

Advances in Complex Systems
© World Scientific Publishing Company

MONTE CARLO SIMULATION AND STATISTICAL ANALYSIS OF THE EFFECT OF CODING TABLE SPECIFICITY ON GENETIC INFORMATION CODING

EVIN GULTEPE

Physics Department, Northeastern University, Boston, Massachusetts, 02115, USA
gultepe.e@neu.edu

MEHMET LEVENT KURNAZ

Department of Physics, Bogazici University, 34342 Bebek, Istanbul, Turkey
levent.kurnaz@boun.edu.tr

Received (received date)

Revised (revised date)

We present a computer simulation, which is inspired by Penna model, to help understanding the effect of genetic coding tables on population dynamics. To represent populations we used real and artificial gene sequences in this model. We coded these sequences using different amino acid tables in Nature, the standard table as well as the tables which are used by mitochondria and some eukaryotes. Contrary to expectations, we find that the standard code table used in most organisms in Nature, does not give the most resilient coding against mutations in our model.

1. INTRODUCTION

Modeling population dynamics has been popular in physics community for a while mostly because complexity of the system bears the necessity of the statistical and computational tools. Physicists brought the models which can give rise to computational simulations together with *looking for the simplest solution* approach into this subject. Among all others [1, 2, 3, 4], Penna model [5] is the most extensive simulation scheme used in population dynamics. Simulations for population dynamics usually take many different factors of life into account. In real life it is very difficult to have only one of the aspects of life count where we neglect the effects of everything else. For example in real life you cannot say that the only cause for death is point mutations because then you have to keep the individuals in the system from dying of "old age" or of malnutrition or of fighting amongst the members. In simulations as well as in probabilistic calculations it is much easier to ignore all these facts of life and concentrate only on one simple aspect. In this work we have neglected all the other aspects of life and for simplicity concentrated only on the effect of point mutations and the robustness of the various genetic coding tables.

2 *Gultepe - Kurnaz*

Genetic information of all living organism (except some viruses) is stored in DNA. The segment of DNA which contains necessary information to produce a specific protein is called *gene*. A real gene is composed of two different parts: a coding portion and a non-coding portion. The coding part, exon, is responsible for protein synthesis whereas the rest, intron, does not code for a protein.

The information in DNA is coded by using four different types of monomers adenine (A), guanine (G), cytosine (C) and thymine (T). These monomers are the letters of the genetic alphabet and they construct 3-letters long words, *codons*. Every codon on DNA codes an amino-acid during the protein synthesis (except for the STOP codons). There are 4^3 , 64 possible combination of codons available on DNA. However; in Nature there are only 20 amino acids available for protein coding and as a result there is no one-to-one codon-amino acid correspondence. The table which determines how the codons are mapped into the amino acids is called *amino acid table* or *genetic coding table*.

For a long time it was believed that the amino acid table of Nature was universal, *Standart Genetic Code* (SGC). However; we now know that there are few exceptions for instance in mitochondria and certain protozoa, which use a different coding scheme [6]. In these particular alternate coding tables the same number of amino acids is used, but some of these amino acids are coded by different codons. For example, in *SGC*, the codon “AUA” codes Isoleucine, the codon “UGA” codes Stop, and the codons “AGA” and “AGG” code Arginine . However, in the table of *Vertebrate Mitochondrial Code*, they encode Methionine, Tryptophan, and Stop respectively (www.ncbi.nlm.nih.gov/Taxonomy/Utils/wprintgc.cgi?mode=c).

Recently, we have developed a Monte Carlo simulation model [7] inspired by the Penna model to investigate the significance of the number of amino acids in population dynamics. In that model, each individual was represented by a human cytokine gene sequence and mutation was assumed to be the only cause of death by eliminating all other effects. In that study, it has been shown that for maximum tolerance against mutations, the number of amino acids which codes the genetic information, is bounded between 20 to 24. The number of amino acids used in Nature, 21 (20 amino acids plus the Stop codon), is in this optimum range.

