intuitive concept of numerical complexity, "*complexity*", K_1, and "*entropy*", K_2, are defined below, together with different measures of the probabilities of complexity states based on different prior probabilities of the 20 amino acids.

Composition or "coloring"

Each complexity state vector, as defined above, has a number of different residue compositions corresponding to all possible assignments of the N letters (residues) to the N numbers in each vector S_j. These

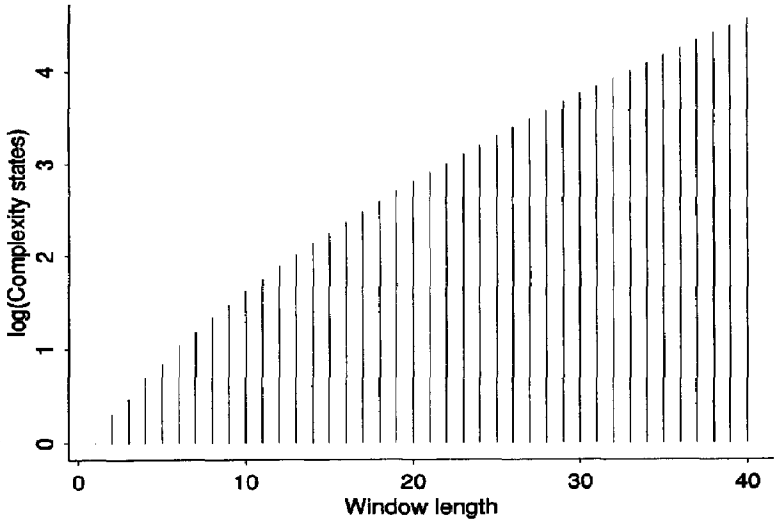

Fig. 2. The numbers of complexity states for the $N = 20$ amino acid alphabet at window lengths up to $L = 40$. The logarithmic scale of complexity states is to base 10.

compositions, which represent, for example, the familiar biochemical concept of peptide amino acid compositions, are named "colorings" in Salamon & Konopka (1992). The number of compositions of any complexity state, denoted F here following the usage "Farben", is given by:

$$F = \frac{N!}{\prod_{k=0}^{L} r_k!}. \quad (1)$$

Here, the values of r_k are the counts of the number of occurrences of each number in the complexity state vector S_j. Formally,

$$0 \leqslant r_k \leqslant N, \quad 0 \leqslant k \leqslant L, \quad \sum_{k=0}^{L} (r_k) = N$$

and, by convention, $0! = 1$.

In practice however, because of the restricted partitioning of L into S_j, only a few values from the possible ranges of r_k and k actually occur for any S_j and the computation uses only the non-zero r_k values. For example, for the vector

(3 2 2 2 2 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0)

F is computed from the r_k values (1, 4, 9, 6) corresponding to one 3, four 2s, nine 1s and six 0s. A unique situation occurs in the cases of window lengths L that are equal to or exact multiples of N, for which there is only one possible coloring of the vector of maximum complexity. For example, for $L = 40$ and the $N = 20$ amino acid alphabet, this vector is:

(2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2).

In contrast, for most of the complexity states and window lengths encountered in practice in protein sequence analysis, very large values of F are obtained from the 20-letter alphabet.

All colorings (compositions) of any numerical state have the same local complexity value, measured as K_1 or K_2, and can be considered to inherit this property from their complexity state vector. However, the *probabilities* may differ between colorings of a single complexity state, depending on the probabilities of occurrence, p_i, or the N different letters (residues). Only uniform probabilities of residues give equiprobable compositions for any complexity state (see below under Choice of prior probabilities).

Sequences

For each composition of a complexity state, as defined above, there exists a (usually) large number of different possible sequences. This number, Ω, is the multinomial coefficient characteristic of the complexity state and is the same for all compositions (colorings) of that state and depends only on N, L and n_i:

$$\Omega = \frac{L!}{\prod_{i=1}^{N} n_i!}. \quad (2)$$

The total number of possible sequences over all the complexity states of window length L is the number of permutations, N^L. Each sequence can be considered to inherit its attributes of complexity and probability from, respectively, its complexity state and composition.

"Entropy" and "complexity"

Two possible formal definitions of local compositional complexity now follow from the attributes of complexity state vectors described above.

"*Complexity*", K_1, is a measure analogous to the "factorial form" of the expression for message

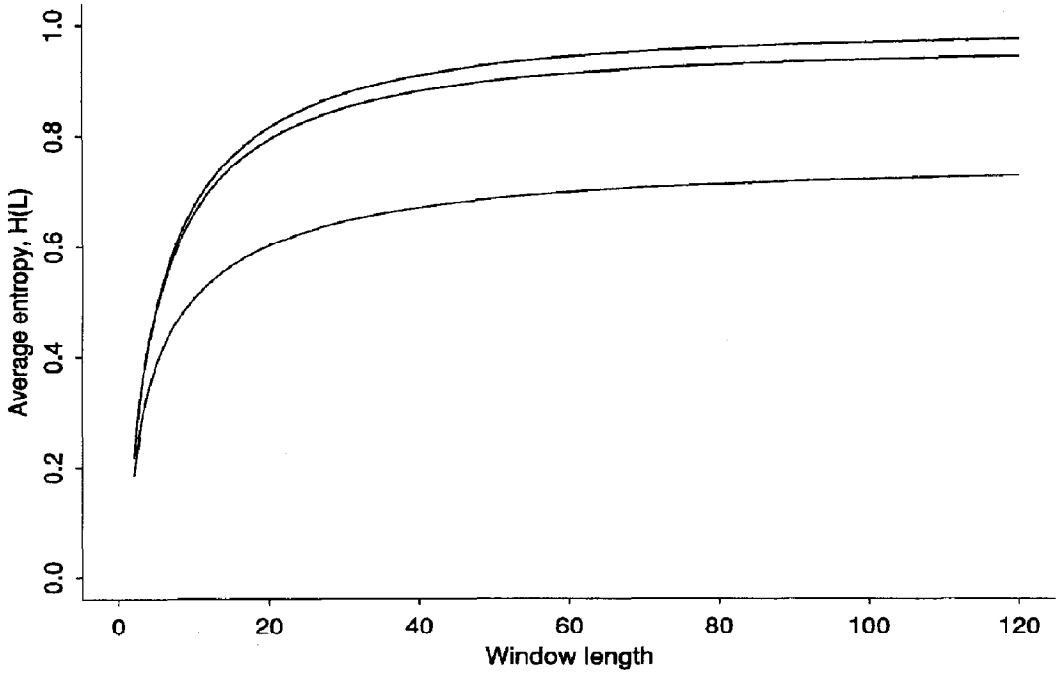

Fig. 3. The average entropy of amino acid compositions, H_L, as a function of window length, L. The three lines, from top to bottom, are for (1) uniform amino acid probabilities, (2) the amino acid frequencies in the Swissprot database, and (3) the amino acid frequencies of a low-complexity subset of the database [similar to that shown in Fig. 7(b)]. The limit entropies, H_{language}, characteristic of these three frequencies are, respectively, 1.0, 0.970 and 0.755. The values of H_L were obtained by simulation, using for each curve 1000 sample sequences each of 1000 residues drawn at random from the appropriate amino acid probabilities.

