Synthetic Gene Design with a Large Number of Hidden Stop Codons

Vinhthuy Phan                   Sudip Saha                      Ashutosh Pandey              Wong Tit-Yee
Dept. of Computer Science     Dept. of Computer Science  The Bioinformatics Program  Dept. of Biology
The Bioinformatics Program  The University of Memphis The University of Memphis    The University of Memphis
The University of Memphis
ssaha@memphis.edu                 akpandey@memphis.edu            tywong@memphis.edu
vphan@memphis.edu

Abstract

Hidden stop codons are nucleotide triples TAA, TAG, and TGA that appear in the second and third reading frames of a protein coding gene. Recent studies reported biological evidence suggesting that hidden stop codons are important in preventing misread of mRNA, which is often detrimental to the cell. We study the problem of designing protein-encoding genes with large number of hidden stop codons under biological constraints including GC content and codon usage of individual organism. In simpler models, we obtained provably optimal results. In more complex models, the designed genes have many more hidden stop codons than wild-type genes do, as observed in an experiment with 8 genomes with a wide range of GC content and codon usage.

1. Introduction

A protein coding gene often begins with a “start codon” - ATG followed by three nucleotides at a time, each representing an amino acid, and eventually ends with a “stop codon” – TAA, TAG, or TGA. Since each codon is composed of three nucleotides, the number of nucleotides in all genes is multiples of three nucleotides. Obviously, the protein sequence is based on the nucleotides on the first reading frame of the gene. Topologically, however, one can also recognized the nucleotide sequences on the second, and third reading frames of a gene, which begins at the second and third nucleotide, respectively, from the start codon. Although nonfunctional, many interesting features have been found in the 2nd and 3rd reading frames. For example, there are many TAA, TAG and TGA triplets in the second and third reading frames. About half of the genes in the Escherichia coli genome, their start codons ATG are followed by a codon that begins with an “A”, forming an ATGA tetramer (Wong, unpublished data). The last three nucleotides (TGA) of this tetramer is a potential stop-codon-forming sequence. These stop-codon-like triples are called “hidden stop codons”.

There are strong evidences to show that these hidden stop codons are important for the well being of the cell. These hidden stop codons in the gene would terminate protein synthesis had the translation been misdirected by the ribosome, (through mechanisms such as codon slippage, or the presence of certain antibiotics), or gene alternation due to deletion or insertion resulting in frameshift mutation. The “ambush” hypothesis [1] suggested that hidden stop codons prevent the event of off-frame reading, which occurs about once every 30,000 codons [2]. Seligman and Pollock [1] found in 38 organisms a strong tendency for using codons being more potential to form hidden stop codons. Recently, Itzkovitz and Alon [3] found evidence for preference of having more hidden stop codons in the selection of the genetic code itself. They showed that there are more hidden stop codons using the standard genetic code than using 99.3% of all random codes and furthermore the possibility of having more hidden stop codons tied closely with the possibility of including more signals in the genome.

The problem of forming hidden stop codons in a gene is not straightforward. The genetic codons are said to be redundant, meaning that, except for the amino acids tryptophan and methionine, all other amino acids are coded by two or more genetic codes. Codons in a living organism are, however, not utilized uniformly. Why an organism prefers a particular set of codons for its proteins remains a fascinating question and is likely to be related to natural selection of individual’s environmental habitat [4]. As codon usage bias dictates the choice of codons for amino acids, it effects significantly the sequence information of genes and hence hidden stop codons.

The fidelity of protein synthesis is often the problem of designing a synthetic gene that encodes for a given protein to be expressed in a host organism. The design of a synthetic gene must satisfy certain constraints, including having a specific GC content [5], codon usage [6-8], and sequence comparability[9] to the host organism [10]. Computational techniques employed by these approaches include genetic algorithms [11] or hidden Markov model [9]. Despite
these advances, expression of synthetic gene is still problematic. Expression failure of synthetic genes might be partially due to frameshift mutation or the codon slippage [12]. In case of codon slippage, the more the hidden stop codons there are, the sooner the slipped gene would be terminated, and thus less harmful products would accumulate inside the host. Such early intervention by hidden stop codons would save energy and other resources for the cell.

In this paper, we introduce algorithms that design protein-encoding genes with large numbers of hidden stop codons under a model with no additional constraints, as well as complex models with specific GC content or codon usage requirements. Without any constraints, a maximal number of hidden stop codons is always achieved. With a required GC content, the design is also optimal in many cases. With a required codon usage, designed genes have many more hidden stop codons than wild-type genes (i.e. those appearing in nature). The algorithms were tested on 8 genomes, where we “redesigned” all genes and found that the synthesized genes had many more hidden stop codons.

