

How optimal is the genetic code?

S. Schönauer and P. Clote
Universität München*

Introduction

The genetic code is well-known to be *fault tolerant*, in the sense that transcription errors in the third codon position frequently do not influence the amino acid expressed, while errors in other codon positions often lead to amino acids having similar chemical properties. Several articles in the recent past ([?, ?, ?] etc.) have studied the question of *optimality* of the genetic code.

In [?] Di Giulio estimated that the natural code has achieved 68% minimization of polarity distance, by comparing the natural code with random *block respecting* codes (those codes obtained by relabeling the 20 amino acids in the natural table by a permutation thereof). When considering single base changes in the codons, let $N_{i,j}$ be the number of times the i -th amino acid changes into the j -th amino acid, and X_i be the polarity index [?] of the i -th amino acid. The *percent minimization* is defined by

$$\frac{\Delta_{mean} - \Delta_{code}}{\Delta_{mean} - \delta_{low}}$$

where

$$\Delta^2 = \frac{\sum_{i,j} (X_i - X_j)^2 N_{i,j}}{\sum_{i,j} N_{i,j}},$$

Δ_{mean} is the average Δ value, obtained by averaging over many random block respecting codes, and Δ_{low} is an approximation of the lowest possible Δ value obtained using the method of Lagrange multipliers to solve a constrained minimization problem.

Again restricting attention only to block respecting codes, in [?] Haig and Hurst considered to what extent the natural code has been optimized with respect to fault tolerance concerning (a) polar requirement,¹ (b) hydrophobicity, (c) molecular volume and (d) isoelectric point. By measuring the values MS_1 , MS_2 ,

MS_3 , MS_0 for the mean squared change in an attribute's value (eg. polar requirement, hydrophobicity, etc.) for all single-base substitutions in first, second, third, resp. *all* codon positions for the natural and random block respecting codes, the authors concluded that "single-base substitutions are strongly conservative with respect to changes in polar requirement and hydrophobicity in the first and third codon positions, but much less so in the second codon position." Moreover, the polar requirement mean square difference MS_0 for the natural code was determined to be 5.194, while only 2 out of 10,000 random codes were found to be more conservative with respect to polar requirement (MS_0 values of 5.167 and 5.189). Polar requirement MS_1 , MS_2 and MS_3 values for the natural code were determined to be 4.88, 10.56 and 0.14, which reflects the fact that error tolerance is highest for transcription errors in the third codon (i.e. on average there is greatest conservation of polar requirement for single-base substitutions in the third codon position).

In contrast to the block respecting codes of [?, ?], Goldman [?] considered more general *shuffled codon* codes, which maintain the same number of codons per amino acid as in the natural code, but do not require the block structure of the natural code table. While there are $20! = 2432902008176640000 > 2.43 \times 10^{18}$ block respecting codes, there are $\frac{64!}{(2!)^9(3!)^2(4!)^8(6!)^3} > 10^{65}$ many shuffled codon codes. Goldman computed the mean square difference MS_0 over all single-base substitutions (for amino acid, non-stop codons) for artificial codes obtained by the *record-to-record travel* algorithm [?] (a heuristic for optimization). The most conservative code found by in [?] had polar requirement value $MS_0 = 4.005$, and for this code more uniformly spread MS_1 , MS_2 , MS_3 values of 3.06, 3.67, and 5.28.

*Research supported by the Volkswagen Stiftung.

¹Polar requirement, as measured by C. Woese et al. [?], is taken to be synonymous with hydrophobicity.

