SUPPLEMENTAL MATERIAL

There are five sections in this supplemental material. The first section, "SCUMBLE", contains additional details/discussions about our algorithm and its properties. The next two sections, "Artifacts 1" and "Artifacts 2", discuss particular weaknesses of the algorithms we have compared SCUMBLE to. The fourth section concerns weaknesses of the specific implementation of CA/RSCU in the much-used program CodonW. The last section, "More prokaryote results", contains a number of results from SCUMBLE that did not make the main paper.

SCUMBLE

There are two distinct levels of the MLE algorithm: Given a set of preference functions, it is easy to find the optimal offsets, as this can be done gene by gene. Finding optimal preference functions, however, is difficult. SCUMBLE starts with a randomly generated preference function—as mentioned in the main text, only one preference function is added at a time—and uses a combination of gradient ascent and Newton's method to find the optimum; at every step of this algorithm the new optimal offsets are calculated. The current implementation typically requires from 10 minutes to an hour to estimate the preference functions for a prokaryote genome for models with up to 10 trends, on a single regular processor. If only models with a few trends are desired, it is much faster. Once the preference functions have been estimated, the optimal offsets can be calculated for any gene, whether or not it was in the gene set used to estimate the preference functions.

SCUMBLE searches for a local maximum of the likelihood function, which is not guaranteed to be the global maximum. Indeed, different runs of the algorithm may produce different results (differing by more than a trivial change of sign), but this seems to only happen when there is little or no bias in the first place. We have never seen different results from replicate runs in the presence of strong bias [for the trend(s) corresponding to the strong bias].

When applied to unbiased random data, our algorithm tends to pick preference functions corresponding to individual codons. This can be used to limit the number of trends worth considering: Once a preference function (together with the preceding preference functions) matches the ideal preference function for a single codon, this indicates that there are no biases left in the data that are strong enough to overcome the very slight natural preference of the algorithm to pick such preference functions.

There is no guarantee that the individual trends found by our algorithm will correspond directly to separate biases. However, even if this is not the case, the trends describe the directions in which the genome shows the greatest variance in a way that is far more statistically accurate than PCA. When the biases do separate well, the preference functions for the weaker biases are constrained to be orthogonal to those of the stronger biases. This is well illustrated by *B. subtilis*: From Fig. 7, it is clear that the real expression level bias contains a fair bit of GC bias, which is absorbed into the first offset. Thus, the genes of different expression levels resemble a slanted line in that plot.

Our approach avoids a bias inherent in CAI: In CAI, the different amino acids have very different contributions to the index, thus proteins with different compositions will tend to have different CAIs, even if the proteins' relative synonymous codon usage (RSCU) values are identical for the different proteins. For instance, using directly the frequencies from the CAI, a "protein" with only Leucine residues will have a CAI of 0.68, while a "protein" with only Glutamine will have a CAI of > 0.96, even though both have synonymous codon usage exactly matching the reference set of highly expressed genes. Our algorithm, on the other hand, will on average assign the same betas regardless of the amino-acid composition; only the statistical accuracy is affected by the composition.

On the other hand, while it is not an optimal estimate of the degree of codon bias in a gene, the CAI is likely an as good or better predictor of how well a protein can be expressed in an organism, as our offsets. For this purpose, the above-mentioned biases related to amino acid composition are real effects: For some amino acids it doesn't really matter what codon you use, while for others the right choice is critical.

Contrary to selection pressures, changes in mutational rates do not naturally lead to exponential changes in the relative frequencies of synonymous codons, thus the offsets will not be linearly related to the mutational pressures. However, if the mutations act independently on the nucleotides in a codon, the exponential form allows us to decompose the probability of the codon into the probabilities of the nucleotides—if the probability of a codon is the product of the probabilities of the nucleotides in that codon, that corresponds directly to the preference $E_i(c)$ being the sum of preferences for each nucleotide; the offset will be identical for all 3 nucleotides in a codon. This guarantees that as long as a mutational pressure only differentiates between two groups of nucleotides (e.g. AT vs. CG, or A vs. CGT), we can model arbitrarily strong mutational pressures with a single trend, with no distortion of the preference function. General single-nucleotide mutational pressures may require up to 3 trends, while if there are nearest neighbor correlations, we can not expect to model them precisely with only a few trends. Of course, weak mutational pressures can always be modelled accurately with a single trend.

Figure S6 shows β_1 for yeast plotted against the number of codons in a gene for the real genome and for a

randomized, unbiased genome. The accuracy of the estimated offset is better the longer the gene is.

SCUMBLE has the same chance to yield either sign for a preference function during each run; Supplemental Tables S1 and S2 contain the signs directly given by SCUMBLE. In the main text, we have when necessary changed the signs of offsets and their preference functions to yield positive correlations with expression/GC/GT/CT content.