Table 1: Standard Genetic Code

<table>
<thead>
<tr>
<th>Amino Acid</th>
<th>Codons</th>
</tr>
</thead>
<tbody>
<tr>
<td>A (Alanine)</td>
<td>GCT, GCC, GCA, GCG</td>
</tr>
<tr>
<td>R (Arginine)</td>
<td>CGT, CGC, CGA, CGG, AGA, AGG</td>
</tr>
<tr>
<td>N (Asparagine)</td>
<td>AAC, AAC</td>
</tr>
<tr>
<td>D (Aspartic Acid)</td>
<td>GAT, GAC</td>
</tr>
<tr>
<td>C (Cysteine)</td>
<td>AAT, AAC</td>
</tr>
<tr>
<td>Q (Glutamine)</td>
<td>CAA, CAG</td>
</tr>
<tr>
<td>E (Glutamic Acid)</td>
<td>GAA, GAG</td>
</tr>
<tr>
<td>G (Glycine)</td>
<td>GTT, GGC, GGA, GGG</td>
</tr>
<tr>
<td>H (Histidine)</td>
<td>CAT, CAC</td>
</tr>
<tr>
<td>I (Isoleucine)</td>
<td>ATT, ATC, ATA</td>
</tr>
<tr>
<td>L (Leucine)</td>
<td>TTG, TGG, CTT, CTC, CTG</td>
</tr>
<tr>
<td>K (Lysine)</td>
<td>AAA, AAG</td>
</tr>
<tr>
<td>M (Methionine)</td>
<td>ATG</td>
</tr>
<tr>
<td>F (Phenylalanine)</td>
<td>TTT, TTC</td>
</tr>
<tr>
<td>P (Proline)</td>
<td>CCT, CCC, CCA, CGG</td>
</tr>
<tr>
<td>S (Serine)</td>
<td>TCT, TCC, TCA, TCG, AGT, AGC</td>
</tr>
<tr>
<td>T (Threonine)</td>
<td>ACT, ACC, ACA, ACG</td>
</tr>
<tr>
<td>W (Tryptophan)</td>
<td>TGG</td>
</tr>
<tr>
<td>Y (Tyrosine)</td>
<td>TAT, TAC</td>
</tr>
<tr>
<td>V (Valine)</td>
<td>GTT, GTC, GTA, GTG</td>
</tr>
<tr>
<td>STOP CODONS</td>
<td>TAG, TGA, TAA</td>
</tr>
</tbody>
</table>

2. Maximization of Hidden Stop Codons in Back Translation

A hidden stop codon is formed from the juxtaposition of two codons such that they form a stop codon triplet in between. For example, juxtaposition of two codons CTG (Leucine) and ATG (Isoleucine): CTGATT would form a hidden stop codon TGA in the second reading frame. We will use the standard genetic code (Table 1), but the algorithms presented in this paper can work with any alternative genetic code.

The HSC problem: Given $P = A_1A_2...A_n$ be the sequence of $n$ amino acids, design a DNA sequence that encodes $P$ with the maximum number of hidden stop codons possible.

For example, given the following amino acid sequence MSDSKED, an encoding DNA sequence with the maximum number of hidden stop codons is ATGAGTGATAGTAAAGAAGAC; there are 4 hidden stop codons.

While Tryptophan and methionine are the only amino acids with 1 codon, the rest have between two and six codons. This means that in general there are exponentially many DNA sequences that encode a given amino acid sequence. Fortunately, we can solve this problem in linear time using dynamic programming. First, for any amino acid $A_i$, we arbitrarily order the codons that encode $A_i$. When we refer to the $j^{th}$ codon for $A_i$, we refer to the codon in this pre-determined order.

Define $H(i,j)$ to be the maximum number of hidden stop codons in all of DNA sequences that encode $A_1A_2...A_i$, where $A_i$ is coded by its $j^{th}$ codon, and additionally define $I_{kj}$ to be 1 if codon $k$ of $A_{i-1}$ followed by codon $j$ of $A_i$ has a hidden stop codon; and 0 otherwise. The optimal structure of $H$ can be defined recursively as follows:

- $H(1,j) = 0$ for all $j$.
- $H(i,j) = \max \{H(i-1,k) + I_{kj}\}$, where $k$ represents each codon that encodes $A_{i-1}$.

