Brute force estimation of the number of human genes using EST clustering as a measure

by D. B. Davison
J. F. Burke

A current question of considerable interest to both the medical and nonmedical communities concerns the number of human transcription units (which, for the purposes of this paper, are "genes") and proteins. Even with the recent announcement of the completion of the draft sequence of the human genome, it is still extremely difficult to predict the number of genes present in the genome. There are several methods for gene prediction, all involving computational tools. One way to approach this question, involving both computation and experiment, is to look at copies of fragments of messenger ribonucleic acid (mRNA) called expressed sequence tags (ESTs). The mRNA comes only from a gene being expressed, or translated, into RNA; by clustering mRNA fragments, we can try to reconstruct the expressed gene. While the final result is a very rough representation of the "true expressed transcript," it is probably within 20% of the real number. Here, we review the issues involved in EST clustering and present an estimate of the total number of human genes. Our results to date indicate that there are some 70 000 transcription units, with an average of 1.2 different transcripts per transcription unit. Thus, we estimate the total number of human proteins to be at least 85 000. The total number of proteins will be higher because of post-translational modification.

Introduction

A landmark in human knowledge was recently reached with the announcement of the completion of the draft sequence of the human genome. This represents the opening of a completely new phase in understanding what it is to be human and in the factors that influence many human diseases. One key datum of great interest is the number of human genes and proteins. Although the question is simple enough to state, in practice it is very hard to answer. Not least in difficulty is the issue of carefully defining what one means by the term "gene."

0018-8646/01/$5.00 © 2001 IBM

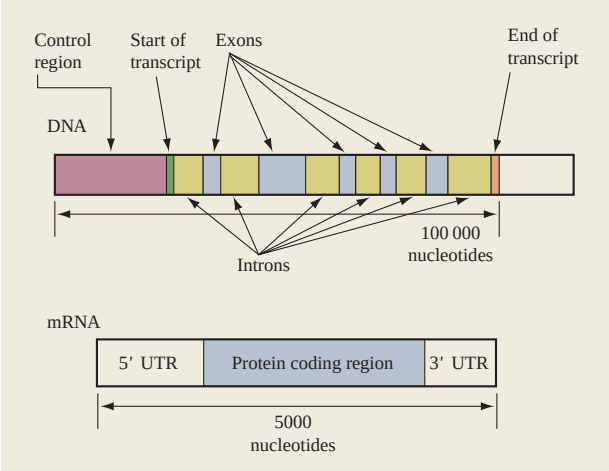

Figure 1

Schematic view of some features of a eukaryotic transcription unit. The top line represents the DNA; the bottom, the messenger RNA. The region between "start of transcript" and "end of transcript" is copied into mRNA. Then the internal noncoding regions (introns) are removed, leaving the protein coding regions (exons) together with some upstream sequence [the 5′ untranslated region (UTR)] and downstream sequence (3′ UTR).

Before doing that, we first present background information, define our methodology, and then present our estimate of the number of human genes and proteins.

A brief introduction to molecular biology

The central dogma of molecular biology states that information flows from DNA to RNA to proteins. Every cell has its entire complement of instructions encoded in its deoxyribonucleic acid (DNA). DNA is made up of four subunits, or bases, termed A, C, G, and T for adenine, cytosine, guanine, and thymine. These subunits are connected into very long chains called chromosomes. Most chromosomes are linear, although bacteria (with some exceptions) have circular chromosomes. Most species have a different number of pairs of chromosomes, even those that are genetically closely related. However, there is enough similarity between regions (hundreds of thousands to millions of base pairs) of any two species' chromosomes that one can usually find a correspondence. Such a correspondence is called *synteny*. For instance, there is enough synteny between human and bovine chromosomes that one can use the human genetic map to locate a small number of genes in the bovine genome.

Humans have 23 pairs of chromosomes, 22 autosomes and two sex-determining chromosomes, the X and the Y. They range in size from 50 to 263 million base pairs.[1] The total size of the human genome is known to be at least three billion bases spread out across the 24 chromosomes.