complexity used in minimal message length encoding (Boulton & Wallace, 1969), and was applied to oligonucleotides by Salamon & Konopka (1992). K_1 is based on Ω of equation (2), that is, the number of sequential rearrangements characteristic of each numerical state vector:

$$K_1 = \frac{1}{L} \log \Omega. \tag{4}$$

The definition of "*entropy*", K_2, is analogous to that of informational entropy (Shannon, 1948) and is given by:

$$K_2 = -\sum_{i=1}^{N} \frac{n_i}{L} \left(\log \frac{n_i}{L} \right). \tag{3}$$

The term "entropy" in biological sequence analysis requires clarification. K_2, as defined above for a single subsequence, has occasionally been called "entropy" in the literature, and could be considered to be an "observed entropy". But it is more accurate to describe K_2 as a log likelihood measure of the complexity of the numerical state of the subsequence. The strict usage of "entropy" is to describe the expected information content of a population, which is a property of a system as a whole. In the present application, "entropy" in this sense could be applied at two levels. First (denoted H_L) is the average value of K_2 over the probability distribution of all the complexity states for a given window length, L. Second (denoted H_{language}) is the average entropy characteristic of the global sequence language and the probabilities, p_i, of its N letters:

$$H_{\text{language}} = -\sum_{i=1}^{N} p_i \log p_i.$$

H_L approaches H_{language} asymptotically as L increases. This dependence is shown in Fig. 3 for uniform equiprobable amino acid frequencies and the frequencies of the total protein sequence database and a "low complexity" subset (defined below). In order to avoid possible confusions surrounding the word "entropy", we have adopted "low-complexity" rather than "low-entropy" as a general term for the classes of residue clusters and short-period repeats under study in this research. For similar reasons, we have also avoided the term "information content" of sequences.

K_1 and K_2 are different complexity measures that approach the same asymptotic limit at large values of L and n_i. The log-likelihood form K_2 can be derived from the factorial form K_1 if the n_i are large enough for Stirling's approximation to be valid (Boulton & Wallace, 1969; Salamon & Konopka, 1992; Harris, 1992). Since this is rarely the case for the analysis of typically-sized protein subsequences with the 20-letter

alphabet, we have explored the relationship between K_1 and K_2, which is plotted in Fig. 4 for window lengths 10, 20 and 40. These window lengths have, respectively, 42, 627 and 35,251 complexity states (Fig. 2). Note that the numbers of different values of K_1 and K_2 are fewer than the number of complexity states for any value of L. This is because some pairs or sets of states (different ones for K_1 and K_2) generate identical complexity values as a consequence of special numerical relationships in the vectors of n_i.

Clearly, K_1 and K_2 are broadly similar measures, and these results justify the use of either of them as a formal definition of local complexity in amino acid sequences. Other valid definitions of compositional complexity could be derived in principle from different premises, for example from a theory of algorithmic complexity shown by Chaitin (1975) to be formally equivalent to information theory. For the plots in Fig. 4, logarithms of the same base (20) were used to compute both K_1 and K_2, and these plots approach a line of slope 1 as L increases. The logarithms could equally well be taken to base 2 or base e, giving the familiar information units, bits or nats per residue. For the results presented in this report, logarithms are base 20 and K_1 is used as the standard complexity measure unless otherwise stated.

Choice of prior probabilities of amino acids

One goal of the work described in this report is to develop algorithms that make an unbiased view of the total abundance and full range of classes of low-complexity regions in protein sequences and sequence databases. For this purpose, it is appropriate to make the least committing assignment to the 20 amino acids of *uniform equal prior probabilities*. This choice is based on the assumption, which is justified in retrospect, that the current protein sequence database is a heterogeneous statistical mixture, such that the initially-unknown amino acid frequencies of the low-complexity subset or subsets need have no similarity to the frequencies observed for the total database. This assumption was based in part on prior knowledge of some of the well-recognized fibrous structural polypeptides and amino acid clusters whose near-homopolymeric or quasiperiodic segments have strikingly different amino acid compositions from the predominant globular proteins of the database. The assumption is fully justified *post hoc* by the residue frequencies actually found in low-complexity partitions of the database that were obtained as a result of this research.

The adoption of uniform prior probabilities of amino acids contrasts with the approach of Karlin

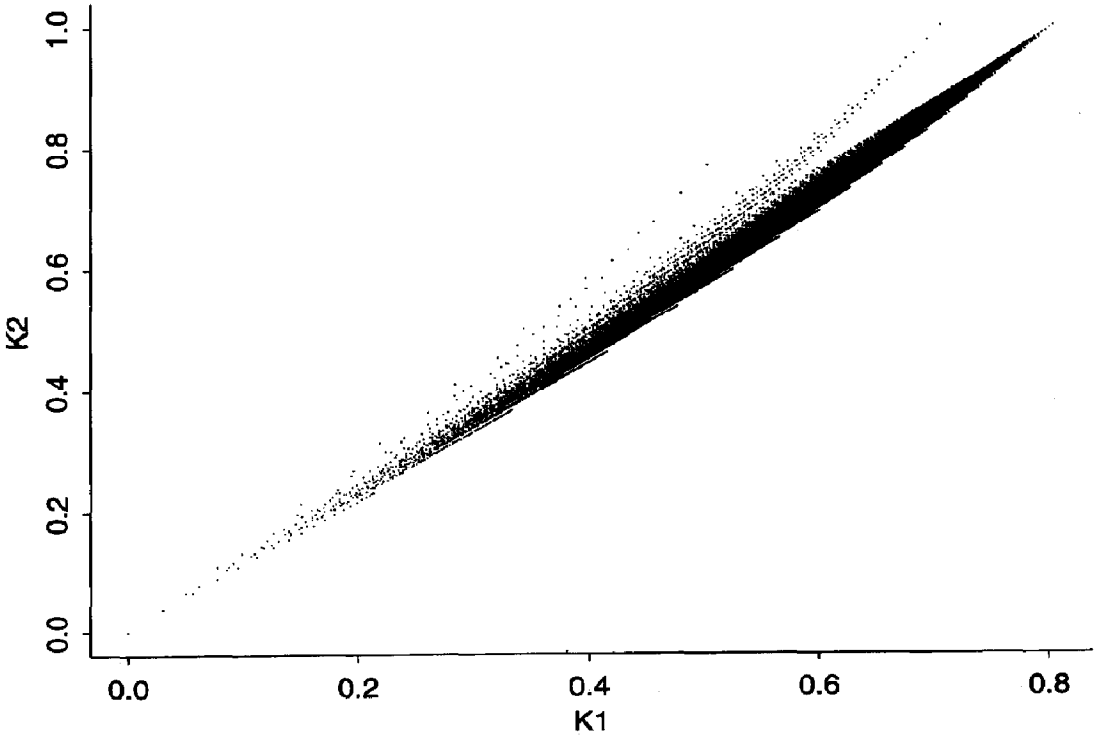

Fig. 4. The relationship between K_2 (log-likelihood or "entropy") and K_1 (factorial form) as measures of local complexity in amino acid sequences. Points, computed using equations (3) and (4), are co-plotted for window lengths 10 (42 complexity states), 20 (627 states) and 40 (35,251 states). These points clearly show the three zones of correlation that correspond to the three window lengths, and these approximate curves approach a slope of 1 as L increases.