In this paper, we used the same model to investigate the endurance of different amino acid tables against point mutations neglecting any other causes of death. We have also carried out similar analyses using probabilistic calculations, and compared the results to the Monte Carlo simulations, with relatively insignificant difference.

2. COMPUTATIONAL METHOD

In our model, an individual was represented by three different gene sequences. First, we have used a real gene from Nature *human cytokine* (LD78 Homo sapiens blood lymphocyte gene on the DNA 17th chromosome) (<http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi?db=nucleotide&val=285912>)

same as in the previous model. This gene is playing an important role in immune system of human body, hence any problem in a functional LD78 protein would be lethal. Afterwards we used the same model on different mammalian genes, in this paper we also report the results for human ARNT gene (Homo sapiens aryl hydrocarbon receptor nuclear translocator, transcript variant 1, mRNA) as an example.

Next, we have created an artificial human gene, the average human gene, which basically reflects the codon usage frequency found in Homo sapiens. This gene consists of 1000 nucleotides (frequency is taken as an integer value per one thousand nucleotides given in [http://www.kazusa.or.jp/codon/cgi-bin/showcodon.cgi?species=Homo+sapiens+\[gbpri\]](http://www.kazusa.or.jp/codon/cgi-bin/showcodon.cgi?species=Homo+sapiens+[gbpri]) in a randomly chosen sequence). This artificial gene is considered to represent the whole human genome. This assures that the results we have obtained are not specific to the human cytokine gene but for whole genome.

For the sake of simplicity, we have not included reproduction and we also have neglected all other effects causing death, except mutation. Using this model, we have investigated the effects of mutations on the population size. A mutation in this model was taken as a change of one nucleotide in the gene. For the sake of simplicity, we have assumed a mutation to be either lethal or silent depending on whether it causes a change in the amino acid chain or not. For this study, we kept all mutation rates equal like in the Jukes-Cantor mutation scheme [8].

When a mutation takes place on the exon part, there are two possibilities: The changed codon either will code the same amino acid or it will code a different amino acid [Fig. 1] since an amino acid can be coded by more than one codon. If the mutant codon still codes for the same amino acid, this mutation is taken to be harmless because at the end it will not affect the synthesized protein. However, if the mutant codon codes for a different amino acid, the protein cannot be produced and the individual would simply die.

To be more explicit, the codons AAA and AAG code the same amino acid, “lysine”; hence if AAA turns into AAG as a result of a mutation the amino acid will not change and the protein can be constructed safely. However; if AAA turns into AGA, which codes the amino acid “arginine”, the amino acid chain will change and we assume that the protein can not build up, which means the represented organism will die. There can be a mutation which converts AAA to AAX where $X \neq A, G, C, \text{ or } T$; then the individual dies automatically. As a model, we are looking at a simpler case where a mutation changes A to one of G, C, or T, but not X.

Since reproduction is not included in the model, the population can only diminish. The decrease in population can be found by calculating the probability of a deleterious mutation. The probability of the mutation changing the amino acid depends on the codon; so one needs to find the probability of hitting each different codon type. First, the probability of hitting a codon type (P_α) is calculated as the ratio of the number of codons of that type in the gene (N_α) to total number of codons. Then we need to exclude the mutations that do not cause a change in the amino acid and calculate the probability of a change occurring in the amino acid