A gene possesses a number of characteristics, as diagrammed in **Figure 1**. It is the basic unit of heredity, that which is passed from generation to generation. It encodes a protein (or, in some cases, a structural RNA). The gene is copied into a messenger ribonucleic acid (mRNA) in a process called transcription. That mRNA is then translated into a protein sequence by a very large protein-RNA molecular machine called the ribosome. Each gene has a nontranscribed region upstream and downstream of the coding region involved in the regulation of gene expression. Even the part that is transcribed into mRNA has portions that are cut out afterward (introns), leaving behind the instructions for making the protein (exons). Furthermore, not all genes are expressed (translated into protein) in all cells. Some genes have a role in only one kind of tissue (e.g., the lung) and would not be expressed in another tissue (e.g., the skin).

These features of the gene—introns, exons, and regulatory regions—are still relatively poorly understood. The signals that distinguish one from the others are subtle. Therefore, at present the best way to locate genes is by experiment. Researchers isolate cells of interest and extract the total RNA, a process called "library construction." The mRNAs are separated from the structural RNAs, then copied into DNA with a special enzyme. These pieces of DNA are called complementary DNA (cDNA). During this process the RNA is frequently partially degraded. The cDNA fragments are called expressed sequence tags (ESTs). A particular gene may not be expressed in a particular cell type, so it will not be present in the library. Other genes may be very highly expressed, so there will be many ESTs from those genes. All cells have certain metabolic functions and structural features that they must have to live, so those "housekeeping" ESTs will be present in all cells.

By examining all of the ESTs available from cells in a wide variety of tissues and developmental states, one can get an idea of how many expressed genes there are. Another complication is that a gene may produce more than one transcript by including or excluding specific exons or by altering the length of a specific exon. This is known as "alternative splicing." As mentioned above, the number of ESTs does not correlate directly with the number of genes because an mRNA may be broken into many pieces during experimental manipulation. Consequently, ESTs must be reassembled into intact cDNAs in order to obtain an estimate of the number of genes in the genome. Each gene may also have more than one transcription product, or isoform, and each isoform may be expressed in a specific tissue or developmental stage. As a result, each gene may have several forms of

[1] See *http://alces.med.umn.edu/tables/hum-chr.html*.

```
>>INS1ECLAC                                                    (884 nt)
 initn: 472 init1: 472 opt: 641
 55.656% identity in 778 nt overlap
                              10        20        30        40
 isois1                GCATCGATATTTTTTCAGGTGATGCCTCTAATTAGTTGAATCTGATG
                                      :::::::: : :: :   :: :: :  ::
 INS1EC    CTGATAAGAGACACCGGCATACTCTGCGACGGTGATGCTGCCAACTTACTG-ATTTAGTG
              30        40        50        60        70         80

           50        60        70        80        90        100
 isois1    TATAATGCGGGCTTTTGAGGTTCTTTCATGGCCAGCGTTAACATTCATTGT-CCTCGTTG
           ::: ::: : : ::::::::: ::    ::::    :::   ::    ::: :::: :
 INS1EC    TATGATG-GTGTTTTTGAGGTGCTCCAGTGGCTTCTGTTTCTATCAGCTGTCCCTCCTGT
               90        100       110       120       130       140

           110         120       130       140       150       160
 isois1    TCAG--TCTGCACAGGTCTACCGCCATGGTCAGAACCCTAAAGGCCATGACAGATTTCGC
           ::::   :::    :::    ::  : ::  : : : :    :: ::: :  : : :: :
 INS1EC    TCAGCTACTGACGGGGTGGTGCGTAACGGCAAAAGCACCGCCGGACATCAGCGCTATCTC
               150       160       170       180       190       200

              170       180       190       200       210       220
 isois1    TGCCGTGACTGCCACCGCGTTTTTCAGCTCACTTACACTTATGAGGCCCGTAAGCCGGGC
           :::  : ::::::         :  :: ::    : ::::::    ::   : : :: ::
 INS1EC    TGCTCTCACTGCCGTAAAACATGGCAACTGCAGTTCACTTACACCGCTTCTCAACCCGGT
               210       220       230       240       250       260
```

Figure 2

Sequence alignment. A portion of a FASTA3 [11] comparison of the bacterial mobile DNA element IS1 from *Escherichia coli* with a distantly related element, IS1*v*ξ (isois1), from *Shigella dysenteriae*.

transcripts. Capturing these transcripts in the form of ESTs, and the subsequent reconstruction of the parent gene sequence, is thus at best inexact. However, by application of a set of assumptions, it may be possible to bound an estimate that reflects the "true reality."