Note that $H(1,j) = 0$ for any $j$ because the first amino acid has no hidden stop codon. Assuming that $H(i-1,k)$ is correctly computed for $1 \leq k \leq N$ (N is the number of codons of $A_{i-1}$), i.e. $H(i-1,k)$ is the maximum number of hidden stop codons in all of DNA sequences that encode $A_1A_2...A_{i-1}$ using the $k^{th}$ codon for $A_{i-1}$ for all $1 \leq k \leq N$. Then, $H(i,j)$ as defined by the recurrence above must be the maximum number of hidden stop codons because the choice of codon for $A_i$ does not depend on other choices of previous amino acids other than the choice of codons for $A_{i-1}$. Thus, by inductive reasoning, $H(i,j)$ is correctly computed based on the fact that $H(1,j)$’s are correctly computed for all $j$ and that $H(1,k)$ is correctly computed for $1 \leq k \leq N$, where $N$ is the number of codons that encode $A_{i-1}$.

The algorithm for computing $H(\cdot,\cdot)$ comes directly from the recurrent relations. Further, it can be computed in linear time in the number of amino acids, because each entry $H(i,j)$ is computed in constant time (N$\leq$6, i.e. each amino acid has at most 6 codons). In the end, we return the maximum of $H(n,j)$, $1 \leq j \leq N$ for all N codons encoding $A_n$. This is the maximum number of hidden stop codons in $A_1A_2...A_n$. 


Assuming $H(\cdot,\cdot)$ is already computed, the actual DNA sequence with the maximum number of hidden stop codons can be computed by tracing back maximum values at each step; see Algorithm 1.

Algorithm 1: HSC($A_1A_2\ldots A_n$)
1. For each amino acid, order the codons that encode it in an arbitrary manner
2. For $j=1$ to $n$: $H[1][j] = 0$
3. For $i=2$ to $n$:
   4. For each codon $j$ that encodes $A_i$: $H[i][j] = \max\{ H[i-1][k] + I_{kj} \}$ for all codons $k$ that encode $A_{i-1}$
5. Return $H[\cdot][\cdot]$

Algorithm 1: TraceBack($H[\cdot][\cdot]$)
1. $m = \max\{ H[n][k] \}$ for all codons $k$ that encode $A_n$
2. Seq[n] = $c = j$ such that $H[n][j] = m$
3. For $i=n$ to $1$:
   4. For each codon $j$ that encodes $A_i$
      5. $\text{max} = 0$
      6. If $H[i][j] + I_{jc} = m$ and $\text{GC content of } j > \text{max}$
         7. $S_{max}[i] = c = j$
         8. $\text{max} = \text{GC content of } j$
9. $m = m - I_{jc}$ such that $j = S_{max}[i]$
10. Return $S_{max}[\cdot]$

3. Maximization of Hidden Stop Codons with Respect to a Desirable GC-Content

GC content of a DNA sequence is the total number of G and C nucleotides in it. This is an important property of genomes and many organisms have distinct differences in this respect. Designed Gene should have favourable GC content in order to avoid formation of undesirable mRNA Secondary structure [13]. Hence, GC content plays an important part of gene design and back translation [5]. We extend the algorithm introduced Section 2 to design DNA sequences with a maximum possible number of hidden stop codons with a desirable GC-content for a wide range of cases. The problem can be formally defined as follows.

The HSC problem with GC content constraint:
Given a GC content $X$ and an amino acid sequence $P = A_1A_2\ldots A_n$, design a DNA sequence that has GC content equal to $X$ and that encodes $P$ with the maximum number of hidden stop codons possible.

This problem has two objectives that can not always be optimized simultaneously. For example, when $X=2$ and $P = NN$, i.e. two consecutive Asparagines, both objectives, i.e. GC-content being equal to 2 and getting maximum number of hidden stop codons cannot both be achieved. Because there are two codons AAT and AAC which encode for N, the only DNA sequence with GC content of 2 is AACAC. But this sequence has no hidden stop codon. On the other hand, the sequence AATAAC has 1 hidden stop codon.