Methods

In this paper, we consider optimality of the natural code in the much larger space of *general* codes, which are taken to be surjective maps from $\{A, C, G, T\}^3$ onto the 20 amino acids plus the stop signal. Since a general code is identified with an onto mapping $c : 64 \rightarrow 21$, it follows that there are $21! \cdot S(64, 21)$ many codes, where $S(n, m)$ is a Stirling number of the second kind. Since $S(64, 21) = 2.95572845518811 \times 10^{64}$, it follows that there are more than 1.51×10^{84} general codes. We define the *fault tolerance* $FT(c)$ of code c is defined to be $\sum_{xyz \in \{A, C, G, T\}^3} [(1) + (2) + (3)]$, where

$$\sum_{x' \in \{A, C, G, T\} - \{x\}} WAC(c(xyz), c(x'yz)) \quad (1)$$

$$\sum_{y' \in \{A, C, G, T\} - \{y\}} WAC(c(xyz), c(xy'z)) \quad (2)$$

$$\sum_{z' \in \{A, C, G, T\} - \{z\}} WAC(c(xyz), c(xyz')). \quad (3)$$

Here, $WAC(A, B)$ is the similarity between amino acids A and B , as given in [?], and $c(xyz)$ is the amino acid encoded by codon xyz using code c .² Note that we apply the WAC matrix, rather than the better known PAM250 matrix [?], since the latter measures substitution frequency between amino acids, as determined in protein families, and so *assumes* the natural genetic code. To avoid this circularity, we applied the WAC matrix, created by measuring the physico-chemical properties in radial shells up to 10Å centered around a given amino acid, thus constituting a description of the “micro-environment” of the amino acid. Define fault tolerance in the first position $FT_1(c)$ of code c by $\sum_{xyz \in \{A, C, G, T\}^3} [(1)]$, and similarly for $FT_2(c)$ and $FT_3(c)$.³ The WAC similarity matrix has integer entries ranging between -5 and 4 , where $WAC(a, b) = 4$ when amino acids a, b are identical, and -5 for very dissimilar amino acids.

Using this, we apply a Monte-Carlo algorithm MC with simulated annealing [?] to try to produce an optimal code from an arbitrary initial code. One step of our MC algorithm performs a random codon re-assignment in code c to another amino acid or stop signal, while ensuring surjectivity of the new code c' ; if $FT(c') > FT(c)$ or if $FT(c') \leq FT(c)$ and the Metropolis criterion is satisfied, then $c = c'$. MC executes N iterations before temperature is lowered.

²One could multiply by the normalizing factor of $1/9 \cdot 64$ to obtain an expected value, but we do not do so.

³One could multiply by the normalizing factor of $1/3 \cdot 64$, but we do not do so.

Results

The fault tolerance $FT(c_n)$ of the natural code is 236. Our MC was run for 3 different values N of iterations per temperature step: 500, 5000 and 5 million.

For $N = 500$, the average FT was -516.16 with standard deviation σ of 94.88, while average FT for optimized codes was -111.64 with σ of 25.13. The table of average number of codons per amino acid for initial versus optimized codes is given below. It is striking that methionine has an average of 7 codons in the optimized codes.

For $N = 5000$, random initial codes had FT value ranging between -700 to -400 , and when optimized using MC , the final FT values ranged from -80 to -10 .

For $N = 5$ million, the optimized code had fault tolerance of 2 (2 hours computation time).

A comparison between the natural code and the best artificial code shows that the latter depends less on third base redundancy,⁴ but instead uses a few “preferred” amino acids which are frequently encoded (up to 8 times), while other amino acids are assigned only very few codons. Obviously these “preferred” codons are highly replaceable. It is interesting to note that optimized artificial codes often have 3 stop codons but no evident block structure.

Conclusion and further work

When iterating 500, 5000 and 5 million steps before changing temperature in MC , we found that the natural code is extremely fault tolerant, in contrast to the results of [?, ?]. However, since the space of general codes (10^{84}) is substantially larger than that of block respecting codes (10^{18}), and of shuffle codon codes (10^{65}), it could be that an implementation on a distributed system or parallel computer could yield many codes more optimal than the natural code. Nevertheless, since it seems that the restriction to block respecting or shuffled codon codes has no justification in nature, we conclude that the natural code is far too optimized for fault tolerance than to allow an explanation of its origin from a random surjective code optimized by FT criteria according to WAC similarity. Making some simple assumptions about mutation rate for a genetic code and comparing the number of MC steps required for optimization, one could speculate about the origin of the code (eg. originally a compact code for fewer amino acids) and the mechanism of change.