EST analysis

A common way to assemble ESTs is by cluster analysis. The goal of such a project is the construction of a gene index in which ESTs and full-length transcripts are partitioned into index classes (or clusters) such that they are placed in the same index class if and only if they represent the same gene. Accurate gene indexing facilitates gene expression studies and reduces the cost of gene discovery through the assembly of ESTs derived from genes that have yet to be positionally cloned or obtained directly through genomic sequencing. Also, effective gene clustering serves as a starting point for the discovery of new gene expression variants such as alternative splicing forms. Torney et al. [1] have developed an algorithm known as d^2 that is used as the basis for a program we have developed that we call d^2_cluster. It is an agglomerative algorithm specifically developed for rapidly and accurately partitioning transcript databases into index classes by clustering ESTs and full-length sequences according to minimal linkage or "transitive closure" rules.

Projects related to EST clustering and assembly include UniGene [2] from the National Center for Biotechnology Information; the TIGR Gene Index [3–5] (*http://www.tigr.org/tdb/hgi/hgi.html*) from the Institute for Genomic Research; the Sequence Tag Alignment and Consensus Knowledgebase [6] (STACK; *http://ziggy.sanbi.ac.za/stack/stacksearch.htm*); the Merck/Washington University Gene Index [7]; and the GenExpress project [8]. All of these projects perform some type of cluster analysis in which sequence similarity is used to form the clusters. A summary of gene clustering project methods in the context of d^2_cluster has been published [9], as has a tutorial on the process [10] (*www.sanbi.ac.za*).

The d^2 algorithm

In 1989, Torney et al. presented an algorithm called d^2 [1] for comparing two gene sequences. Originally developed for quickly locating repetitive sequences in DNA, it has proven to have other uses as well. In this section we compare and contrast d^2 with sequence-searching methods. Most sequence-comparison algorithms are context-dependent; i.e., one can obtain a traditional sequence alignment (**Figure 2**). A set of letters from one sequence can be written over a set of letters from another sequence and lines drawn between related or identical letters. In other domains, this is called approximate string

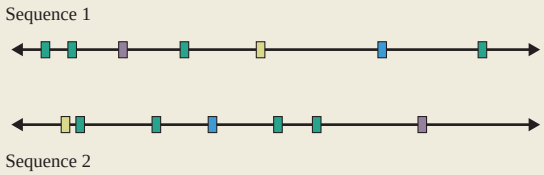

Figure 3

Idealized d^2 sequence comparison. The core idea is that the same words occur the same number of times in both sequences. The blocks of color denote similar sequences as detected by d^2. Note that there are four green blocks and one each of the purple, blue, and yellow blocks. Since the blocks are not in linear order between Sequence 1 and Sequence 2, no alignment such as that shown in Figure 2 is possible.

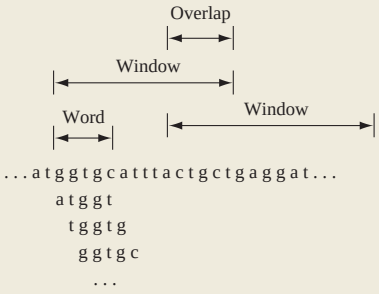

Figure 4

Graphic depiction of d^2 core parameters: word size, window size, and window overlap.

matching. In contrast, a context-independent, word-based method such as d^2 seeks to determine only whether the substrings (words) of a particular size occur the same number of times in both sequences, regardless of location (**Figure 3**). The d^2 algorithm is a member of a class of algorithms known as statistical distances.

FASTA vs. d^2

The earliest widely used sequence-searching program was probably FASTA [11], which gains its speed by breaking up the query and target sequences into overlapping pieces of defined sizes (one or two for amino acid sequences, one to six for nucleotide sequences). The lists are compared, and when sets of contiguous words are found longer than 15 nucleotides (in the case of a nucleotide search) or six amino acids (in the case of an amino acid search), they are assembled into an alignment and presented to the user (Figure 2). In contrast, d^2 counts word multiplicities: Do the same words occur the same number of times in both sequences? Locality of reference is lost in such methods, but sequences can be scrambled with respect to one another, or contain deletions and insertions, and still be judged similar by d^2. (Recovering locality of reference is discussed below.) The d^2 algorithm is a distance measure, so *smaller* scores represent better matching, in contrast to FASTA, where larger scores denote better matches (a similarity measure).