Let $S$ be the set of DNA sequences that encode $P$ such that all sequences in $S$ have the maximum number of hidden stop codons. The algorithm that computes $H(\cdot,\cdot)$ in the previous section only calculates one sequence in $S$. The set $S$, however, has many sequences because each amino acid that does not contribute to a hidden stop codon can adopt many possibilities. Amino acids, which contribute to a hidden stop codon, can also be replaced with alternatives in some cases without compromising that HSC. Define $S_{max}$ and $S_{min}$ to be two sequences in $S$ with the highest and lowest GC content, respectively. If the required GC content falls within this range of GC contents, then both the objectives can be achieved. First, we will show how to construct $S_{max}$ and $S_{min}$ using an extended definition of $H(\cdot,\cdot)$. Second, we will show how to construct any DNA sequence that has the maximum number of hidden stop codons possible with GC content in the range between that of $S_{max}$ and $S_{min}$. Algorithm 2 constructs $S_{max}$. It is very similar to Algorithm 1 in constructing a DNA sequence with the maximum number of hidden stop codons, but additionally keeps track of codons that yield maximum GC content. $S_{min}$ can be constructed similarly.

Algorithm 2: TraceBack2($H[\cdot][\cdot]$)
1. $m = \max\{ H[n][k] \}$ for all codons $k$ that encode $A_n$
2. Seq[n] = $c = j$ such that $H[n][j] = m$
3. For $i=n$ to $1$:
   4. For each codon $j$ that encodes $A_i$
      5. $\text{max} = 0$
      6. If $H[i][j] + I_{jc} = m$ and GC content of $j > \text{max}$
         7. $S_{max}[i] = c = j$
         8. $\text{max} = \text{GC content of } j$
9. $m = m - I_{jc}$ such that $j = S_{max}[i]$
10. Return $S_{max}[\cdot]$

Now, given a GC content $X$ (between the GC content of $S_{min}$ and of $S_{max}$) and $P = A_1A_2\ldots A_n$ be the amino acid sequence of amino acids, we will construct a DNA sequence by combining both $S_{min}$ and of $S_{max}$ to achieve a GC content very close to $X$.

Proposition: Suppose $S_{max} = a_1\ldots a_n$ and $S_{min} = b_1\ldots b_n$ where both codons $a_i$ and $b_i$ encode the amino acid $A_i$ for $1 \leq i \leq n$. There always exists a $j$ such that the GC content of $a_1\ldots a_j b_{j+1}\ldots b_n$ differs from $X$ by at most $1$.

Proof: For $1 \leq k \leq n-1$, let $x_k$ be the GC content of $a_1\ldots a_k$ and $y_k$ to be the GC content of $b_{k+1}\ldots b_n$. The GC content of $a_1\ldots a_kb_{k+1}\ldots b_n$ is $x_k + y_k$. Similarly, the GC content of $a_1\ldots a_kb_{k+1}\ldots b_n$ is $x_{k+1} + y_{k+1}$. Observe

143
that \( x_{k+1} + y_{k+1} = (x_k + u) + (y_k - v) \), where \( u \) is the GC content of the codon \( a_{k+1} \) and \( v \) is the GC content of the codon \( b_{k+1} \). Due to the construction of \( S_{\text{max}} \) and \( S_{\text{min}} \), we have \( u \geq v \). Further, \( u - v \leq 3 \) because each codon has at most 3 G’s or C’s. This means that as \( k \) varies from 1 to \( n-1 \), the GC content \( x_k + y_k \) is monotonically increasing and further each increment in GC content is at most 3. Thus, given a \( X \) between the GC content of \( S_{\text{min}} \) and of \( S_{\text{max}} \), we can find a \( j \) between 1 and \( n \) such that \( a_1 b_2 \cdots b_n \) differs from \( X \) by at most 1. This is done by increasing \( k \) from 1 to \( n \) until it is closest to \( X \). Further, this construction is done in linear time.

We have shown how to construct a DNA sequence that encodes any amino acid sequence with the maximum number of hidden stop codons, given that the specified GC content is within those of \( S_{\text{min}} \) and \( S_{\text{max}} \). The achieved GC content is exactly the same or differs by 1 from the specified GC content. For long sequences, this difference is negligible. What if the specified GC content, \( X \), is outside the range of \( S_{\text{min}} \) and of \( S_{\text{max}} \)? In such cases, we can “fix” the GC content by using unfavorable codons in some cases and thereby reducing the hidden stop codons to ensure the desirable GC content can be achieved. Still, this number is always higher than the count of hidden stop codons in wild-type gene as shown in Section 5.