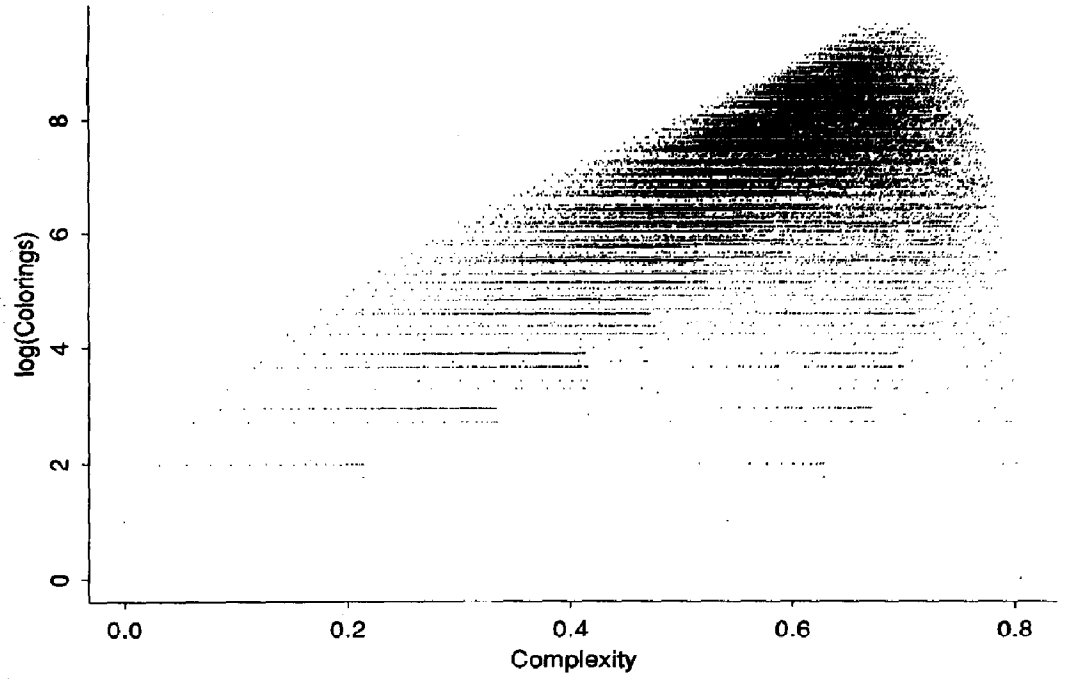

Fig. 5. The number of colorings (amino acid compositions) of each complexity state at window length 40 (35,251 states). Numbers of colorings, F, were computed by equation (1) and are plotted on a logarithmic scale to base 20 as a function of complexity, K_1, computed by equation (4).

et al. (1990, 1991) and Karlin & Brendel (1992) in their statistical methods for analysis of residue patterns, clusters and alignments. The latter methods depend on specific target frequencies of residues and residue patterns and employ the empirical frequencies that are observed in the sequences and databases under study as priors in first order random or higher order Markov chain models. The work reported here, which is mostly based on uniform probabilities of amino acids, has also explored the consequences of unequal amino acid frequencies by means of simulated random shuffles of sequence databases.

Probabilities of complexity states

For uniform residue frequencies, all compositions (colorings) of a complexity state S_j are equiprobable, and the probability of the state follows simply from equations (1) and (2):

$$P_0 = \Omega F / N^L. \tag{5}$$

Similarly, the probability of each of the F compositions of that state is Ω/N^L. P_0 is a point probability. The corresponding probability, Q_0, of observing any complexity state that is of equal or lesser probability to S_j has also been computed, using exhaustive enumeration of the cumulative probability distribution:

$$Q_0 = \sum_{S_{Q_0}} P_0(S_{Q_0}). \tag{6}$$

Here, S_{Q_0} is the set of numerical state vectors that are of equal or lesser probability to S_j, that is, $S \in S_{Q_0}$ iff $P_0(S) \leqslant P_0(S_j)$.

Enumeration of the *a priori* probabilities of complexity states, P_0 and Q_0, is computationally intensive for the 20-letter alphabet and the window lengths required for research on protein subsequences (typically 10–150 residues). We have generated look-up tables from *a priori* calculations of P_0 up to $L = 5000$ and Q_0 up to $L = 40$ for the uniform amino acid frequencies. For *a priori* calculations from unequal frequencies, the probabilities of all the colorings of each complexity state must be enumerated individually, and this is feasible only for very short windows. The scale of this computational problem is illustrated by Fig. 5, which shows the number of colorings F as a function of K_1 for each of the 35,251 complexity states of window length 40.

Probabilities of observed sequence segments

In contrast to enumerations *a priori*, the compositional probabilities of any *observed* sequence windows can be calculated rapidly for any prior probabilities, p_i, of the 20 amino acids (although for most purposes reported here uniform priors, $p_i = 0.05$, are appropriate for reasons described above). If the numbers of each residue observed in a sequence segment of length L are n_i, the *multinomial probability*, $P_{\text{multinomial}}$, of the composition of that segment is:

$$P_{\text{multinomial}} = \Omega \prod_{i=1}^{N} p_i^{n_i} \quad (7)$$

where Ω is the multinomial coefficient given by equation (2). As the n_i increase to large values, an approximation, commonly called "chi-square" may be used as an estimate of compositional probability:

$$\chi^2 = \sum_{i=1}^{N} \frac{(f_i - p_i)^2}{p_i} \quad (8)$$

where the f_i values are the observed frequencies, n_i/L, of the 20 amino acids. This approximation is not strictly χ^2 distributed, and there is no established theoretical basis for its application to this problem. Nevertheless, it has proved to be a useful supplementary heuristic in optimal segmentation algorithms for a few amino acid sequences, even in cases with some n_i values <5.

Another approximate measure is the statistic, D_{Clark}, described by Clark (1952) and suggested to us by Jean-Michel Claverie. This is an extension of the coefficient of divergence for use with multiple characters (in this application, the $N = 20$ amino acids):

$$D_{\text{Clark}} = \sum_{i=1}^{N} \left(\frac{f_i - p_i}{f_i + p_i}\right)^2. \quad (9)$$

D_{Clark} is potentially applicable (Clark, 1952) as a statistic that is relatively independent of sample size (in this application, window length, L).

Implementations

The methods described in this report have been implemented as a structured and documented suite of programs, SPLEX, in C language for Sun and Silicon Graphics workstations and servers. This software including source code is available by anonymous ftp from ncbi.nlm.nih.gov (130.14.20.1) in subdirectory pub/splex. This includes libraries for handling sequences and sequence databases in the FASTA format of Pearson & Lipman (1988), generalized windowing functions, and options for production of descriptive statistics from sequence databases and shuffled databases, in addition to the look-up tables of P_0 and Q_0 and the specific algorithms for calculations of all the theoretical and observed functions defined above. The C language uses the NCBI tools for ease of portability of code (NCBI Programmer's Reference, 1992, NCBI Software Development Kit, National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Building 38A, 8600 Rockville Pike, Bethesda, MD 20894, U.S.A.). For the data and analyses presented here, additional manipulations were made to the values produced by SPLEX using the Splus programming, statistical analysis and graphical system (Becker *et al.*, 1988).

3. PROTEIN SEQUENCE COMPLEXITY: OBSERVED AND THEORETICAL DISTRIBUTIONS

In this section, we compare the theoretical probability distribution of protein subsequence complexity with the frequencies observed in the protein sequence database and a random shuffle of this database. Swissprot [release 22.0, May 1992 (Bairoch & Boeckmann, 1992)] was chosen on grounds of its well-structured format which enables easy parsability, its relative completeness of annotation, and the fact that unnecessary redundancy due to alternative versions of the same sequence has been removed by careful manual merging procedures. To make an appropriate population of protein sequences for statistical analysis, Swissprot 22.0 was further reduced by removal of (1) the six artificial "warning" entries constructed to represent Alu sequence subclasses, and, to restrict the analysis to the 20-letter alphabet, (2) the 610 entries that contain the amino acid redundancy codes, B, Z and X. The resulting database, denoted "Swissprot" in this paper, comprised 24,434 entries containing a total of 8,268,602 residues.