Parameters of the d^2 algorithm

The d^2 algorithm contains three main parameters: window length (WL), window overlap (WO), specified as a percentage of the WL, and word (or k-tuple) size (**Figure 4**). Locality is recovered by breaking up a sequence into a set of overlapping windows. Otherwise, the program can show that two sequences contain similar words, but not the locations of those words. This is the chief distinction between a sequence-alignment program (such as FASTA) and a sequence-comparison program (such as d^2). Typically, the word size is 8 for EST clustering, with a window length of 100 nucleotides and a window overlap of 20.

The algorithm

Equation (1) defines the d^2 measure,

$$d^2 = \sum_{n=1}^{u} \sum_{i=1}^{4^{ktup}} [m_D(w_i) - m_Q(w_i)]^2, \quad (1)$$

where n is the window number, u is the total number of windows in a sequence, $m_D(w_i)$ is the multiplicity of word i of length *ktup* in the database sequence, and $m_Q(w_i)$ is the multiplicity of word w_i in the query sequence. Note that d^2 is 0 when the windows are identical. With this as background, we now describe the d^2_cluster algorithm (first presented in Burke et al. [12]).

Description of the d^2_cluster method

The d^2_cluster method is agglomerative: Every sequence begins in its own cluster, and the final clustering is achieved through a series of mergers [13]. The d^2_cluster method can be described in terms of minimal linkage (sometimes called single linkage or "transitive closure" in the sequence analysis literature). The term "transitive closure" refers to the property that any two sequences with a given level of similarity will occupy the same cluster. Hence, **A** and **B** are in the same cluster even if they share no similarity when there exists a sequence **C** with enough similarity to both **A** and **B**. The criterion for joining clusters is the detection of two sequences, one from each cluster, that share a window of 100 bases (for most EST clustering; the variable is called *Window_Size*).

Those two sequences are joined only if they are at least 80% similar. This variable is referred to as the *Stringency*. To detect the overlap criterion we use the d^2 algorithm and set parameters and threshold values as described [1, 14, 15]. The initial and final states of the algorithm constitute a partition of the input sequences in which each sequence is in a cluster and no sequence appears in more than one cluster.

For ease of notation, let the following conventions hold:

1. We signify the d^2 distance between two sequences, say **A** and **B**, as d2(**A**,**B**).
2. Given two clusters, say clusters i and j, the operation *MERGE*(cluster i, cluster j), also denoted as *MERGE*(cluster i ← cluster j), means that all sequences in cluster j are assigned to cluster i.
3. The database to be clustered contains N sequences that are numbered 0 through (N−1); let sequence (i) be denoted Si or S(i).
4. The membership of sequence Si is denoted Ci.

The notation d2(**A**,**B**) is conveniently used, although d2(,) is a function not only of **A** and **B** but also of various parameters (as specified in [1, 14, 15]). The *MERGE* operation can be expressed in terms of step 4 above. For all sequences S_r such that $C_r = j$, C_r is reset to be $C_r = i$. C-style pseudocode for the *MERGE* operation is given in **Figure 5(a)**, and for step I, $1 < i < N$, in **Figure 5(b)**.

In **Table 1** we describe the progression of the d^2_cluster algorithm inductively in that we first detail what happens in the first two iterations (I1 and I2) and then describe how one performs iteration (i+1) given that iteration (i) has been completed. Technically speaking, it is sufficient to state only the first step and then to give the instructions for steps (i) to (i+1), but we detail the first two steps for clarity. The clustering is finished when *N* iterations are completed. Transitive closure is obtained because clusters are joined if they contain any sequences with sufficient identity.

```
(a)

MERGE( cluster Ci, cluster Cj) {
                For(r=0;r<N;r++)
                    if( Cr==Cj )
                            Cr=Ci
}.
```

```
(b)

STEP I {
        Select Si;
        For( j=(i+1);j<N;j++)
        {
                if( d2(Si,Sj)<THRESHOLD
                {
                        if(Ci < Cj)
                                MERGE( cluster Ci ← cluster Cj )
                        if(Ci > Cj)
                                MERGE( cluster Cj ← cluster Ci )
                        if(Ci==Cj)
                                (do nothing);
                }
        }
}
```

Figure 5

C-style pseudocode for (a) the merge operation and (b) step I ($1 < i < N$) of d2_cluster.

The d^2_cluster method as described above can be mapped to the minimal linkage algorithm commonly seen in statistics and engineering texts. Define a discrete distance, d0, on sequences to be

d0(A,B) = 0, if d2(A,B) < THRESHOLD

and

d0(A,B) = 1, if d2(A,B) >= THRESHOLD.