Many questions remain unanswered from our preliminary study.

⁴We plan to quantify this by computing FT_1, FT_2, FT_3 .

1. Using the simpler record-to-record travel algorithm of [?], can one find more optimized general codes than the natural code?
2. How does the $FT(c)$ measure using WAC correspond to the measures Δ_{mean} and MS_0 ?
3. Can one measure the extent to which the natural code is optimized against *general* (not necessarily block respecting or shuffle codes)? Using Lagrange multipliers Di Giulio [?] computed an approximation of 68% optimality for block respecting codes. Unfortunately, the method of [?] *changes* polarity values to unrealistic values. To rectify this, we plan to apply techniques from constraint programming languages.
4. Consider the mapping $c \mapsto opt(c)$, where opt is the optimized code (according to our Monte-Carlo procedure with simulated annealing). To what extent is opt a continuous map (i.e. if $c \approx c'$ then is $opt(c) \approx opt(c')$, where one defines an appropriate metric on the space of codes? Is the landscape of codes *rugged*? In analogy to [?], one can define *neutral networks* of codes as the set of codes which optimize to the same optimal code. What is the structure of such neutral networks?

- [6] S. Kirkpatrick, C.D. Gelatt Jr., and M.P. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.
- [7] C. Reidys, P.F. Stadler, and P. Schuster. Generic properties of combinatorial maps: Neutral networks of RNA secondary structures. *Bull. Math. Biol.*, 1996.
- [8] L. Wei, R.B. Altman, and J.T. Chang. Using the radial distribution of physical features to compare amino acid environments and align amino acid sequences. In R.B. Altman, A.K. Dunker, L. Hunter, and T.E. Klein, editors, *Pacific Symposium on Biocomputing '97*, pages 465–476. World Scientific: Singapore, New Jersey, London, Hong Kong, 1995. Symposium held in Maui, Hawaii from Jan 6–9, 1997.
- [9] C.R. Woese, D.H. Durge, S.A. Dugre, M. Condo, and W.C. Saxinger. On the fundamental nature and evolution of the genetic code. *Cold Spring Harbor Symp. Quant. Biol.*, 31:723–736, 1966.

S. Schönauer and P. Clote
 LFE Theoretische Informatik
 Institut für Informatik
 Universität München
 Oettingenstraße 67
 D-80538 München
 FAX: +49-89-2178 2238
 {schoenau,clote}@informatik.uni-muenchen.de

Acknowledgements

Thanks to B. Steipe for references, and pointing out the unsuitability of using the PAM250 matrix in our computations and to R. Matthes for computing the number of general codes.

References

- [1] M.O. Dayhoff, R.M. Schwartz, and B.C. Orcutt. A model of evolutionary change in proteins. In M.O. Dayhoff, editor, *Atlas of Protein Sequence and Structure*, volume Vol. 5, supp. 3, pages 3456–352. National Biomedical Research Foundation, Silver Springs, Md, 1978.
- [2] G. Dueck. New optimisation heuristics: The great deluge algorithm and the record-to-record travel. *J. Comput. Phys.*, 104:86–92, 1992.
- [3] M. Di Giulio. The extension reached by the minimization of the polarity distances during the evolution of the genetic code. *J. Mol. Evol.*, 29:288–293, 1989.
- [4] N. Goldman. Further results on error minimization in the genetic code. *J. Mol. Evol.*, 37:662–664, 1993.
- [5] D. Haig and L. Hurst. A quantitative measure of error minimization in the genetic code. *J. Mol. Evol.*, 33:412–417, 1991.

Amino acid	Init. code	Opt. code
Ala	3	2
Arg	3	1
Asn	2	3
Asp	3	1
Cys	2	3
Gln	3	4
Glu	2	1
Gly	3	2
His	3	2
Ile	2	3
Leu	3	3
Lys	2	1
Met	2	7
Phe	3	3
Pro	3	3
Ser	3	2
Thr	3	2
Trp	2	3
Tyr	3	2
Val	2	2
stop	2	4