"Shuffled Swissprot" is a first-order random shuffle of this database that preserves the exact total amino acid composition and set of sequence lengths of the database. For each shuffle, a randomization of the 8,268,602 amino acids was first constructed, by taking one amino acid at a time, then this string was segmented into the exact lengths of the 24,434 original sequences, thus preserving the same order of sequence lengths. This procedure ensures that the sequence end-effects are the same for analyses of subsequences of both Swissprot and Shuffled Swissprot. Ten random shuffles of this type were generated with different seeds, and one was chosen as typical [from inspection of *log (frequency)* vs K_1 plots for $L = 20$] for further analysis.

A small number of complexity states predominate

Figure 6 shows the theoretical discrete probability distributions of K_1 for the complexity states of window lengths, 10, 20 and 40 [Fig. 6(a)], together with the corresponding distributions of the frequencies of these states in Swissprot [Fig. 6(b)] and Shuffled Swissprot [Fig. 6(c)]. These plots show the characteristic, spiky, irregular, dependence of probability on complexity: at the larger window lengths, there are several cases of complexity states that have very close K_1 values but orders of magnitude different probabilities. At any window length, the specific pattern of spikiness is essentially the same for the observed and shuffled distributions, demonstrating that this property is primarily a consequence of the numerical relationships of the complexity states, rather than an effect of different amino acid frequencies or sampling variation.

Another striking feature of these distributions is that a small fraction of the complexity states, which

are clustered at similar K_1 values, contain almost all the probability. Taking the theoretical distributions calculated from uniform amino acid frequencies: For $L = 10$, 8 out of the 42 states (19%) contain 97% of the probability. For $L = 20$, 37 out of the 627 states (5.9%) contain 98% of the probability, and the most probable two states

(3 2 2 2 2 2 1 1 1 1 1 1 1 0 0 0 0 0 0 0)

(3 2 2 2 2 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0)

contain 18% of the probability. For $L = 40$, 798 out of the 35,251 states (2.3%) contain 96% of the probability. Thus, as L increases, a decreasing fraction of the complexity states fall into the "typical" set of relatively high probability, and the great majority of states ("non-typical") are of extremely low probability.

This phenomenon, as seen in Fig. 6, illustrates two well-established properties of information theory, namely *ergodicity* and *asymptotic equipartition* (Shannon, 1948; Cover & Thomas, 1991). As a result of ergodicity, the *typical set* of high-probability complexity states tend to have values of K_1 or K_2 that are close to the average entropy, H_L, characteristic of the window length, L, and increasingly approach the limit of the average entropy of the language, H_{language}, with increasing L. As a result of asymptotic equipartition, with increasing values of L, the states of the typical set, that contain almost all the probability, become increasingly equiprobable and contain only a very small fraction of the total states. These properties are also illustrated by the trends seen with increasing window length in Figs 3 and 4. In Fig. 6, the limit entropies of the languages are 1.0 for the theoretical and 0.97 for Swissprot and Shuffled Swissprot. Thus many of the striking properties of these sequence complexity distributions are consequences of well-established theorems of information theory, and are essentially the same for both observed and theoretical distributions. Other numerical relationships are seen in the theoretical distributions when P_0 is plotted on a logarithmic scale as a function of complexity [Fig. 7(a)]: First, $\log(P_0)$ shows a relatively strong, roughly linear correlation with K_1 and can therefore be considered to be an approximate complexity measure. Second, parallel lines of

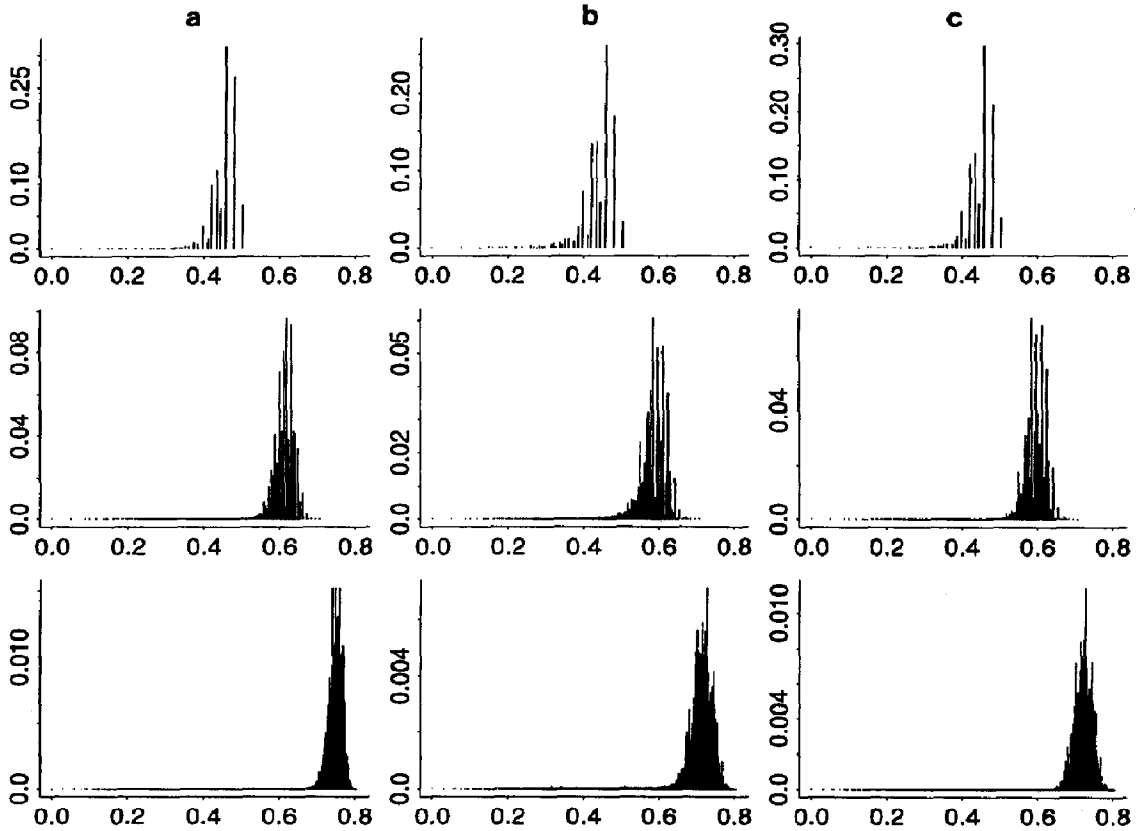

Fig. 6. Theoretical and observed distributions of the complexity measure, K_1. The horizontal axis of all the plots is the complexity, K_1, computed using equation (4), at window lengths, L, of 10 (top row of plots), 20 (middle row) and 40 (bottom row). (a) The theoretical probabilities of the complexity states [P_0, equation (5)] based on uniform amino acid frequencies. (b,c) The frequencies of the complexity states observed in, respectively, Swissprot and Shuffled Swissprot.

striation arise [Fig. 7(a)] from subsets of the complexity states that have the same number of compositions, F.