Artifacts 1: Amino acid frequencies

Methods

To demonstrate certain weaknesses of the different correspondence analysis methods, we generated several sets of data that share the same synonymous codon usage probabilities, but have different amino acid distributions. We used the yeast genome as a template, i.e., we used that number of genes with their respective lengths. For each gene we first randomly assigned an amino acid to each position, according to the desired distribution. We then assigned codons to each amino acid using our two-trend model for the yeast genome, using either the offsets estimated for that gene in yeast by SCUMBLE or—if a weaker bias was desired—some constant fraction thereof. For our "unbiased" model we use uniform distribution of all codons, i.e., the expected frequency of an amino acid is proportional to the number of codons encoding that amino acid.

Results

Correspondence analysis on relative synonymous codon usage (CA/RSCU) values has long been one of the most popular approaches to analysis of codon usage, but this approach is known to be prone to artifacts [1]. Due to the lack of proper statistical weighting, noise from rare amino acids can dominate actual signals, as shown in Fig. S9. It also does not consider the length of a gene, and thus a weak but clear signal in long genes might be overwhelmed by noise from short genes that are given too high statistical weight.

Within-block correspondence analysis (WCA) has been suggested as a better method [2], although it has not yet been widely used for analysis of codon usage. However, although it solves those specific problems exchibited by CA/RSCU, also WCA is at some risk for artifacts and other problems. First, although WCA is not improperly sensitive to the overall amino acid distribution the way CA/RSCU is, it is somewhat sensitive to correlations between gene length and amino acid distribution: amino acids that are present at a higher rate in long genes will on average get too high statistical

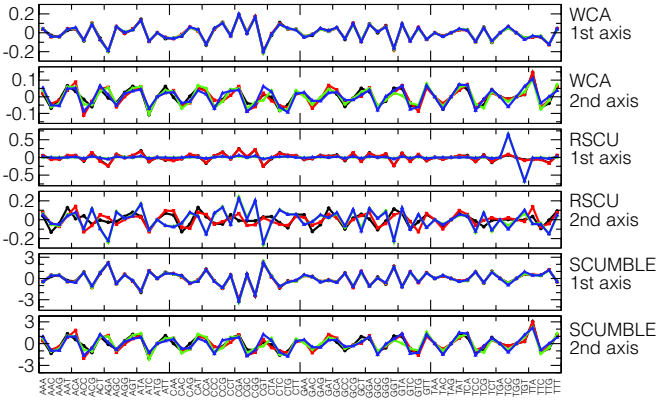

FIG. S9: Artifacts from rare amino acids. The codon weights for the first and second axes, determined by each method, are shown. Black (circles) and red (squares): Unbiased amino acid distribution. Green (diamonds) and blue (triangles): Cystein is only 1/10th as common (per codon) as the other amino acids. Synonymous codon biases are weak (1/5th of strength in yeast). The results from CA/RSCU are dominated by the artifact for cystein (2nd and 3rd panel), while WCA and SCUMBLE are largely unaffected by the rare amino acid.

weight, while amino acids that primarily are present in short genes get too low weight (Fig. S10). As the ratio of hydrophilic to hydrophobic residues tends to depend on the protein (domain) size, this effect could distort estimates of synonymous codon usage bias in real organisms.

Second, while correspondence analysis is often said to produce axes that are orthogonal to each other, they are in reality only orthogonal in the space in which the eigenvalue decomposition in CA takes place. For WCA, the transformation between this space and synonymous-codon-usage-probability space depends on the amino acid distribution, thus the a.a. distribution can indirectly affect the directions of at least all but the first axis (Fig. S11). While this is a weak effect, it is independent of the signal to noise ratio, i.e., it can be relevant even in the presence of very strong synonymous codon bias.

Artifacts 2: Nonlinearities

Often, only a small fraction of the genes are subject to a given bias to a significant extent. We simulated this situation using our probabilistic model for a pure GC bias (this gives exactly the same RSCU values as a randomly generated nucleotide sequence with different GC fractions). Figure S12 shows results for a synthetic genome where 90% of the genes have the same expected codon usage, with a high GC fraction, while 10% of the genes have lower, varying GC fractions but otherwise the same expected codon usage.