Linkage methods are usually presented in terms of distance matrices. Since there are initially *N*

Table 1 The d^2_cluster algorithm.

The initial state (I0): Each sequence is in its own cluster (i.e., Si is in cluster i or Ci = i).
The first iteration (I1): The first sequence in the database, S0, is selected as a query. For each sequence in Si(1⩽i<N), MERGE(cluster C0 ← cluster Ci) if d2(S0,Si)<THRESHOLD.
The second iteration (I2): The second sequence in the database (S1) is now selected as a query. Note that C1 = 1 unless sequence 1 was merged into cluster 0 during step I1. For all seqs, Si (2⩽i<N), *MERGE*(cluster C1 ← cluster Ci) if d2(S1,Si)<THRESHOLD.
The (k)th iteration I(k): Suppose we have completed (k−1) iterations. We select sequence Sk as a query. For all seqs, Si(k+1⩽i<N), if d2(Sk,Si)<THRESHOLD, then merge clusters Ck and Ci according to the following schedule: If Ci<Cj, then *MERGE*(cluster Ci ← cluster Cj); If Ci>Cj, then *MERGE*(cluster Cj ← cluster Ci); If Ci=Cj, then do nothing.

Table 2 Description of the d^2_cluster algorithm in terms of minimal linkage.

1. Initially all N clusters contain one sequence. D is an N by N matrix.
2. Search distance matrix for smallest distances between clusters (or, equivalently, take the first zero distance; search in any order).
3. When a zero distance is found at, say, row i column j ($i < j$), delete rows i and j and columns i and j. Replace them with a new row and column specifying the distance, d_0 (again 1 or 0 as defined above), from the merged cluster to other clusters.
4. Repeat steps 2 and 3 ($N - 1$) times until joining information exists for all clusters. Stop when there exist no additional zero entries in the distance matrix.

Table 3 Determination of the average number of alternative splices per cluster.

Stringency (%)	*Subclusters*	*Clusters*	*Ratio*	*Singletons*
80	82,657	67,499	1.2	149,510
85	82,113	66,654	1.2	141,345
90	82,907	67,410	1.2	145,263
99	81,858	70,852	1.2	234,436

sequences/clusters, let D = {d0(i,j)} be the N by N matrix of discrete distances. In agglomerative clustering, the dimensionality of D is reduced as sequences are clustered, and in minimal linkage the distance between two clusters **X** and **Y** is

d0(**X**,**Y**) = min{d0(Si,Sj): all Si in **X**, all Sj in **Y**}.

Table 2 describes the d^2_cluster algorithm in terms of minimal linkage. When the clustering is completed, the matrix **D** will have as many rows and columns as there are distinct clusters.

Screening expressed sequence tags

Before sequence can be input to the program, a variety of troublesome details must be considered. Biological sequence is not clean; it can contain a wide variety of contaminants that must be removed for the clustering to proceed optimally. These contaminants fall into several classes. First, there can be vector sequence, left over from the original cloning of the cDNAs. Vector can be readily identified because only a small number of sequence types are used as vector. There are more important challenges, however. Genomic sequence from all organisms contains repetitive sequences. These can be long or short, and may be highly variable in sequence. Independent databases of repetitive sequence are available [16]. When clustering ESTs, looking for novel genes, one will not wish to rediscover standard housekeeping genes again and again. These, too, will be screened out. In all cases, we use the Cross_match program[2] from P. Green of the University of Washington. Screened-out sequence is replaced by Xs, and we require at least 100 bases of non-X sequence in an EST to admit it to the clustering step.

Clustering

The sequences input to the d^2_cluster algorithm are compared to one another: $n(n - 1)/2$ comparisons. Large-scale clustering of 5 000 000 sequences can take about 72 hours on four SGI Origin 2000 processors for screening and clustering (approximately 1.25×10^{13} comparisons). After these comparisons are generated, clusters are computed; then each cluster is assembled using the PHRAP program[2] from P. Green of the University of Washington. The assembly step can take several weeks on four to eight Origin 2000 CPUs. Output comprises a number of clusters and singletons (single sequences that do not cluster with any other sequences; they are also called singleton clusters). Each cluster may have one or more subclusters, and a subcluster may be a singleton subcluster. A singleton subcluster is a single EST that is similar to other sequences in the cluster but does not assemble into a contiguous stretch of DNA with the other members of the cluster.