The excess of low-complexity subsequences in natural proteins

The frequency distributions of complexity for Swissprot (Fig. 6) appear to resemble those for the Shuffled Swissprot at the high-complexity end of the range, but the real sequences show a large excess of the low-complexity states. The magnitude of this excess is seen more clearly from plots of the log (to base 20) of the frequencies or probabilities of the states [Fig. 7(b)], and in Log-Odds Ratio (LOR) plots (Fig. 8). The Log-Odds Ratio compares the observed and theoretical distributions:—

$$\mathrm{LOR} = \log\left(\frac{\text{observed}}{\text{expected}}\right) = \log\left(\frac{f(S_j)}{P_0(S_j)}\right). \quad (10)$$

LOR, under the name "surprisal", was shown by Salamon & Konopka (1992) to have interesting properties when plotted as a function of the complexity, K_1, of short oligonucleotides. Different subsets of nucleotide sequences (exons and introns from primate, viral and organellar genomes) all gave approximate straight lines that showed characteristic differences in slope between exons and introns. The approximate linearity of these plots has not yet been fully explained (see accompanying paper by Salamon *et al.*, 1993), but the differences in slope and intercept could clearly be exploited in automated discrimination methods for different classes of nucleotide sequences.

The LOR plots for the protein sequences of Swissprot at window lengths 10, 20 and 40 [Fig. 8(a)] show a more elaborate structure than the corresponding plots for short oligonucleotide windows (Salamon & Konopka, 1992). Clearly, there is a strong correlation between LOR and K_1, with many of the complexity states tending to be concentrated in central lines that show different slopes in the low-complexity and high-complexity regions. The corresponding plots for Shuffled Swissprot [Fig. 8(b)] show only the high complexity states, since the expected numbers of windows in the low-complexity states are very close to zero for a random database of 8,268,602 residues. The plots for Shuffled Swissprot are very closely similar to the high-complexity part of the corresponding Swissprot plots, showing a similar concentration of states on a central line and a matching pattern of scattered points. The scatter of LOR values is evidently a consequence of the numerical relationships of the state vectors rather than a sampling effect (this is confirmed below following complexity partitioning of the database). A satisfactory theoretical explanation for the characteristics of these plots has not yet been developed, and the corresponding plots for nucleotide sequences at large window lengths may be more revealing for this purpose (Salamon *et al.*, 1993).

4. HEURISTICS FOR OPTIMAL SEGMENTATION ALGORITHMS

In this section, we compare different measures of compositional complexity and probability for their practical utility in protein sequence and database analysis. Some algorithms have been published for optimal segment identification in protein sequences but these used intrinsic residue propensities such as hydrophobicity (Auger & Lawrence, 1989; Chappey & Hazout, 1992). There is no prior experience of the use of local complexity in such optimal segmentation methods, and exploration of the characteristics of different measures is necessary.

First, all the measures defined above were compared in simple "profiling" of sequences at fixed window lengths in one-residue steps. A typical sample of the results at window length 12 is shown in Fig. 9. The region of sequence profiled, which contains segments of contrasting complexity, is residues 400–500 of the human RING3 protein shown in Fig. 1. Clearly, all these measures give broadly similar results. As expected from the analyses shown in Figs 4 and 7, the measures K_1, K_2, and $\log(P_0)$ (also Q_0 and $P_{\text{multinomial}}$, not shown) give almost identical profiles, and χ^2 is a close approximation to these. D_{Clark} shows less clear-cut contrast between regions of high and low complexity, consistent with the greater averaging of information that is characteristic of the coefficient of divergence.

Also shown, labeled "offset = 3", is a typical result of a method based on self-self sequence alignment. The values for this method are similarity measures for subsequences aligned at different offsets. This method thus gives a relatively direct estimate of the autocorrelation within a window, and is the basis of an algorithm (States & Claverie, 1993) for localization of low-period repeats. The self-alignment method also provides a possible means of very approximate localization of low-complexity regions, whether or not periodic, as shown in the Fig. 6 profile and other similar results, although this method gives relatively inaccurate definition of the boundaries of these regions.

This study using profiling methods therefore supports the use of any of these measures for approximate, first-pass identification of regions that contain low-complexity subsequences within them. In the SEG algorithm, K_2 is used for this purpose on grounds of computational efficiency.

Second, the values for all the complexity and probability measures were computed over a range of window lengths ($L = 2$–40) for several protein sequences that contain low-complexity regions. Results for the human RING3 protein are given as three-dimensional perspective plots in Fig. 10. In these landscapes, low-complexity and low-probability are upwards on the y-axis, so that the local optima sought by the segmentation algorithm correspond to the highest peaks within local regions of the ridges.