4. Maximization of Hidden Stop Codons with a Desirable Codon Bias

Organisms generally follow a distinctive codon usage for each amino acid. The six codons of Leucine, for instance, are not utilized uniformly. Which codon of which amino acid is selected more often depends not only on the organism and but also on the specific gene of the organism. It was observed that highly expressed genes often contained preferred codons while rarely expressed genes often employed more non-preferred codons for protein synthesis [14]. A distribution of codons implicitly affects the GC content of genes and genomes. As such, we can think of codon usage as a more general property. In the design of gene, during protein back-translation, codon usage is an important factor to consider in many cases [7, 15] due to the fact that codon usage is tightly related to the expression level of genes [16]. The codon usage table for an organism specifies the percentages of codons for an amino acid used in the genome. We translate this into the design of gene with maximum number of hidden stop codons as follows.

The HSC problem with Codon Usage: Given an amino acid sequence \( P = A_1 A_2 \cdots A_n \) and a codon usage vector \( C = (c_1, \ldots, c_{64}) \), design a DNA sequence with the maximum number of hidden stop codons such that the number of occurrences of codon \( i \) is \( c_i \).

The dual objectives are difficult to achieve. We resort to designing a gene that fits the codon usage while having as many hidden stop codons as possible. First, a DNA sequence is constructed with the maximum number of hidden stop codons without any restriction of codon usage. Then, it is “fitted” to the codon usage by destroying as few hidden stop codons as possible. This algorithm can be outlined as follows:

1. Construct a DNA sequence \( S \) with the maximum number of hidden stop codons using Algorithm 1.
2. Let \( A = (a_1, \ldots, a_{64}) \) be the codon usage of \( S \).
3. \( D = (d_1, \ldots, d_{64}) = C - A = (c_1-a_1, \ldots, c_{64}-a_{64}) \).
   Note that if \( c_1,\ldots,c_i \) represent the codon usage of a particular amino acid, then \( d_1+\ldots+d_i = 0 \).
4. Examine each codon \( j \) in \( S \) that is not contributing to a hidden stop codon. If codon \( j \) is overused, i.e. \( d_j < 0 \), replace it with an underused codon \( k \) of the same amino acid, i.e. \( d_k > 0 \). Update \( d_j \) and \( d_k \).
5. If there are still overused or underused codons, repeat step 4 on codons that contribute to hidden stop codons.

5. Experimental Results and Discussions

We selected 8 organisms from NCBI (Table 2) to study distribution of hidden stop codons and determine the effectiveness of the proposed algorithms. These organisms were selected based on their popularity, relatively small genomes, and diverse GC contents. We “redesigned” all protein-encoding genes of each genome and compared their percentage of hidden stop codons (over the total number of codons in a gene) to those of wild-type genes (i.e. genes reported by NCBI) and of random genes. There are 6 different designs:

1. “Optimal” is the design specified by Algorithm 1, which produces protein-encoding genes without any constraints of GC content or codon usage.
2. “Max wrt GC” is the design specified by Algorithm 2, which attempts to maximize hidden stop codons while respecting the GC content of the wild-type gene under consideration.
3. “Max wrt codon usage” is the design that attempts to maximize hidden stop codons while respecting the codon usage of the wild-type gene under consideration. This is described in Section 4.
4. “Random wrt codon usage” is a design that samples uniformly random codons for each amino acid from a pool that has the same codon usage as the wild-type gene under consideration.
5. “Random” is a purely random design that samples genes from the pool of genes encoding the given protein sequence.
6. “Wild-type” is the actual gene that encodes the given protein. Information about these genomes is obtained from NCBI; see Table 2.
### Table 2. Studied Organisms

<table>
<thead>
<tr>
<th>Organism/Accession #</th>
<th>Genome Size</th>
<th># of Genes</th>
<th># of Proteins</th>
<th>GC %</th>
</tr>
</thead>
<tbody>
<tr>
<td>B. afzelii NC 008277</td>
<td>905,394</td>
<td>894</td>
<td>855</td>
<td>27.67</td>
</tr>
<tr>
<td>Rickettsia typhi NC 006142</td>
<td>1,111,496</td>
<td>919</td>
<td>838</td>
<td>29.48</td>
</tr>
<tr>
<td>B. thuringiensis NC 008600</td>
<td>5,257,091</td>
<td>4883</td>
<td>4736</td>
<td>35.27</td>
</tr>
<tr>
<td>E. coli NC 008253</td>
<td>4,938,920</td>
<td>4780</td>
<td>4620</td>
<td>50.22</td>
</tr>
<tr>
<td>A. tumefaciens NC 003063</td>
<td>2,075,577</td>
<td>1884</td>
<td>1851</td>
<td>59.11</td>
</tr>
<tr>
<td>R. leguminosarum NC 008380</td>
<td>5,057,142</td>
<td>4800</td>
<td>4693</td>
<td>61.17</td>
</tr>
<tr>
<td>M. tuberculosis NC 000962</td>
<td>4,411,532</td>
<td>4048</td>
<td>3989</td>
<td>65.13</td>
</tr>
<tr>
<td>T. thermophilus NC 005835</td>
<td>1,894,877</td>
<td>2035</td>
<td>1982</td>
<td>69.05</td>
</tr>
</tbody>
</table>