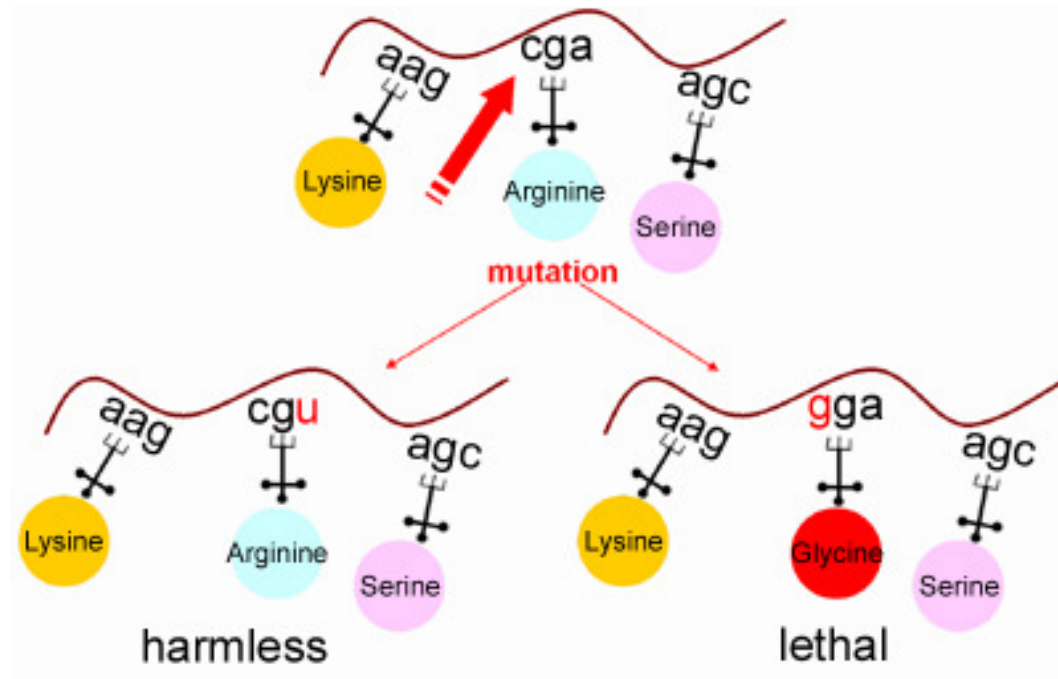


Fig. 1. A mutation on exon part may be either harmless or lethal depends on whether it change the coded amino acid.

caused by a change in one nucleotide ($P(d/\alpha)$).

We used only the exon (protein coding) part of the gene considering any mutations on the intron part is harmless. As a simple example, the human cytokine gene has a total length of 2068 nucleotides; 621 nucleotides in exon part and 1447 ones in intron. The probability of hitting the exon part of the gene is simply the ratio of the exon part to the total gene:

$$P(\text{hitting exon}) = \frac{621}{2068} = 0.3032 \quad (1)$$

Hence; the probability of having a *deleterious* mutation for all of the gene is simply a product of mutation probability and probability of hitting the exon part of gene. As the chances of hitting any part of the gene is a same, we can neglect the intron part in the simulation since this would only be a multiplicative constant in the problem. Therefore the probability of having a deleterious mutation for all of the human cytokine gene is simply:

$$P(\text{deleterious}) \propto \sum_{\alpha=1}^{64} [P_{\alpha} P(d/\alpha)] = 0.7729 \quad (2)$$

The survival probability can be calculated by:

$$P(\text{surviving}) = 1 - P(\text{deleterious}) = 0.2271 \quad (3)$$

If we take an initial population of N_0 genes (individuals), after n number of mutations, to the first order, the number of surviving individuals (N_n) is given by:

$$N_n \approx N_0 P(\text{surviving})^n \quad (4)$$

Hence, we obtain the “probability of survival” with the slope of the number of surviving individuals versus time graph:

$$\text{slope} \approx \ln[P(\text{surviving})] = -1.4823 \quad (5)$$

Similarly the probability of survival can be calculated for all the genes separately. However in this calculation once we make a change in the gene sequence and if the individual survives, we forget about the change we have made and restart the process for the second mutation cycle with the original gene sequence. In Nature, if the individual survives, the second mutation cycle starts with the mutated gene sequence and not the original one. Therefore, to be able to get closer to Nature we have also written a simulation code which allows for the mutation in the gene sequence to be kept in the next mutation mutation cycle.

3. SIMULATION

The calculation scheme above assumes that the mutations at each step are not preserved. However in Nature this is not true. After the first mutation, the system will not forget about this mutation and restart with the original sequence. To test the importance of “forgetting” the previous mutations in our calculations we have also done simulations where the mutations, if not lethal, are kept in the gene sequence. In this simulation, the population consists of individuals which are described by only one gene. Genes are represented by arrays which contain 0, 1, 2, and 3’s instead of the nucleotides Adenine (A), Guanine (G), Cytosine (C) and Thymine (Uracil (U)) respectively. A sign bit which shows if the gene has a deleterious mutation (1) or not (0) is also included in the array [7].