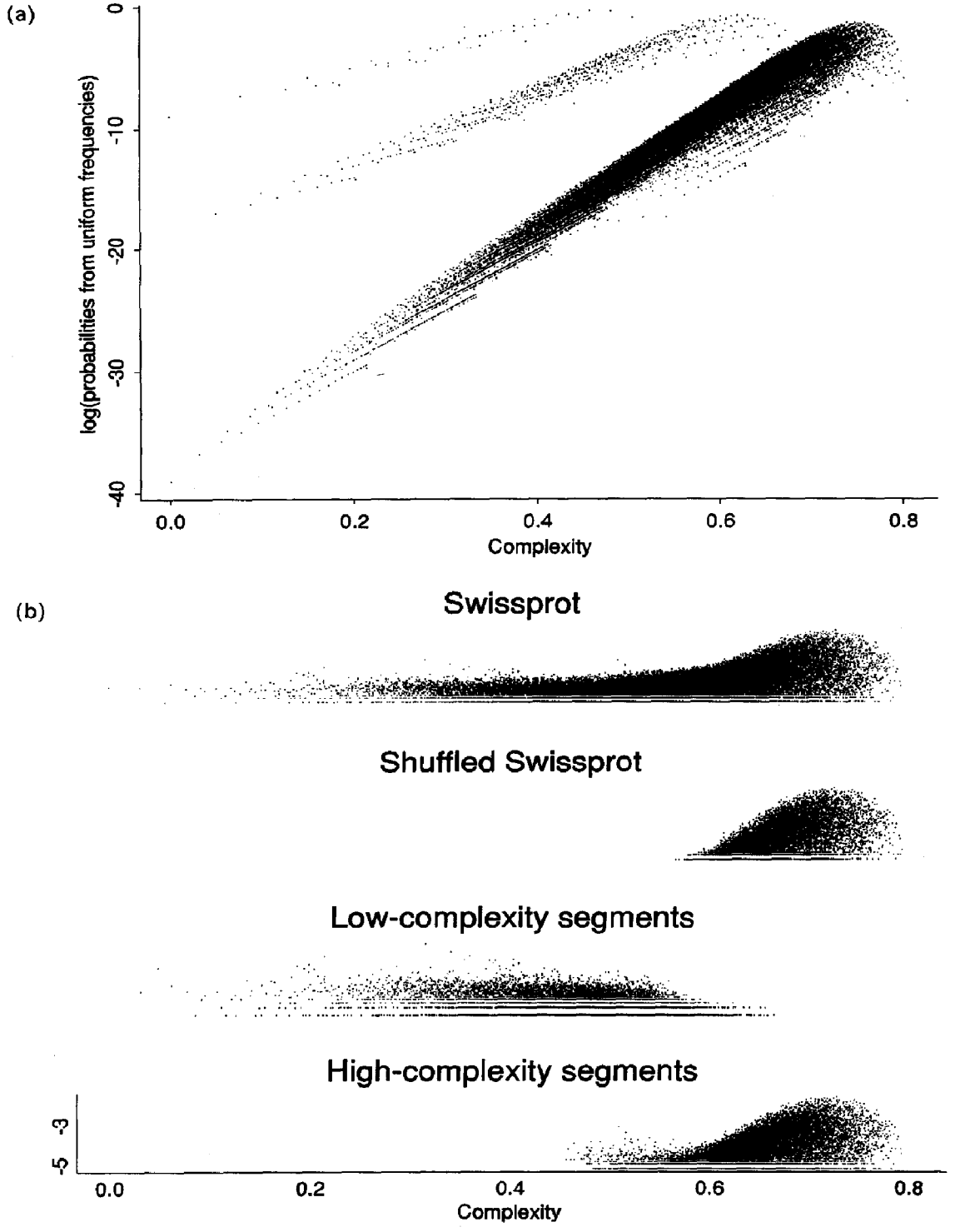

Fig. 7. Log-probability and log-frequency distributions of the complexity K_1. (a) Theoretical P_0 distributions calculated for uniform amino acid frequencies. Points are co-plotted for window lengths $L = 10$ (42 states), 20 (627 states) and 40 (35,251 states), giving the three clearly distinct regions of correlation from top to bottom respectively. (b) Observed log-frequency distributions for window length $L = 40$. Each of the four plots has a vertical log-frequency axis similar to that indicated for the bottom plot. The horizontal scale of K_1 is the same for all four distributions. The separate, complementary distributions for low-complexity and high-complexity segments were derived from Swissprot by automated partitioning using the SEG algorithm. A moderately stringent threshold of complexity was used for the first pass of this partitioning to ensure that the low-complexity fraction was relatively uncontaminated with spurious high-complexity segments. Consequently, a few low-complexity segments ($K_1 \leq 0.58$) are evident in the high-complexity set.

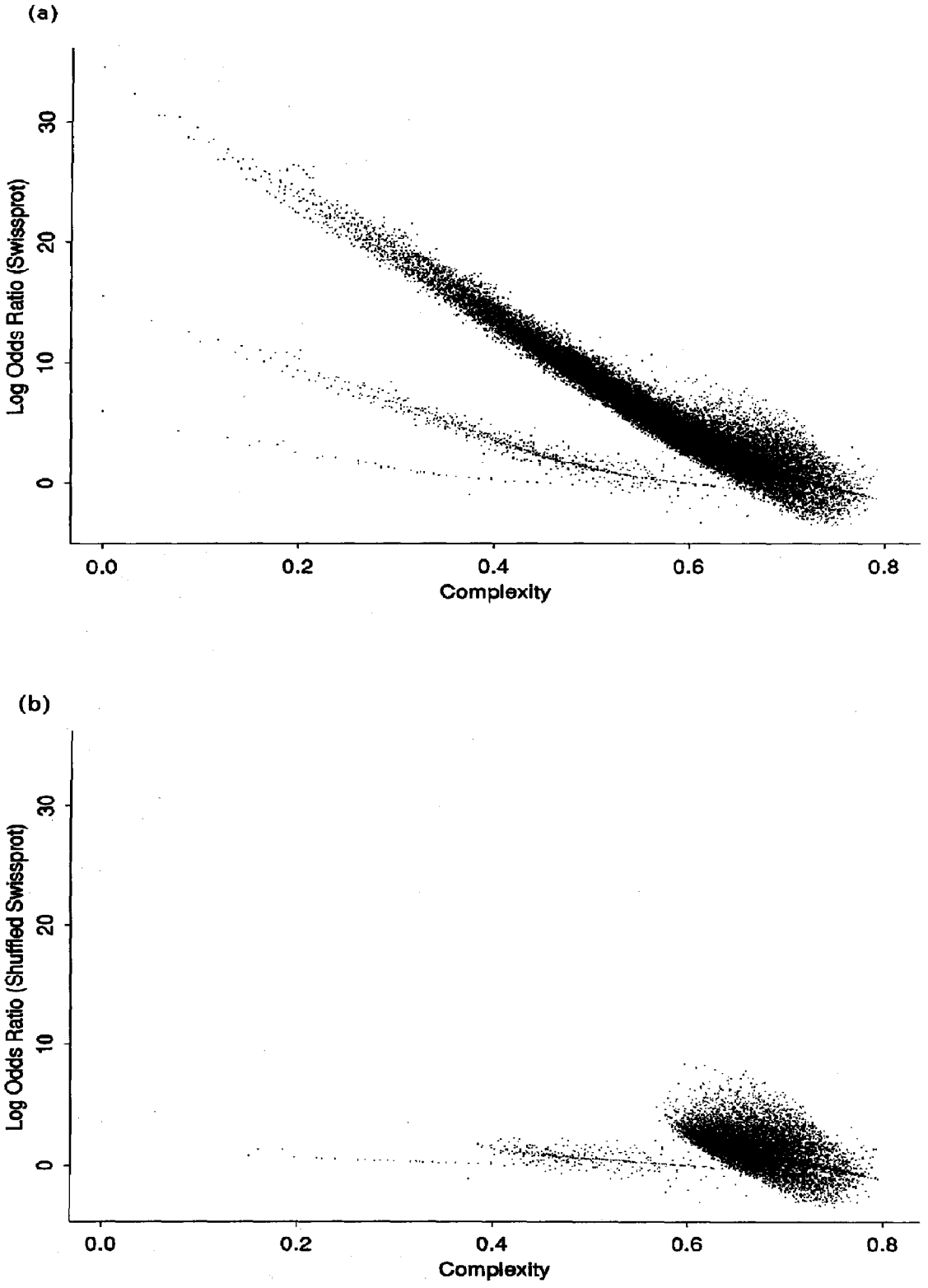

Fig. 8. LOR plots as a function of complexity, K_1, for Swissprot (a) and Shuffled Swissprot (b). Points were computed using equations (10) and (4). As in Figs 4 and 7(a), points are co-plotted for window lengths $L = 10$ (42 states), 20 (627 states) and 40 (35,251 states), giving the three clearly-distinct elongated clusters from, respectively, bottom-left to top-right.

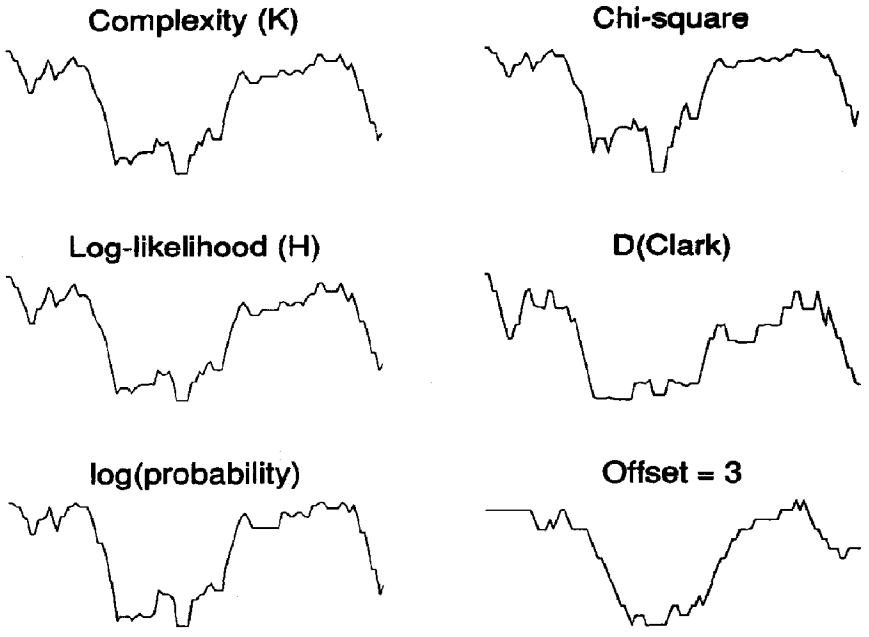

Fig. 9. "Profiles" of residues 400–500 of the human RING3 protein using different complexity or probability measures. Complexity, log-likelihood, log-probability ($\log P_0$), χ^2 and D_{Clark} were computed respectively by equations (4), (3), (5), (8) and (9), using all subsequences of length $L = 12$. "Offset = 3" is explained in the text. For visual comparability, the measures are all normalized to the same range on the vertical axes, using sign reversal where appropriate to make the bottom of the profiles correspond to the lowest complexity or probability. The horizontal axes are the same for all the plots and represent position in sequence.