Clustering public ESTs

We can now turn to the results of clustering human ESTs. We used ESTs available in the public domain as of February 11, 2000. There were 1 373 183 ESTs in the input file. The DoubleTwist, Inc. Clustering and Alignment Tools, Version 3.5[3] were used to screen, cluster, and

[2] Unpublished; see *www.phrap.org*.

[3] *http://www.doubletwist.com*.

assemble the data on four processors of an eight-processor Silicon Graphics Origin 2000 with 3 GB of RAM. The standard screening files supplied by DoubleTwist were used. Standard input parameters were used except for the stringency of d^2 comparison, the variable called *set_d2_string*. We used values of 0.8, 0.85, 0.9, and 0.99, corresponding respectively to an identity of 80%, 85%, 90%, and 99%. These values caused the program to be increasingly stringent in its clustering. Sequences must have at least *set_d2_string* identity to be joined into a cluster, as described above. **Table 3** summarizes the results of these clusterings. At a stringency of 80%, there are 82657 subclusters (of any type) in 67499 clusters, or 1.2 transcripts per cluster. This result places a loose lower bound on the estimate of the number of genes for this stringency. There must be about this many *transcripts* in the dataset. If we assume that each cluster is a unique transcript, this stringency implies that there are about 67500 parent transcripts. For several reasons, however, this is underestimated. First, the public data do not reflect transcripts from all possible tissue types, disease types, and developmental states. Additionally, technical details of the generation of ESTs limit the number of ESTs that can be obtained from a particular tissue. Very rare transcripts (for example, those occurring less than ten times in a tissue) are not likely to be represented. Therefore, estimating the number of genes, transcripts, and proteins involves some guesswork as to the level of underrepresented genes and transcripts in the publicly available data. We do not attempt to estimate the number of genes included in this category. Thus, from the data given above, we estimate that there is an average of 1.2 transcripts per gene, giving a total of some 81000 different transcripts in the human genome. This means that there are at least 81000 different proteins before post-translational modification.

Another notable feature of the table is that as the stringency of clustering increases, the number of singletons also increases, except at 85% stringency (Table 3). This is what one would intuitively expect—if there are fewer clusters, there should be more singletons. When 85% similarity is required, there are fewer clusters, subclusters, and singletons. This apparent conundrum is resolved by noting that these clusters and subclusters contain more ESTs than at other stringencies. This situation will be investigated further.

Carrying a similar analysis through for 85%, 90%, and 99% clustering gives the results presented in **Table 4**. The estimate remains remarkably consistent. This is either a reflection of an artifact of the d^2_cluster method, or the data, or it reflects the underlying number of genes and transcripts in the human genome.

Table 4 Estimates of transcription units under varying levels of stringency.

Stringency (%)	*Clusters*	*Total transcription units (rounded)*
80	67,500	81,000
85	66,654	79,984
90	67,410	80,892
99	70,852	85,722

The number of human genes and proteins

On the basis of the data presented above, we believe that there are between 1.2 and 1.5 transcripts per transcription unit in the human genome. Our clustering suggests to us that there are about 70000 transcription units in the human genome. This implies that there are up to 84000 different proteins produced in a human cell before post-translational modification, such as the attachment of phosphate, adenylate, lipids, or sugar groups. This is obviously an extremely coarse estimate.

What are the sources of error in this estimate? First, we note that the d^2_cluster algorithm does not perform well with sequences less than 80% similar (by the nature of its context-independent measure). In a previous paper [9] we attempted to gauge absolute upper bounds for the error rates of the d^2_cluster algorithm's actual type I and type II errors. The method is described in detail in that paper. Those results indicate that the type II error is bounded above by 0.8%. The probability of not joining sequences that belong together (type I error bounds) is less than 0.4%. Thus, the sensitivity and selectivity of the d^2_cluster algorithm is at least 99%. If there is a problem with this estimate, it is likely to be in the underlying assumptions about the data. Another complicating factor is that similar exons exist in gene families, so it is very difficult to state that a cluster is a true transcript if a 100-base-pair exon of at least 80% identity was used in order to estimate accurately the true number of genes/clusters.

EST-based estimates of gene number have included counting the number of 5′-complete EST clusters [17], the number that are near CpG islands and are 5′-complete (Incyte Genomics). CpG islands are regions of DNA known to be near expressed genes that have a high content of the nucleotides G and C. The TIGR Human Gene Index[4] estimates 100000 genes, while Incyte Genomics[5] estimates 140000 genes derived from a different clustering approach using their proprietary data in combination with the public data. We have not included

[4] *http://www.tigr.org/tdb/hgi/index.html*.
[5] *http://www.incyte.com/company/news/1999/genes.shtml*; see also *http://www.incyte.com/company/news/webcast/slides/sld007.html*.

such analyses here. It is clear from other data available to us that the publicly available sequences do not fully reflect the full depth of the human genome. It is almost certain that as additional EST data become available, the EST-based estimate of the number of genes will change. The estimate of the number of transcripts per transcription unit will also increase.

A number of recent publications contain non-EST-based estimates of the number of human genes. One estimate, based upon gene density in completely sequenced chromosomes 21 and 22 and sampling theory, is that there are some 35000 genes in the human genome [18]. Crollius et al. [19] present a very different survey based on pufferfish sequences and estimate 28000–34000 genes. In a recent survey of gene number estimates, Aparicio [20] concluded that EST-based estimates are high, and sampling theory, computational gene modeling, and other approaches are likely to be more accurate.

We recognize that a good check of the reliability of EST clustering would be to compare clusters generated by this program (or any EST clustering program) against predicted proteins from chromosomes 21 and 22. These calculations are in progress. This is not an ideal check, because gene-modeling programs have their own biases. In particular, they do not recognize 5′-most and 3′-most exons well. Also, their overall accuracy is not clear. In particular, they cannot model genes that have not been identified. Nevertheless, comparing EST clustering results to sequenced chromosomes will provide useful insights into both types of programs.

It is reasonably clear that methods based on EST clustering appear to estimate some 70000–140000 transcription units. Methods based on other criteria offer considerably different estimates. The key assumption in our estimate is that a cluster corresponds to a transcription unit, which corresponds to the classical "gene" of human genetics. In the coming years, it will be fascinating to learn which assumptions and approaches turn out to reflect biology.

Acknowledgments

The authors would like to thank an anonymous reviewer for suggestions that improved the manuscript.