Both WCA and CA/RSCU show a clear nonlinear relationship between the 1st and 2nd axis. The codon

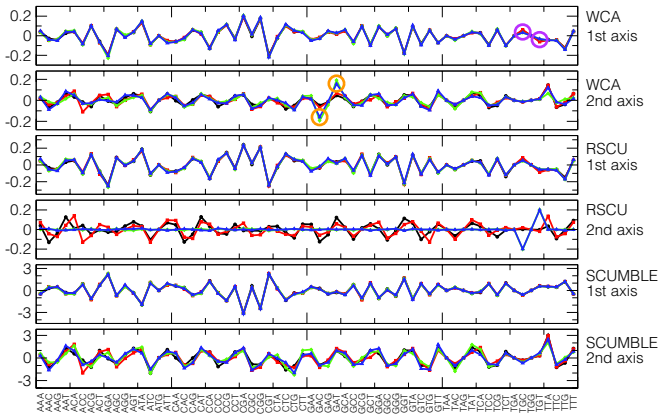

FIG. S10: Artifacts from length-dependent amino acid frequencies. The codon weights for the first and second axes, determined by each method, are shown. Black (circles) and red (squares): Unbiased amino acid distribution. Green (diamonds) and blue (triangles): The frequency of aspartic acid is negatively correlated with gene length, while the frequency of cystein is positively correlated with gene length. Both correlations are strong but reasonable. Synonymous codon biases are weak (1/5th of strength in yeast). For WCA, the weights for cysteine along the first axis change significantly (top panel, circled in magenta), and aspartic acid shows up as a clear artifact for the second axis (2nd panel, circled in orange). For CA/RSCU, the second axis for the biased distribution is entirely an artifact dominated by cysteine (4th panel)—although cysteine is not overall rare, it is rare in short genes, which cause the greatest statistical fluctuations. The results from SCUMBLE are unaffected by the amino acid distribution (bottom two panels); there are no systematic differences. All these results are highly reproducible.

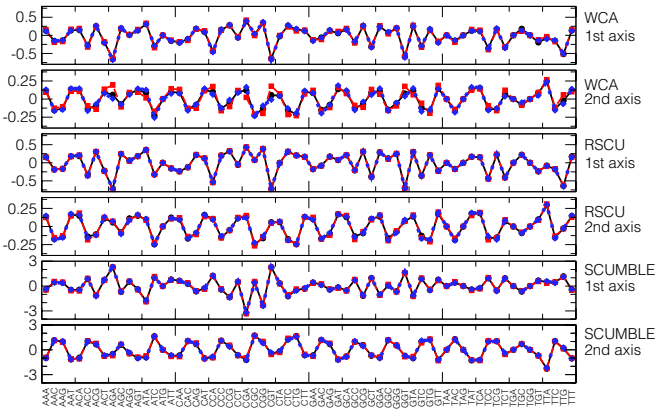

FIG. S11: Artifacts from amino acid-dependent orthogonality. The codon weights for the first and second axes, determined by each method, are shown. Black (circles, solid line): Unbiased amino acid distribution. Red (squares, dashed line): A selected subset of the amino acids are 3-4 times as common as the others (per codon). Blue (diamonds, dotted line): A different selected subset of the amino acids are 3-4 times as common as the others. Synonymous codon biases are as strong as in yeast; statistical fluctuations are negligible. For WCA, the second axis differs due to different requirements for orthogonality (2nd panel). Both CA/RSCU and SCUMBLE give essentially identical results for all three amino acid distributions.

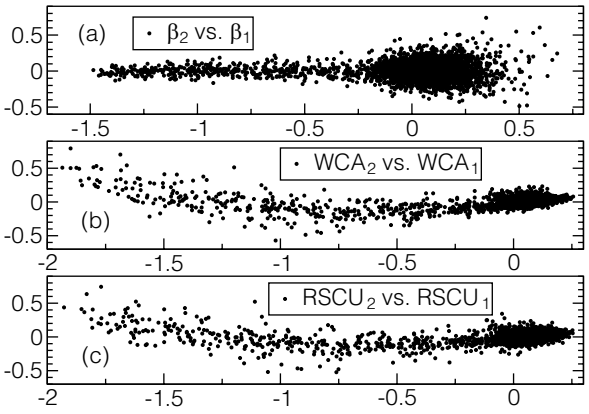

FIG. S12: Synthetic genome where 90% of the genes have identical, high-GC codon usage, while 10% have varying GC bias. The position of each gene on the first two axes are plotted for each algorithm. In all cases, the first axis corresponds to GC content.

weights (the position of a codon on a given axis) for the 2nd axis are shown in Fig. S13(b), and are clearly dominated by two codons, namely AGA (arginine) and TTA/UUA (Leucine). These are precisely the two codons that have *two* fewer G/Cs than some of their synonymous codons, and they will thus experience a stronger initial GC bias, but will saturate earlier than other GC-poor codons. SCUMBLE shows no nonlinear relationship between β_1 and β_2, as these nonlinearities are captured by the probabilistic model.

The codon weights from WCA and CA/RSCU for the 1st axis are distorted by these nonlinearities. For a weak GC bias, WCA, CA/RSCU and SCUMBLE all yield similar preference functions/codon weights [after rescaling WCA and CA/RSCU; Fig. S13(c)], but these are clearly different from the codon weights from WCA and CA/RSCU for the strong bias, above. SCUMBLE, however, yields almost identical preference functions for strong and weak bias.