The differences between the plots in Fig. 10 reveal features that are critical for the performance of these measures as optimization heuristics. One feature is the distinction between "log-probability" and "complexity" measures: $\log(P_0)$ [and the very similar $\log(Q_0)$ and $\log(P_{\text{multinomial}})$ not shown in Fig. 10] and the χ^2 approximation are essentially log-probability measures. These have flat baselines (Fig. 10) and the peaks correspond precisely to the subsequences of lowest probability as determined by the different measures. Thus log-probability measures, calculated over all subsequences within a defined sequence, provide a simple and direct route to the optimal low-complexity segment, given that the optimum is defined as the segment of lowest probability. Experience with many natural proteins has now justified the view that this is indeed the most satisfying definition of "optimum" for the purpose of differentiating regions of contrasting complexity in natural proteins. Boundaries produced by this $\log(P_0)$ definition partition the compositional information efficiently between neighboring low- and high-complexity segments with minimal cross-contamination. This method also seems biologically appropriate because many such segments contrast in functional and evolutionary characteristics, as illustrated by Fig. 1 and other examples partitioned by means of the SEG algorithm. In our implementation of SEG, $\log(P_0)$ is used as default (taken from a look-up table) and the other measures are available as options. The χ^2 heuristic is advantageous in some cases because it tends to give more stringent optimal segments, more sharply contrasting with their flanking sequences than those produced using $\log(P_0)$. This property is reflected in the more jagged peaks seen in Fig. 10.

In contrast to log-probability measures, complexity, K_1, (and the very similar log-likelihood, K_2) and D_{Clark} have curved baseline that approach limits asymptotically with increasing L. These curves can be flattened empirically using average values from similations, as shown, for example, for K_2 in Fig. 3. However, uncertainties in such correction factors, particularly for the shorter values of L, make these measures less reliable than the log-probability measures for definition of the boundaries of optimal low-complexity segments. Such corrected complexity measures can, however, be used to find local minima of complexity that are not necessarily optimal. This is sometimes justified as the basis of a very rapid computation by the method of steepest descents, once a general region of low-complexity sequence has been defined approximately and "trimming" from both ends is required. The log-likelihood measures, K_2, corrected by means of the top curve in Fig. 3, has been used to implement such steepest-descent-trimming in the FSEG program, a very rapid but approximate variant of SEG.

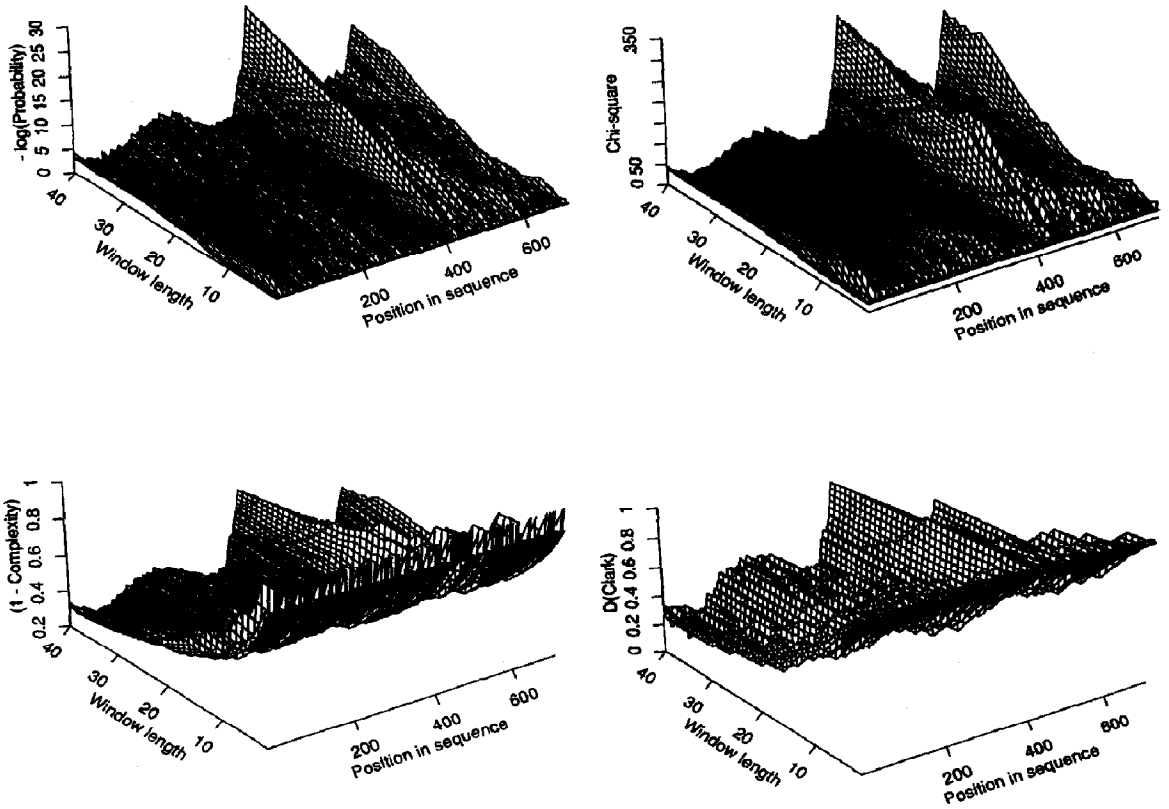

Fig. 10. Landscapes of subsequence complexity and probability for the human RING3 protein over a range of window lengths from $L = 2$ to 40. Log-probability ($\log P_0$), complexity, K_1, χ^2 and D_{Clark} were computed respectively by equations (5), (4), (8) and (9). Not shown are plots for Q_0 and $P_{\text{multinomial}}$, which gave almost identical landscapes to P_0, and for lot-likelihood, K_2, which gave very similar results to complexity, K_1. All the perspective plots are viewed from the same angle relative to the axes, and the tops of the vertical axes correspond to the lowest complexity or probability. The number on the "position in sequence" axes refer to the N-terminal residue of each window. Incomplete windows at the C-terminal end of the RING3 protein are not plotted.

5. COMPLEXITY PARTITIONS OF THE PROTEIN SEQUENCE DATABASE

The striking excess of low-complexity segments in proteins in the Swissprot database shows clearly in Figs 7(b) and 8(a), demonstrating the statistical heterogeneity of the database. Following optimal segmentation of the database using SEG [Fig. 7(b), bottom two plots], the low-complexity set is itself a statistical mixture and consists of a number of compositional classes, each of which has one or a few predominant amino acids. For example, there are "glutamine-rich" and "glycine-proline-rich" classes, each of which contain a number of subclasses differing in possible structures and functions. The results of such statistical and functional classification of low-complexity segments is reported separately (Wootton & Federhen, 1993; and unpublished work).