References

1. D. C. Torney, C. Burks, D. Davison, and K. M. Sirotkin, "Computation of d2: A Measure of Sequence Dissimilarity," *Computers and DNA, SFI Studies in the Sciences of Complexity*, Vol. VII, G. Bell and T. Marr, Eds., Addison-Wesley Publishing Co., Reading, MA, 1990, pp. 109–125.
2. M. S. Boguski and G. D. Schuler, "ESTablishing a Human Transcript Map," *Nature Genet.* **10,** 369–373 (1995).
3. M. D. Adams, A. R. Kerlavage, R. D. Flieschmann, R. A. Fuldner, C. J. Bult, N. H. Lee, E. F. Kirkness, K. G. Weinstock, J. D. Gocayne, O. White, G. Sutton, J. A. Blake, R. C. Brandon, M. W. Chiu, R. A. Clayton, R. T. Cline, M. D. Cotton, J. Earle-Hughes, L. D. Fine, L. M. FitzGerald, W. M. FitzHugh, J. L. Fritchman, N. S. M. Geoghagen, A. Glodek, C. L. Gnehm, and C. Venter, "Initial Assessment of Human Gene Diversity and Expression Patterns Based upon 83 Million Nucleotides of cDNA Sequence," *Nature* **377** (suppl.), 3–17 (1995).
4. G. Sutton, O. White, M. D. Adams, and A. R. Kerlavage, "TIGR Assembler: A New Tool for Assembling Large Shotgun Sequencing Projects," *Genome Sci. & Technol.* **1**, 9–18 (1995).
5. O. White and A. R. Kerlavage, "TDB: New Databases for Biological Discovery," *Meth. in Enzymol.* **206,** 27–41 (1996).
6. W. Hide, J. Burke, A. Christoffels, and R. Miller, "A Novel Approach Towards a Comprehensive Consensus Representation of the Expressed Human Genome," *Genome Informatics 1997*, Satoru Miyano and Toshihisa Takagi, Eds., Universal Academy Press, Inc., Tokyo, Japan; ISSN 0919-9454 (1997), pp. 187–196.
7. B. A. Eckman, J. S. Aaronson, J. A. Borkowski, W. J. Bailey, K. O. Elliston, A. R. Williamson, and R. A. Blevins, "The Merck Gene Index Browser: An Extensible Data Integration System for Gene Finding, Gene Characterization and EST Data Mining," *Bioinformatics* **14,** 2–13 (1998).
8. R. Houlgatte, R. Mariage-Samson, S. Duprat, A. Tesslier, S. Bentolila, B. Lamy, and C. Auffray, "The GenExpress Index: A Resource for Gene Discovery and the Genic Map of the Human Genome," *Genome Res.* **5,** 272–304 (1995).
9. J. Burke, D. Davison, and W. Hide, "d2_cluster: A Validated Method for Clustering EST and Full-Length cDNA Sequences," *Genome Res.* **9,** 1135–1142 (1998).
10. Alan Christoffels, Robert Miller, Andrey Ptitsyn, Chellappa Gopalakrishnan, Janet Kelso, and Winston Hide, "EST Clustering," *Proceedings of the 7th International Meeting for Intelligent Systems in Molecular Biology*, Heidelberg, Germany, 1999; *http://www.sanbi.ac.za/submissionl.PDF*.
11. W. R. Pearson, "Flexible Sequence Similarity Searching with the FASTA3 Program Package," *Meth. Mol. Biol.* **132,** 185–219 (2000).
12. J. P. Burke, H. Wang, W. Hide, and D. Davison, "Alternative Gene Form Discovery and Candidate Gene Selection from Gene Indexing Projects," *Genome Res.* **8,** 276–290 (1998).
13. R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, Third Edition, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1992.
14. W. Hide, J. Burke, and D. Davison, "Biological Evaluation of d^2, an Algorithm for High-Performance Sequence Comparison," *J. Comp. Biol.* **1,** 199–215 (1994).
15. T. J. Wu, J. P. Burke, and D. B. Davison, "A Measure of DNA Sequence Dissimilarity Based on Mahalanobis Distance Between Frequencies of Words," *Biometrics* **53,** 1431–1439 (1997).
16. J. Jurka, "Repeats in Genomic DNA: Mining and Meaning," *Curr. Opin. Struct. Biol.* **8,** 333–337 (1998).
17. F. Liang, I. Holt, G. Pertea, S. Karamycheva, S. L. Salzberg, and J. Quackenbush, "Gene Index Analysis of the Human Genome Estimates Approximately 120,000 Genes," *Nature Genet.* **25,** 239–240 (2000).
18. B. Ewing and P. Green, "Analysis of Expressed Sequence Tags Indicates 35,000 Human Genes," *Nature Genet.* **25,** 232–234 (2000).
19. H. R. Crollius, O. Jaillon, A. Bernot, C. Dasilva, L. Bouneau, C. Fischer, C. Fizames, P. Wincker, P. Brottier, F. Quétier, W. Saurin, and J. Weissenbach, "Estimate of Human Gene Number Provided by Genome-Wide

Analysis Using Tetraodon Nigroviridis DNA," *Nature Genet.* **25,** 235–238 (2000).
20. S. A. J. R. Aparicio, "How to Count . . . Human Genes," *Nature Genet.* **25,** 129–130 (2000).

Received October 25, 2000; accepted for publication January 2, 2000

Daniel B. Davison *Bioinformatics, Bristol-Myers Squibb, 5 Research Parkway, Wallingford, Connecticut 06492 (Daniel.Davison@bms.com).* Dr. Davison is the Associate Director of the Bioinformatics Department at Bristol-Myers Squibb. He received his B.S. degree in biology from Syracuse University in 1977, his master's degree in molecular biology from the State University of New York at Stony Brook in 1981, and his Ph.D. in genetics from SUNY Stony Brook in 1985. After postdoctoral training at the University of Houston and the Los Alamos National Laboratory, he returned to the University of Houston as an Assistant and later Associate Professor of Biology and Biochemistry, with a joint appointment in the Computer Science Department. In 1997 he moved to Bristol-Myers Squibb. Dr. Davison has been active in bioinformatics since 1979 and has numerous biological and computational publications.

John F. Burke *DoubleTwist, Inc., 2001 Broadway, Oakland, California 94612 (jburke@doubletwist.com).* Mr. Burke is Vice President of Bioinformatics at DoubleTwist, Inc. He received both his B.S. and his M.S. degrees in mathematics from the University of Houston, the latter in 1997. He worked at the former MasPar Computer Corporation until joining DoubleTwist. Mr. Burke has been active in biostatistics and mathematics since 1992.