In every time step, all of the individuals undergo a random mutation. If the mutation is deleterious, i. e. if it changes the amino acid, the sign bit is changed to ‘1’, the individual is deleted from the population (death) and the time step is recorded. Otherwise, the sign bit is kept ‘0’ and the individual survives. The initial population consists of 10^9 individuals.

When the mutation cycle is finished, the number of surviving individuals in each time step is calculated. Since the probability of mutation is independent of the number of individuals, surviving individuals also give us the population size. Hence, we have an exponential population decay and the exponent depends on the probability of surviving ($P(\text{surviving})$). Logarithm of the population is fitted to a straight line and the slope of the line is calculated. All simulations are run for 10 times and probability of surviving is calculated according to the weighted average of these 10 runs. Increasing the number of runs did not produce any significant changes

in our results, and also as our aim was to use these simulations as a justification for the computations, we did not increase the number of simulation runs.

4. RESULTS AND DISCUSSION

In this work, we have investigated the following amino acid tables in addition to *SGC*:

- Alternative Yeast Nuclear Code (AYNC),
- Ascidian Mitochondrial Code (AMC),
- Blepharisma Nuclear Code (BNC),
- Ciliate, Dasycladacean and Hexamita Nuclear Code (CDHNC),
- Echinoderm Mitochondrial Code (EMC),
- Euplotid Nuclear Code (ENC),
- Flatworm Mitochondrial Code (FMC),
- Invertebrate Mitochondrial Code (IMC),
- Mold, Protozoan, and Coelenterate Mitochondrial Code and Mycoplasma/Spiroplasma Code (MSC),
- Vertebrate Mitochondrial Code (VMC), and
- Yeast Mitochondrial Code (YMC).

The results of our simulations, using two different human gene sequences, are given in Table 1 and Table 2. We have selected these two genes as representatives, however, the same results have been obtained in many other human genes simulated. We have also used some genes from *Mus musculus* (common house mouse) and *Rattus* (rat) where we have obtained similar results. As these detailed results are more appropriate for an evolutionary biology journal, we report only the representative results.

Also biologists have also been trying to find simplified amino acid alphabets. The simplest reason for this is the fact that even though some mutations result in a different aminoacid, this aminoacid can still hold the place of the previous one and keep the resulting protein functional. One of these methods is the MJ matrix constructed using Wang and Wang's method [9] which is based on Miyazawa-Jernigans (MJ) residue - residue potentials [10]. Another method is the BLOSUM50 matrix, built using Murphy, Wallqvist and Levys method [11] derived by Henikoff and Henikoff [12]. We have used both of these methods with our calculations to reduce the aminoacids to four major groups. This basically should be the worst case scenario to test if the aminoacid substitutions play a crucial role to change the results presented below.

If it is assumed that the genetic code table is optimized for increasing chance of survival against mutations, then the results of Table 1 and Table 2 cause many concerns. Even though there are few code tables giving results which favor the usage of *SGC* (like VMC, AYNC, and YMC), we can see that if we use a different code

Table 1. Average slopes of the population decrease comparing different genetic code tables with human cytokine gene. Larger the magnitude of the slope is, the more the survival chance.

Code	Simulation	Calculation	BLOSUM50	MJ
FMC	-1.4056 ± 0.0005	-1.4053	-0.5360	-0.4894
EMC	-1.4164 ± 0.0005	-1.4163	-0.5555	-0.5035
IMC	-1.4320 ± 0.0001	-1.4319	-0.5555	-0.5035
ENC	-1.4415 ± 0.0005	-1.4409	-0.5611	-0.5035
BNC	-1.4691 ± 0.0003	-1.4685	-0.5734	-0.5133
MSC	-1.4759 ± 0.0003	-1.4755	-0.5724	-0.5035
CDHNC	-1.4784 ± 0.0005	-1.4779	-0.5715	-0.5035
AMC	-1.4784 ± 0.0005	-1.4779	-0.5810	-0.5035
SGC	-1.4830 ± 0.0005	-1.4826	-0.5888	-0.5305
VMC	-1.5020 ± 0.0001	-1.5017	-0.5907	-0.5639
YMC	-1.5677 ± 0.0007	-1.5638	-0.6714	-0.5724
AYNC	-1.5800 ± 0.0001	-1.5793	-0.6444	-0.5620

Table 2. Average slopes of the population decrease comparing different genetic code tables with human ARNT gene. Larger the magnitude of the slope is, the more the survival chance.

Code	Simulation	Calculation	BLOSUM50	MJ
FMC	-1.4038 ± 0.0004	-1.4036	-0.4931	-0.4623
EMC	-1.4101 ± 0.0007	-1.4099	-0.4969	-0.4736
CDHNC	-1.4268 ± 0.0004	-1.4268	-0.4813	-0.4481
IMC	-1.4288 ± 0.0004	-1.4285	-0.4969	-0.4736
BNC	-1.4368 ± 0.0004	-1.4368	-0.4925	-0.4597
ENC	-1.4566 ± 0.0005	-1.4565	-0.5054	-0.4736
AMC	-1.4586 ± 0.0005	-1.4583	-0.5186	-0.5027
MSC	-1.4609 ± 0.0006	-1.4607	-0.5092	-0.5027
SGC	-1.4653 ± 0.0005	-1.4650	-0.5120	-0.4783
VMC	-1.4882 ± 0.0008	-1.4878	-0.5300	-0.5382
AYNC	-1.5215 ± 0.0005	-1.5213	-0.5469	-0.4925
YMC	-1.5295 ± 0.0005	-1.5271	-0.5735	-0.5262

table, for example *FMC*, our white blood cell production would be more resilient towards mutations.

To be certain that the results obtained in Table 1 and Table 2 do not depend on particular genes, we have first created an artificial average human gene and we run simulations using this average human gene as representative of individuals. Table 3 shows the results from these simulations.

In Table 3, *FMC* still performs much better and the worst performance is still by *AYNC* followed by *YMC*. The human average gene gives comparable results to the

Table 3. Average slopes of the population decrease using different genetic code tables and the average human gene. Larger the magnitude of the slope is, the more the survival chance.

Code	Simulation	Calculation	BLOSUM50	MJ
FMC	-1.4081 ± 0.0005	-1.4083	-0.5140	-0.4885
EMC	-1.4204 ± 0.0005	-1.4207	-0.5213	-0.5027
IMC	-1.4438 ± 0.0003	-1.4439	-0.5213	-0.5027
CDHNC	-1.4498 ± 0.0001	-1.4501	-0.4992	-0.4773
BNC	-1.4554 ± 0.0006	-1.4558	-0.5116	-0.4916
ENC	-1.4595 ± 0.0005	-1.4596	-0.5242	-0.5027
MSC	-1.4655 ± 0.0005	-1.4658	-0.5274	-0.5027
AMC	-1.4693 ± 0.0006	-1.4697	-0.5399	-0.5027
SGC	-1.4712 ± 0.0004	-1.4716	-0.5316	-0.5101
VMC	-1.4969 ± 0.0005	-1.4971	-0.5507	-0.5658
YMC	-1.5540 ± 0.0006	-1.5543	-0.6119	-0.5686
AYNC	-1.5803 ± 0.0001	-1.5804	-0.5892	-0.5420

genes chosen in this work, indicating that the advantage of *SGC* cannot be simply explained on the basis of singular mutations without taking higher order effects like protein folding into account. Using different substitution matrices, we get slightly different results, the performance of FMC is not as good, but as our aim was to see the performance of the standard code. Different substitution matrices was not the explanation why Nature uses the standard code even though its performance is not the best among the coding tables used in this work.

5. CONCLUSION

In this paper, we used a computer simulation parallel to a series of probabilistic calculations to study the robustness of different coding tables in response to random mutations. Furthermore, we changed the genetic code used in the simulations to analyze its effect on population stability.

We have used different code tables utilized in the Nature in connection with two sample genes, human cytokine gene and human ARNT gene . Since these genes are one of the key factors for our bodies, we assumed that if the organism fails to produce any of the proteins coded by these genes, it will not be able to survive.

The simulations as well as calculations show that *SGC*, which is being used in most of the vertebrates, actually does not give the most resilient organisms against point mutations if we only apply simple simulation rules.

To test these results, we have also created an artificial average human gene. If we compare our results as *SGC* versus the rest of the coding schemes, the results do not change, i. e., *SGC* is still not the best solution. However, when we compare other code tables within themselves we see that some code tables give better results

with different human genes whereas others give better results with the artificial average human gene. This result is most pronounced with *CDHNC*, which survives much better if we use the average human gene than the human cytokine gene.

It should be noted that this is an over-simplified model, where we have simply compared our Monte Carlo simulation program against probability calculations. Once this is established, it will be possible to expand this analysis to (i) unequal mutation rates, (ii) functionally conserved amino acid substitutions based on substitution or similarity matrices, and (iii) genes from different organisms. The work of Maeshiro and Kimura [13] has suggested that any coding table has two important features: robustness and changeability. In this work we have concentrated on the performance of different coding tables with respect to robustness, however this by no means answers how the standard genetic code has evolved. In order to address this issue, one would also need to incorporate the changeability aspect of a generic coding table, and study the effect of changeability under rapidly shifting environmental conditions. Ongoing work is trying to achieve this challenging task using similar simulations.

Acknowledgments

We are grateful to Dr. Isil Aksan Kurnaz and Dr. Muhittin Mungan for their contributions on the model and the calculations. This work has been supported by Bogazici University under BAP 07B302.

References

- [1] Medeiros, N. G. F. and Onody, R. N., Heumann-Hotzel model for aging revisited , *Phys. Rev. E* **64** p. 041915 (2001).
- [2] Mueller, L. D. and Rose, M. R., Evolutionary theory predicts late-life mortality plateaus, *PNAS* **93**, pp. 15249–15253 (1996).
- [3] Cui, Y., Chen, R. S., and Wong, W. H., The coevolution of cell senescence and diploid sexual reproduction in unicellular organisms, *PNAS* **97**, pp. 3330–3335 (2000).
- [4] Heumann, M. and Htzel, M., Generalization of an aging model, *J. Stat. Phys.* **79**, pp. 483–490 (1995).
- [5] Penna, T. J. P., A Bit-String Model for Biological Aging, *J. Stat. Phys.* **78**, pp. 1629–1633 (1995).
- [6] Weaver, R. F., *Molecular Biology* (McGraw-Hill, New York, 2002).
- [7] Gultepe, E. and Kurnaz, M. L., Monte Carlo simulation and statistical analysis of genetic information coding, *Physica A.* **357**, pp. 525–533 (2005).
- [8] Jukes, T. H. and Cantor, C. R., Evolution of Protein Molecules, in *Mammalian Protein Metabolism*, edited by H. N. Munro, (Academic Press, New York, 1969), pp. 21-123.
- [9] Wang, J. and Wang, W., A computational approach to simplifying the protein folding alphabet, *Nature Structural Biology* **6**, pp. 1033–1038 (1999).
- [10] Miyazawa, S. and Jernigan, R. L., Residue-residue potentials with a favorable contact pair term and an unfavorable high packing density term, for simulation and threading, *J. Mol. Biol.* **256**, pp. 623–644 (1996).
- [11] Murphy, L.R., Wallqvist, A. and Levy, R. M., Simplified Amino Acid Alphabets for

10 *Gultepe - Kurnaz*

Protein Fold Recognition and Implications for Folding, *Prot. Eng.* **13**, pp. 149–152 (2000).

- [12] Henikoff, S. and Henikoff, J. G., Amino Acid Substitution Matrices from Protein Blocks, *Proc. Natl. Acad. Sci.* **89**, pp. 10915–10919 (1992).
- [13] Maeshiro, T. and Kimora, M., The role of robustness and changeability on the origin and evolution of genetic codes, *Proc. Natl. Acad. Sci.* **95**, pp 5088–5093 (1998).