It is also interesting to ask if the high-complexity subset of the database [the "typical 85%", Fig. 7(b), bottom plot] is statistically homogeneous and whether it can be adequately approximated by a random model. This is important because most sequence alignment methods and commonly-used database search algorithms such as BLAST and FASTA (Altschul *et al.*, 1990; Pearson & Lipman, 1988), are based on first-order random models in which sequences are treated as random draws sampled with replacement from the observed single residue frequencies. These search methods give spurious results with query sequences that contain highly non-random low-complexity sequences. Comparisons of the high-complexity subset of Swissprot with Shuffled Swissprot suggest that the former does indeed approximate to first-order random. This shows in the similar distributions of complexity [Fig. 7(b) and comparison of Fig. 8(a,b)], and particularly clearly in the point-by point comparison of the LOR plots (Fig. 11). Figure 11 illustrates the fact that these values are determined by the numerical relationships of the complexity state vectors: the points for the high-complexity subset of Swissprot are clearly very close to those of the shuffled database. This supports the conclusion that the random expectation used as

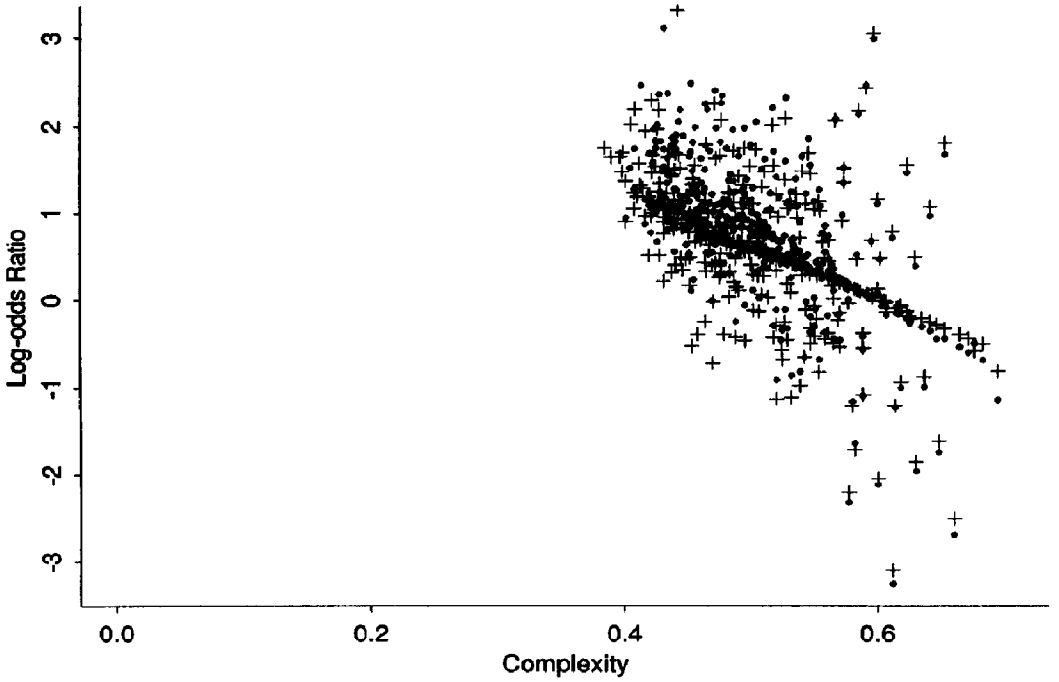

Fig. 11. Point-by-point comparison of the LOR plots for the high-complexity subset of Swissprot (●) and Shuffled Swissprot (+) at window length 20. Points were computed using equations (10) (LOR) and (4) (complexity, K_1).

the statistical basis of current database search algorithms (Karlin & Altschul, 1990; Altschul *et al.*, 1990) is adequate for the typical 85% of protein subsequences. The random model is not appropriate for low-complexity segments, and statistically valid results can be obtained from present database search methods only if these segments are filtered from query sequences. Further research is required to develop comparison and search methods appropriate for low-complexity regions of proteins.

Acknowledgements—We thank Chip Lawrence, John Spouge, Peter Salamon, Andrzej Konopka, Warren Gish, Stephen Altschul, Jean-Michel Claverie and David States for helpful discussions.

REFERENCES

Altschul S. F., Gish W., Miller W., Myers E. W. & Lipman D. J. (1990) *J. Mol. Biol.* **215,** 403.
Auger I. E. & Lawrence C. E. (1989) *Bull. Math. Biol.* **51,** 39.
Bairoch A. & Boeckmann B. (1992) *Nucleic Acids Res.* **20** Suppl., 2019.
Beck S., Hanson I., Kelly A., Pappin D. J. & Trowsdale J. (1992) *DNA Seq.* **2,** 203.
Becker R. A., Chambers J. M. & Wilks A. R. (1988) *The New S Language*. Wadsworth & Brooks, Pacific Grove, Calif.
Boulton D. M. & Wallace C. S. (1969) *J. Theor. Biol.* **23,** 269.
Chaitin G. J. (1975) *J. Assoc. Comp. Mach.* **22,** 329.
Chappey C. & Hazout S. (1992) *CABIOS* **8,** 255.
Clark P. J. (1952) *Copeia* **2,** 61.
Cover T. M. & Thomas J. A. (1991) *Elements of Information Theory*. Wiley, New York.
Csiszar I. & Korner J. (1981) *Information Theory: Coding Theorems for Discrete Memoryless Systems*. Academic Press, New York.
Hardy G. H. & Wright E. M. (1938) *An Introduction to the Theory of Numbers*. Oxford University Press, London.
Harris P. L. (1992) An information-theoretic approach to modelling the degree of repetition in genome sequences. M.S. thesis, Applied Mathematics, San Diego State University, Calif.
Haynes S. R., Mozer B. A., Bhatia-Dey N. & Dawid I. B. (1989) *Dev. Biol.* **134,** 246.
Haynes S. R., Dollard C., Winston F., Beck S., Trowsdale J. & Dawid I. B. (1992) *Nucleic Acids Res.* **20,** 2603.
Karlin S. & Altschul S. F. (1990) *Proc. Natl. Acad. Sci. U.S.A.* **87,** 2264.
Karlin S., Blaisdell B. E. & Brendel V. (1990) *Meth. Enzym.* **183,** 388.
Karlin S., Bucher P., Brendel V. & Altschul S. F. (1991) *Annu. Rev. Biophys. Biophys. Chem.* **20,** 175.
Karlin S. & Brendel V. (1992) *Science* **257,** 39.
Konopka A. K. & Owens J. (1990a) In *Computers and DNA* (Edited by Bell G. I. & Marr T.), p. 147. Addison Wesley, Reading, Mass.
Konopka A. K. & Owens J. (1990b) *Gene Anal. Tech.* **7,** 35.
Pearson W. R. & Lipman D. J. (1988) *Proc. Natl. Acad. Sci. U.S.A.* **85,** 2444.
Salamon P. & Konopka A. K. (1992) *Comput. Chem.* **16,** 117.
Salamon P., Wootton J. C., Konopka A. K. & Hansen H. K. (1993) *Comput. Chem.* **17.**
Shannon C. E. (1948) *Bell Syst. Tech. J.* **28,** 379, 623.
States D. J. & Claverie J. M. (1993) In preparation.
Wootton J. C. & Federhen S. (1993) Submitted.