Figure 1 shows the average percentage of hidden stop codons over all protein-coding genes in 8 studied organisms. Several observations can be made. First, GC content greatly affects the percentage (and number) of hidden stop codons. Organisms with low GC content have more hidden stop codons; this is not surprising because the GC content of stop codons (TAG, TAA, TGA) is inherently TA-rich. This trend is universal and is observed in wild-type, random designs as well as in optimal designs. So we think Figure 1 paints a general picture of gene design with maximal hidden stop codons.

Second, the optimal design without any constraint produces genes with a strong average of 33% hidden stop codons (ranging between 30-38% over the selected organisms). This means that protein synthesis will terminated in case the gene sequence is misread in about 3 codons from the place of error.

Third, if we need to respect the GC content of each wild-type gene (Algorithm 2), the result is still encouraging and in fact is still essentially optimal for GC content up to about 50%, (see “optimal” and “Max wrt GC” in Figure 1). This is argued theoretically in Section 3. Beyond 50%, the percentage of hidden stop codons drop significantly, and to be inline GC content, the design also suffers. Similar conclusion can be reached for the design that respects codon usage (Section 4). Nevertheless, these designs still have the constant positive factor of having more hidden stop codons than wild-type genes, even if the dual objectives of maximizing the number of hidden stop codons and achieving a specified GC content (or codon usage) is not always achievable.

Figure 1 also shows an interesting relationship between codon distribution and the percentage of hidden stop codons. The shapes of the two curves representing the uniformly random and theoretically optimal designs (without GC content and codon constraints) are very similar. Similarly, if the designs must respect codon usage (“random wrt codon usage”, and “Max wrt codon usage”), then the shapes of the curves are also similar. One implication is that our design achieves a constant factor increase in the number of hidden stop codons from a random design.

![Figure 1](image1.png)

**Figure 1.** Each dot represents an organism. The x-axis shows the GC content of each organism between 27.67 and 69.05%. The y-axis shows the percentage of hidden stop codons out of the total number of codons. The percentages of hidden stop codon are compared for 3 designs of genes (“optimal”, “Max wrt GC”, “Max wrt codon usage”), 2 random genes, and wild-type genes.

![Figure 2](image2.png)

**Figure 2.** Difference in positions of the first 10 hidden stop codons between wild-type and designed genes respecting codon usage. The difference is averaged over all genes of each organism.

Last but not least, we note that the positions – not just percentage – of hidden stop codons are important in preventing frameshift mutations. The cost of off-frame translation is expectedly higher when frameshifts occur closer to the beginning of a gene [1]. We found that our design of genes containing high number of hidden stop codons with codon usage constraint has a strong positional advantage in comparison to wild-type genes. This was
accomplished by comparing the position of the first 10 codons that contribute to hidden stop codons in both wild-type genes and genes designed with a maximum number of hidden stop codons while respecting the codon usage (Section 4). Specifically, we measured the difference in position of the 1st (2nd, 3rd, … and 10th respectively) hidden stop codon of the wild-type gene and the designed gene. Figure 2 shows that average positions of the kth hidden stop codon (1 \leq k \leq 10) in the designed genes always occur significantly before those of wild-type genes in all 8 organisms. This shows that the effect of frame-shift errors near the start of transcription is specifically less dangerous in the designed genes than in wild-type genes.

6. Conclusion
Several algorithms were presented to aim at designing genes with a large number of hidden stop codons. These algorithms can be used together with other gene designs strategies to promote early termination of frame-shifted translation and thus suppress the harmful effects produced as a result of misread genes. In this work, we also discovered interesting hints on the biology/evolutionary influence of hidden stop codons. For instance, Figure 1 suggested that there was no difference in the hidden stop codon percentage between a wild-type gene and a random gene drawn from the pool of all genes with the same codon usage. As it is known that codon usage is selected for [4], this observation suggests that the number of hidden stop codons in wild-type genes is a result of that particular codon usage being selected for.

Acknowledgement
We thank the Bioinformatics Program at the University of Memphis for partially supporting this research.

References: