An Improved Implementation of Effective Number of Codons (N_c)

Xiaoyan Sun,[1] Qun Yang,[1] and Xuhua Xia*[,1,2]

[1]State Key Laboratory of Paleobiology and Stratigraphy, Nanjing Institute of Geology and Palaeontology, Chinese Academy of Science, Nanjing, China

[2]Department of Biology and Center for Advanced Research in Environmental Genomics, University of Ottawa, Ottawa, Ontario, Canada

***Corresponding author:** E-mail: xxia@uottawa.ca.

Associate editor: Sudhir Kumar

Abstract

The effective number of codons (N_c) is a widely used index for characterizing codon usage bias because it does not require a set of reference genes as does codon adaptation index (CAI) and because of the freely available computational tools such as CodonW. However, N_c, as originally formulated has many problems. For example, it can have values far greater than the number of sense codons; it treats a 6-fold compound codon family as a single-codon family although it is made of a 2-fold and a 4-fold codon family that can be under dramatically different selection for codon usage bias; the existing implementations do not handle all different genetic codes; it is often biased by codon families with a small number of codons. We developed a new N_c that has a number of advantages over the original N_c. Its maximum value equals the number of sense codons when all synonymous codons are used equally, and its minimum value equals the number of codon families when exactly one codon is used in each synonymous codon family. It handles all known genetic codes. It breaks the compound codon families (e.g., those involving amino acids coded by six synonymous codons) into 2-fold and 4-fold codon families. It reduces the effect of codon families with few codons by introducing pseudocount and weighted averages. The new N_c has significantly improved correlation with CAI than the original N_c from CodonW based on protein-coding genes from *Saccharomyces cerevisiae*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Escherichia coli*, *Bacillus subtilis*, *Micrococcus luteus*, and *Mycoplasma genitalium*. It also correlates better with protein abundance data from the yeast than the original N_c.

Key words: effective number of codons, codon usage, codon adaptation, translation elongation, gene expression.

Introduction

Ever since the empirical documentation of the correlation between codon usage and transfer RNA (tRNA) abundance (Ikemura 1981), studies on codon–anticodon adaptation have progressed in theoretical elaboration (Bulmer 1987, 1991; Xia 1998, 2008; Higgs and Ran 2008; Jia and Higgs 2008; Palidwor et al. 2010), in critical tests of alternative theoretical predictions (Xia 1996, 2005; Carullo and Xia 2008; van Weringh et al. 2011), and, in particular, in formulation and improvement of various codon usage indices to characterize codon usage bias (Sharp and Li 1987; Wright 1990; Xia 2007). Codon usage indices such as CAI (Sharp and Li 1987; Xia 2007) are positively correlated not only with translation elongation efficiency but also with splicing strength of yeast intron splice sites (Ma and Xia 2011) and translation initiation efficiency measured by ribosomal loading (Xia et al. 2011).

Codon usage bias is often measured by two classes of indices, one class being codon specific and the other being gene specific. A representative of the first class is the relative synonymous codon usage (Sharp et al. 1986), and representatives of the second class are the effective number of codons or N_c (Wright 1990), the Codon Adaptation Index (CAI; Sharp and Li 1987; Xia 2007), the frequency of optimal codons or F_{op} (Ikemura 1981), and the codon bias index (CBI; Bennetzen and Hall 1982). Although comparative studies (Comeron and Aguade 1998; Duret and Mouchiroud 1999; Coghlan and Wolfe 2000) suggest that CAI is the best in predicting gene expression levels, N_c has one advantage over CAI, F_{op}, or CBI in that it does not require external information (which is often unavailable) other than the codon frequencies of the gene. In contrast, CAI requires a reference set of known highly expressed genes, F_{op} needs information on relative tRNA abundance (it defines translationally optimal codons as those forming Watson–Crick base pair with the anticodon of major tRNA species in each codon family), and CBI needs information on both gene expression and relative tRNA abundance. For this reason, N_c has been frequently used in biological research to characterize codon usage bias, partly facilitated by the CodonW program and its web server at http://mobyle.pasteur.fr/cgi-bin/portal.py#forms::codonw (last accessed August 28, 2012).

Sometimes additional information on tRNA does not help predict gene expression or codon usage. For example, the *Bacillus subtilis* genome codes a $tRNA^{Ala/GGC}$ for decoding GCY codons. The GCC codon, which forms Watson–Crick base pair with the anticodon, is not used as frequently as the GCU codon which wobble-pairs with the anticodon. One might argue that, according to previous studies

(Grosjean et al. 1978; Fiers and Grosjean 1979; Grantham et al. 1981; Ikemura 1981), the intermediate binding strength between codon and anticodon is optimal, especially for highly expressed genes. A weak binding at the third codon position is preferred with strong binding at the first two codon positions, and a strong binding at the third codon position is preferred with weak binding at the first two codon positions. Thus, GCU is preferred because of the strong binding in the first two positions. However, this explanation does not work for Gly where four tRNA$^{Gly/GCC}$ genes are present for decoding GGY codons, and GGC is used more frequently than GGU. Before we gain a better understanding between codon–anticodon adaptation, codon usage bias indices, such as N_c, remain useful.

However, there are several problems with N_c, both in concept and in computer implementation, that affect its performance and limit its application. We will detail them individually and propose modifications and improvements.

Conceptual Problems with N_c and Solutions

To facilitate presentation, we will list the N_c-related definitions below. For an individual codon family of m synonymous codons whose counts are $n_1, n_2, \ldots, n_m$, we have $n = \sum n_i$ and $p_i = n_i/n$. The original N_c formulation for this codon family is as follows:

$$F_{CF} = \frac{n\sum_{i=1}^{m} p_i^2 - 1}{n-1} \qquad (1)$$
$$N_{c.CF} = 1/F_{CF}$$

where the subscript CF stands for "codon family" and refers to the fact that F_{CF} and $N_{c.CF}$ are for a specific codon family instead of for a gene.

One problem, which was recognized at the very beginning (Wright 1990), is that $N_{c.CF}$ can have values much greater than m. For example, if a 4-fold GGN codon family ($m = 4$) has $n_i = 2$, then $n = 8$, $p_i = 0.25$, and $N_{c.CF} = 7$ according to equation (1) instead of the maximum expected value of 4. When $N_{c.CF}$ values from different codon families are compiled to arrive at a final N_c value for a gene, the value can be much greater than 61 for a standard genetic code, especially when n is small. This problem has not been fixed except by a post hoc rescaling of the resulting N_c values, such as is done in CodonW (e.g., the N_c values are rescaled to the range of 20–61 for standard genetic code). Such rescaling does not address the problem that N_c for a gene can be dramatically biased by codon families each with few codons. In addition, the rescaling is conceptually confusing. For example, when one obtains an N_c of 61, one expects the codon usage to be equal (unbiased). However, almost all genes with an N_c value of 61 computed from CodonW actually do not use synonymous codons equally. In other words, many genes get an N_c value of 61 for wrong reasons. It is paradoxical that the formulation of F_{CF} in equation (1), originally intended for correcting bias associated with small n in measuring homozygosity in population genetics, becomes the very source of often dramatic bias associated with small n in the context of measuring codon usage bias.

Another problem with the formulation in equation (1) is the loss of information. If $n = 2$ for a 2-fold codon family with $n_1 = n_2 = 1$, then F_{CF} is 0, and the data cannot be used to compute $N_{c.CF}$. For a 3-fold codon family, F_{CF} is also 0 when $n_1 = n_2 = n_3 = 1$ or when $n_1 = n_2 = 1$ and $n_3 = 0$. For a 4-fold codon family, F_{CF} is also 0 when $n_1 = n_2 = n_3 = n_4 = 1$ or when $n_1 = n_2 = n_3 = 1$ and $n_4 = 0$. This implies that information contained in codon families with a small n often cannot be used.

To alleviate the two problems above, one may redefine F simply as

$$F_{CF} = \sum_{i=1}^{m} p_i^2 \qquad (2)$$

Now the maximum N_c for a codon family with m codons will be exactly m (when synonymous codons are equally used), so that, for the standard genetic code, the maximum possible value for N_c would be exactly 61. The minimum of N_c based on F_{CF} in equation (2) is the number of codon families when only one codon is used in each codon family. F_{CF} in equation (2) and that in equation (1) approach each other when n becomes very large. When n is small, F_{CF} in equation (2) is more preferable than that in equation (1). As will be shown later, the new F_{CF} not only eliminates the clumsy need for N_c rescaling but also leads to better prediction of protein abundance and better correlation with CAI.

We have not yet addressed the potential bias introduced by small n. Suppose we have a 2-fold codon family with $n_1 = 90$ and $n_2 = 10$. This would give us an $N_{c.CF}$ of 1.22 based on F_{CF} defined in equation (2). However, if we have $n_1 = 9$ and $n_2 = 1$, the resulting $N_{c.CF}$ is the same, but $N_{c.CF}$ with $n = 100$ is clearly more trustworthy than $N_{c.CF}$ with $n = 10$. Proper handling of small n values is crucial for a good codon usage index.

Two commonly used approaches to alleviate the effect of a small n are 1) pseudocount and 2) weighting. With the pseudocount approach, we may redefine

$$F_{CF} = \sum_{i=1}^{m} \left(\frac{n_i+1}{n+m}\right)^2 \qquad (3)$$

Equation (3) implies that, when there is no information for a codon family (i.e., when $n = 0$), then we assume equal codon usage. This is reasonable biologically because a codon family that is hardly used is expected not to be under strong selection for codon usage bias, although mutation bias may also cause codon usage bias (Xia 1996, 2005). The approach is also reasonable statistically because we adopt the (implicit) null hypothesis of no codon usage bias when there is no data to reject it. A more general specification of the pseudocount approach is to replace 1 in the numerator of equation (3) by a constant C and the m in the denominator by m*C. In conjunction with the pseudocount approach, we may also specify a minimum n for a codon family to be included in computing N_c.

Although the pseudocount approach can be applied to the computation of F_{CF}, the weighting approach can be applied to compiling individual $N_{c.CF}$ values to the final N_c value for the gene so as to minimize the potential bias introduced by codon families with small n values. Suppose we have three 2-fold codon families, with $n_1 = n_2 = 200$, $n_3 = 4$, and $F_{CF1} = F_{CF2} = 1$, and $F_{CF3} = 0.5$. The average of the three F values ($\bar{F}$) is $2.5/3 \approx 0.8333$, and the number of effective codons contributed by the three codon families is $3/\bar{F} = 3.6$. However, it is unreasonable to have equal weight for the three F values obtained with dramatically different n values. A weighted $\bar{F}$ is

$$\bar{F} = \frac{n_1F_{CF1} + n_2F_{CF2} + n_3F_{CF3}}{n_1 + n_2 + n_3} = \frac{402}{404} \approx 0.9951 \qquad (4)$$

Thus, the three codon families will contribute only 3.0149 (=$3/\bar{F}$) to the final N_c instead of 3.6 as before. Such a value reflects better the extremely strong codon usage bias observed in the two codon families with a large n, which suggests strong codon usage bias.

With the weighting scheme, the final gene-specific N_c is

$$N_c = N_s + \frac{K_2 \sum_j^{K_2} n_j}{\sum_{j=1}^{K_2} \left(n_j F_{CF.j}\right)} + \frac{K_3 \sum_j^{K_3} n_j}{\sum_{j=1}^{K_3} \left(n_j F_{CF.j}\right)} + \frac{K_4 \sum_j^{K_4} n_j}{\sum_{j=1}^{K_4} \left(n_j F_{CF.j}\right)} \qquad (5)$$

where N_s is the number of codon families with a single codon, for example, the Met and the Trp codon families in the standard genetic code, with a single AUG and UGG codon, respectively, $F_{CF.j}$ is F_{CF}, defined in equation (3) for codon family j, and K_i is the number of i-fold codon families. There are cases where $N_s \neq 2$. For example, the vertebrate mitochondrial code (transl_table = 2) has $N_s = 0$. In contrast, the alternative yeast nuclear code (transl_table = 12) has N_s equal to 3, that is, with a Ser family containing a single CUG codon in addition to the Met and Trp codon families. Similarly, Blepharisma nuclear code (transl_table = 15) has an additional single-codon Gln (UGA) codon family, leading to $N_s = 3$. Two other genetic codes with $N_s = 3$ are the Chlorophycean mitochondrial code (transl_table = 16) and the *Scenedesmus obliquus* mitochondrial code (transl_table = 22), each with an additional single-codon Leu (UAG) codon family. *Thraustochytrium* mitochondrial code (transl_table = 23) also has $N_s = 3$ with an additional single-codon Leu (UUG) codon family.

Most of the known genetic code have only one 3-fold codon family, that is, the Ile codon family, so $K_3 = 1$. However, there are several exceptions. For example, in addition to the 3-fold Ile codon family, the echinoderm and flatworm mitochondrial code (transl_table = 9) has a 3-fold Asn (AAH, where H stands for A, C, or U) codon family, the euplotid nuclear code (transl_table = 10) has a 3-fold Cys (UGH) codon family, and the alternative yeast nuclear code (transl_table = 12) has a 3-fold Leu (CUH) codon family. In particular, the alternative flatworm mitochondrial code (transl_table = 14) has three 3-fold codon families, Ile (AUH), Asn (AAH), and Tyr (UAH). Multiple 3-fold codon families in one genetic code were unknown to Wright when he formulated N_c (Wright 1990).

Equation (5) does not include 6-fold or 8-fold compound codon families. We provide reasons for why such compound codon families should be broken into two separate codon families in computing N_c in the next section.

Implementation Problems with N_c and Solutions

There are two problems with the implementation of N_c. The first involves the diverse array of genetic codes. Few implementations of N_c accommodate all genetic codes, which have now numbered 18. Currently, the most comprehensive N_c implementation is CodonW, which accommodates eight different genetic codes. However, there is a misspecification of the yeast mitochondrial code. CTN codons code for Thr in this genetic code, but CodonW specifies CTN as stop codons. In any case, leaving out the other 10 genetic codes severely limits the utility of N_c, especially for evolutionary biologists who are particularly interested in odd creatures that tend to feature one of those rare genetic codes. The implementation of the new N_c function in the most recent version of DAMBE (version 5.3.00) accommodates all known genetic codes.

The other problem, which is partially in concept and partially in implementation, involves the compound codon families of which there are two kinds. The first is often referred to as the 6-fold codon families each being composed of a 2-fold codon family and a 4-fold codon family, for example, those encoding amino acids Arg, Leu, and Ser in the standard genetic code. The second kind contains eight synonymous codons made of two 4-fold codon families. For example, amino acid Ser in the alternative flatworm mitochondrial code (transl_table = 14) has eight synonymous codons that belong to two synonymous codon families, that is, TCN and AGN codon families (where N stands for any nucleotide). This genetic code was first reported (Bessho et al. 1992) after the original formulation of N_c by Wright (1990). The existence of this particular genetic code was disputed before (Telford et al. 2000) but was subsequently verified in at least two nematode species (Jacob et al. 2009).

The two codon families within each compound codon family are translated by different tRNAs and consequently could be under quite different selection pressure. Take, for example, the Leu codons in *Escherichia coli* 536. The 4-fold CUN codon family is translated by tRNAs from five genes, one with a G at the first anticodon site to translate Y-ending codons (where Y stands for C or U) and four with a C at the first anticodon site to translate the CUG codon, with no tRNA that forms Watson–Crick base pair with the CUA codon. This leads to a dramatic underuse of the CUA codon and over-representation of the CUG codon relative to other synonymous codons in the *E. coli* 536 genome. In contrast to the strong codon usage bias in the 4-fold Leu (CUN) codon family, there is no codon usage bias in the 2-fold UUR codon family (where R stands for A or G). This 2-fold codon family is translated by tRNAs encoded by two tRNA genes in the *E. coli* 536 genome, one with a C at the first anticodon site to translate the UUG codon and the other with a U at the first anticodon position to translate the UUA codon. This implies little selection in favor of one codon

Table 1. Pearson Correlation Coefficient (*r*) between Codon Adaptation Index and the Two Versions of N_c: the New N_c Developed in This Article and Implemented in DAMBE ($N_{c.New}$) and N_c from CodonW ($N_{c.Old}$).

Species	GC%[a]	N_{gene}[b]	Ref.File[c]	*r* ($N_{c.New}$)	*r* ($N_{c.Old}$)	T[d]	P[d]
Escherichia coli	50.80/51.82/55.88	4,254/4,233	Eeco_h	−0.7743	−0.7382	3.884	<0.0001
Bacillus subtilis	43.50/44.23/44.53	4,176/4,141	Ebsu_h	−0.5807	−0.4737	−6.766	<0.0001
Micrococcus luteus	73.00/73.16/95.14	2,236/2,235	rib. prot.	−0.7853	−0.7331	−4.127	<0.0001
Mycoplasma genitalium	31.69/31.55/23.04	475/473	rib. prot.	−0.7629	−0.7173	−1.551	0.1209
Saccharomyces cerevisiae	38.30/39.63/37.95	5,863/5,834	Eysc_h	−0.8738	−0.8444	−6.078	<0.0001
Drosophila melanogaster	42.40/53.80/63.80	22,102/22,075	Edro_h	−0.8613	−0.8318	−10.947	<0.0001
Caenorhabditis elegans	35.40/42.97/39.66	23,894/23,829	Ecel	−0.6736	−0.6430	−5.908	<0.0001

Note.—All correlations have $P < 0.0001$. The differences between each pair of *r* ($N_{c.New}$) and *r* ($N_{c.old}$) are highly significant ($P < 0.0001$) except for *Myc. genitalium*, which has relatively few genes. Overall, *r* ($N_{c.New}$) is significantly greater than *r* ($N_{c.Old}$) based on a paired-sample *t*-test on the seven pairs of *r* values ($t = 4.4926$, DF = 6, $P = 0.0041$, two-tailed test).

[a] Genomic GC%/coding sequence GC%/third codon position GC%.

[b] In format A/B where A is the total number of coding sequences (CDSs) and B is the number of CDSs after excluding those that CodonW cannot compute N_c from. GenBank accession numbers are *S. cerevisiae* (NC_001133 to NC_001148), *E. coli* (NC_010473), *B. subtilis* (NC_000964), *D. melanogaster* (NC_004353, NC_004354, NT_033777 to NT_033779, and NT_037436), and *C. elegans* (NC_003279 to NC_003284).

[c] Name of file containing known highly expressed genes distributed with EMBOSS (Rice et al. 2000). For *M. luteus* and *Myc. genitalium*, codon frequencies from ribosomal proteins are used.

[d] *T* and *P* values based on the test in Box 15.3 in Sokal and Rohlf (2012). All tests are two tailed.

against the other, and the two codons are used almost exactly equally in coding sequences of *E. coli* 536.

Given that the two synonymous codon families within each compound codon family are subject to quite different selection pressure for codon usage bias and often exhibit dramatically different codon usage bias (e.g., strong bias in the CUN codon family but no bias in the UUR codon family in *E. coli* 536), it makes little sense to lump them together in computing a single F_{CF}. Unfortunately, the original formulation of N_c (Wright 1990), as well as all subsequent implementations including the most widely used CodonW, did not separate the 6-fold or the 8-fold compound codon family into two separate codon families, but instead all lump the two codon families together into a single compound codon family and compute a single F_{CF}.

The implementation of the new N_c in DAMBE is the first to separate each 6-fold compound codon family into separate 2-fold and 4-fold codon families, and each 8-fold compound codon family into two separate 4-fold codon families. Treated in this way, the number of codon families is increased from 20 to 23 in the standard genetic code. With the extreme codon usage bias, that is, only one synonymous codon is used in each codon family, N_c will reach its minimum of 23.

Evaluation of N_c

The value of any new bioinformatic tools ultimately depends on whether they can solve biological problems better than existing ones. We will evaluate the new N_c in two ways. First, given the empirical results that CAI is the best predictor of gene expression at both mRNA and protein level (Comeron and Aguade 1998; Coghlan and Wolfe 2000), we will examine whether the new N_c correlates better with CAI than the original N_c (e.g., N_c computed by CodonW). Second, we will check whether the new N_c can predict protein production better than the original N_c.

New N_c Correlates Better with CAI Than the Original N_c

We retrieved protein-coding sequences from three eukaryotic species (*Saccharomyces cerevisiae*, *Drosophila melanogaster*, and *Caenorhabditis elegans*) and four prokaryotic species (*E. coli* representing the Gram-negative bacteria, *B. subtilis* representing the Gram-positive bacteria, *Micrococcus luteus* representing GC-rich bacteria, and *Mycoplasma genitalium* representing AT-rich bacteria). These seven species all have well-annotated genomes, and the first five also have a set of known highly expressed genes needed to compute CAI ("Ref.File" in table 1). For the last two species, the ribosomal proteins, which are typically highly expressed, are used as the set of highly expressed genes for computing CAI by the improved CAI implementation in DAMBE (Xia 2007).

The new N_c was computed by using DAMBE (with all default options that represent the method presented in this article) and the original N_c by using CodonW. CodonW cannot compute N_c for a few short genes that miss 2-fold or 4-fold codon families. These genes were excluded in computing the new N_c by DAMBE as well to facilitate a fair comparison.

The new N_c consistently exhibits stronger correlation with CAI than the original one computed by CodonW for all seven genomes, with the difference being highly significant ($P < 0.0001$) for six genomes (table 1), except for *Myc. genitalium* that has fewer genes and consequently a reduced power to detect the difference. A paired-sample *t*-test shows that the difference is highly significant ($T = 4.4926$, DF = 6, $P = 0.0041$, two-tailed test). This is not surprising because the advantage of our proposed modifications seem obvious.

The new N_c helps reveal patterns that would be hidden with the old N_c. Three genes (*yagF*, *yagG*, and *yagH*) from the defective CP 4–6 prophage of *E. coli* (Wang et al. 2010) have

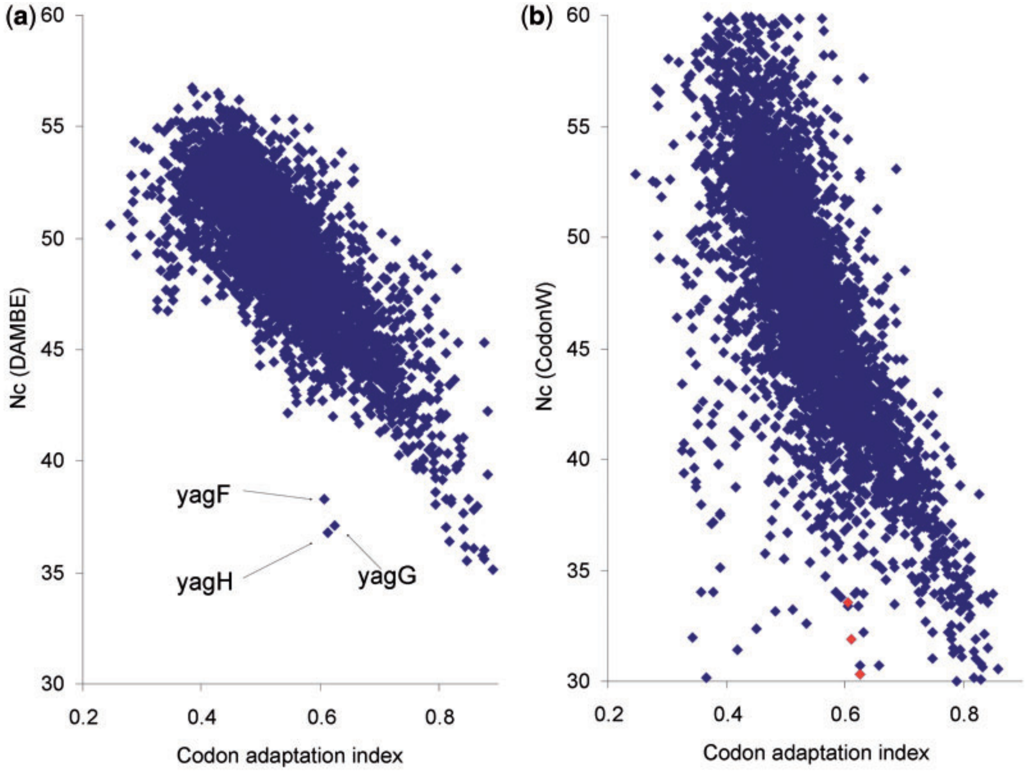

Fig. 1. The new N_c facilitates the detection of newly "immigrant" genes that exhibit codon usage bias different from the "native" genes. (*a*) Three genes (*yagF*, *yagG*, and *yagH*) from the defective CP 4–6 prophage of *Escherichia coli* (Wang et al. 2010) have strongly biased codon usage (relatively small N_c) but relatively poor codon adaptation (mediocre CAI values). (*b*) The distinction of the three genes is lost in the plot when the old N_c (computed from CodonW) is used.

strongly biased codon usage, resulting in relatively small N_c values. However, their codon usage bias is not concordant with that in highly expressed *E. coli* genes, resulting in relatively small CAI values. This codon usage pattern sets the three genes apart from the rest of *E. coli* genes (fig. 1*a*), which highlight the value of using the "N_c versus CAI" plot to detect recently horizontally transferred genes when the source genome and the target genomes have undergone codon adaptation in different directions. Interestingly, the separation of the three prophage genes from the rest of the *E. coli* genes is obscured when N_c is computed from CodonW (fig. 1*b*). The largest mucin gene (mucin 14 A) in *D. melanogaster* also exhibits strong codon usage bias ($N_c = 38.6$) but in the direction opposite to those highly expressed *D. melanogaster* genes, with a CAI value equal to 0.1277, which is the second smallest among all *D. melanogaster* genes.

The New N_c Predicts Protein Production Better Than the Original N_c

For checking whether the new N_c can predict protein production better than the original one, we used the experimentally quantified protein production in the yeast, *S. cerevisiae* (Ghaemmaghami et al. 2003). This data set, with protein abundance data for 3,850 yeast genes after excluding 18 genes that do not have a matched name in the current yeast database, can be found in the online supplemental file GhaemmaghamiProtein.xls in a previous study (Xia et al. 2011). After excluding genes that miss 2-fold or 4-fold codon families, 3,839 genes remain, and their log-transformed values (ln Prot) were correlated to CAI, the new N_c computed from DAMBE, and the original N_c from CodonW.

The new N_c correlates better with ln Prot than the original N_c, with Pearson correlation being −0.5739 between the new N_c and ln Prot and −0.5412 between the old N_c and ln Prot. The two Pearson correlation coefficients are significantly different ($z = -2.093$, $P = 0.0364$) according to the test detailed in Sokal and Rohlf (2012, pp. 573–575).

The correlation between CAI and ln Prot is 0.5981. It is highly significantly stronger than that between the old N_c and ln Prot ($z = 3.722$, $P = 0.0002$) but not significantly stronger than that between the new N_c and ln Prot ($z = 1.629$, $P = 0.1033$)

In summary, the new N_c offers four key advantages over the original N_c: 1) the minimum and maximum will now be the number of codon families and the number of sense codons, respectively, 2) biases associated with codon families with a small number of codons are alleviated by pseudocounts and by weighting, 3) compound codon families are properly handled by separating them into individual codon families, and 4) all known genetic codes are accommodated. It consistently correlates better with CAI and can predict protein production better than the original N_c.

Acknowledgments

This study was supported by the Discovery Grant from the Natural Science and Engineering Research Council of Canada to X.X., a Chinese Academy of Science innovation project (KZCX2-YW-JC104) to Q.Y., and the CAS/SAFEA International Partnership Program for Creative Research Teams.

References

Bennetzen JL, Hall BD. 1982. Codon selection in yeast. *J Biol Chem.* 257:3026–3031.

Bessho Y, Ohama T, Osawa S. 1992. Planarian mitochondria. II. The unique genetic code as deduced from cytochrome c oxidase subunit I gene sequences. *J Mol Evol.* 34:331–335.

Bulmer M. 1987. Coevolution of codon usage and transfer RNA abundance. *Nature* 325:728–730.

Bulmer M. 1991. The selection-mutation-drift theory of synonymous codon usage. *Genetics* 129:897–907.

Carullo M, Xia X. 2008. An extensive study of mutation and selection on the wobble nucleotide in tRNA anticodons in fungal mitochondrial genomes. *J Mol Evol.* 66:484–493.

Coghlan A, Wolfe KH. 2000. Relationship of codon bias to mRNA concentration and protein length in *Saccharomyces cerevisiae*. *Yeast* 16:1131–1145.

Comeron JM, Aguade M. 1998. An evaluation of measures of synonymous codon usage bias. *J Mol Evol.* 47:268–274.

Duret L, Mouchiroud D. 1999. Expression pattern and, surprisingly, gene length shape codon usage in *Caenorhabditis*, *Drosophila*, and *Arabidopsis*. *Proc Natl Acad Sci U S A.* 96:4482–4487.

Fiers W, Grosjean H. 1979. On codon usage. *Nature* 277:328.

Ghaemmaghami S, Huh WK, Bower K, Howson RW, Belle A, Dephoure N, O'Shea EK, Weissman JS. 2003. Global analysis of protein expression in yeast. *Nature* 425:737–741.

Grantham R, Gautier C, Gouy M, Jacobzone M, Mercier R. 1981. Codon catalog usage is a genome strategy modulated for gene expressivity. *Nucleic Acids Res.* 9:r43–r79.

Grosjean H, Sankoff D, Jou WM, Fiers W, Cedergren RJ. 1978. Bacteriophage MS2 RNA: a correlation between the stability of the codon:anticodon interaction and the choice of code words. *J Mol Evol.* 12:113–119.

Higgs PG, Ran W. 2008. Coevolution of codon usage and tRNA genes leads to alternative stable states of biased codon usage. *Mol Biol Evol.* 25:2279–2291.

Ikemura T. 1981. Correlation between the abundance of *Escheriachia coli* transfer RNAs and the occurrence of the respective codons in its protein genes. *J Mol Biol.* 146:1–21.

Jacob JE, Vanholme B, Van Leeuwen T, Gheysen G. 2009. A unique genetic code change in the mitochondrial genome of the parasitic nematode *Radopholus similis*. *BMC Res Notes.* 2:192.

Jia W, Higgs PG. 2008. Codon usage in mitochondrial genomes: distinguishing context-dependent mutation from translational selection. *Mol Biol Evol.* 25:339–351.

Ma P, Xia X. 2011. Factors affecting splicing strength of yeast genes. *Comp Funct Genomics.* 2011:212146.

Palidwor GA, Perkins TJ, Xia X. 2010. A general model of codon bias due to GC mutational bias. *PLoS One* 5:e13431.

Rice P, Longden I, Bleasby A. 2000. EMBOSS: the European Molecular Biology Open Software Suite. *Trends Genet.* 16:276–277.

Sharp PM, Li WH. 1987. The Codon Adaptation Index—a measure of directional synonymous codon usage bias, and its potential applications. *Nucleic Acids Res.* 15:1281–1295.

Sharp PM, Tuohy TM, Mosurski KR. 1986. Codon usage in yeast: cluster analysis clearly differentiates highly and lowly expressed genes. *Nucleic Acids Res.* 14:5125–5143.

Sokal RR, Rohlf FJ. 2012. Biometry, 4th ed. New York: Freeman.

Telford MJ, Herniou EA, Russell RB, Littlewood DT. 2000. Changes in mitochondrial genetic codes as phylogenetic characters: two examples from the flatworms. *Proc Natl Acad Sci U S A.* 97:11359–11364.

van Weringh A, Ragonnet-Cronin M, Pranckeviciene E, Pavon-Eternod M, Kleiman L, Xia X. 2011. HIV-1 modulates the tRNA pool to improve translation efficiency. *Mol Biol Evol.* 28:1827–1834.

Wang X, Kim Y, Ma Q, Hong SH, Pokusaeva K, Sturino JM, Wood TK. 2010. Cryptic prophages help bacteria cope with adverse environments. *Nat Commun.* 1:147.

Wright F. 1990. The "effective number of codons" used in a gene. *Gene* 87:23–29.

Xia X. 1996. Maximizing transcription efficiency causes codon usage bias. *Genetics* 144:1309–1320.

Xia X. 1998. How optimized is the translational machinery in *Escherichia coli*, *Salmonella typhimurium* and *Saccharomyces cerevisiae*? *Genetics* 149:37–44.

Xia X. 2005. Mutation and selection on the anticodon of tRNA genes in vertebrate mitochondrial genomes. *Gene* 345:13–20.

Xia X. 2007. An improved implementation of Codon Adaptation Index. *Evol Bioinform.* 3:53–58.

Xia X. 2008. The cost of wobble translation in fungal mitochondrial genomes: integration of two traditional hypotheses. *BMC Evol Biol.* 8:211.

Xia X, MacKay V, Yao X, Wu J, Miura F, Ito T, Morris DR. 2011. Translation initiation: a regulatory role for poly(A) tracts in front of the AUG codon in *Saccharomyces cerevisiae*. *Genetics* 189:469–478.