In WCA and CA/RSCU, deviations from average codon usage is scaled according to that average codon usage. However, if there is a large deviation along one axis (e.g. GC bias), that will affect the expected deviations in other directions, and WCA and CA/RSCU do not compensate for this. This is illustrated in Fig. S14: the standard deviation for WCA_2 or $RSCU_2$ is more than twice as large for genes left of the dashed red line as for the main bulk of genes to the right of the dashed line. This effect can be much larger for biases primarily involving amino acids with only 2 synonymous codons, as these are far more constrained by GC bias.

All of these artifacts can be observed for the genome of *Pseudomonas aeruginosa* (Fig. S15. To make sure that

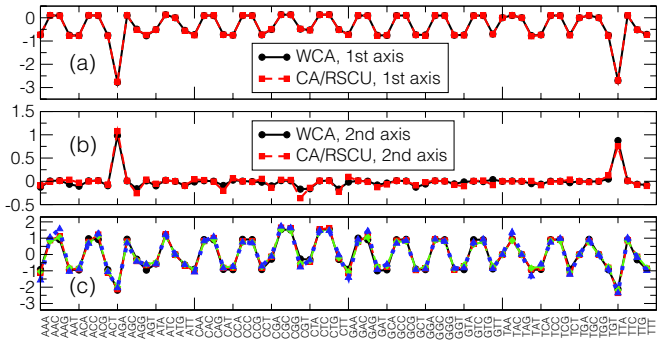

FIG. S13: (a), (b): Synthetic genome where 90% of the genes have identical, high-GC codon usage, while 10% have varying GC bias. The position of each codon on the first and second axes are plotted for WCA and RSCU. (c) Synthetic genome where 90% of the genes have uniform codon usage, while 10% have random, weak GC bias. The position of each codon on the first axis is plotted for all three methods, as are the results from SCUMBLE for the data used in (a) and (b).

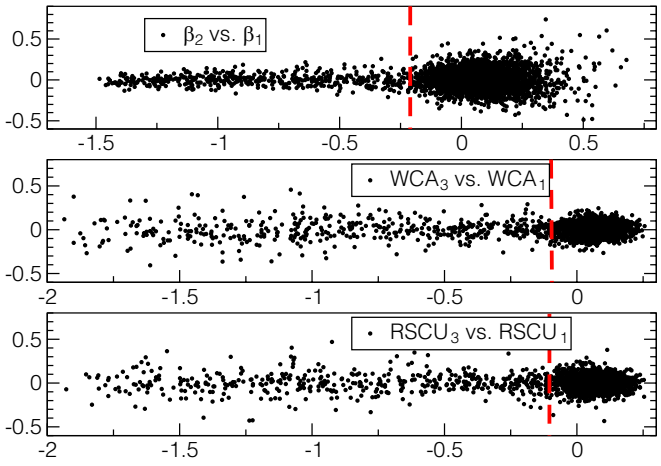

FIG. S14: Synthetic genome where 90% of the genes have high-GC codon usage, 10% have varying GC bias, and all genes have a small secondary bias (random direction in codon space). The position of each gene on the first two axes are plotted for each algorithm. In all cases, the first axis corresponds to GC content. The same number of genes are to the right of the dashed red line for each algorithm.

these were indeed artifacts, and the result of some bias that SCUMBLE failed to detect, we generated a randomized version of the *P. aeruginosa* genome using the 2-trend model from SCUMBLE. The results were very similar when we applied WCA to this randomized genome, except that the axes were not in the exact same order (data not shown). Results are very similar for RSCU as for WCA. SCUMBLEdoes not display these artifacts. Furthermore, the probabilistic model makes it easy to test whether a trend is indeed an artifact in SCUMBLE, by checking if it appears also in randomly generated data (with that trend absent from the model used to generate

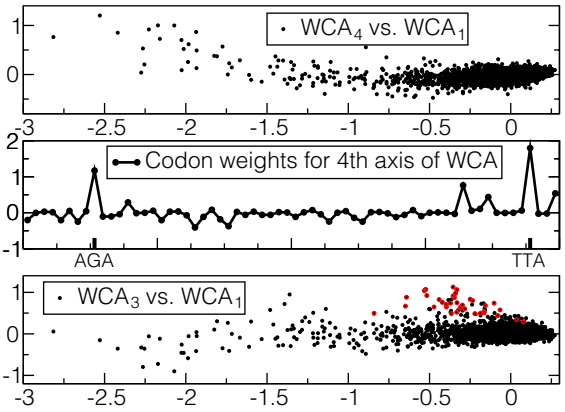

FIG. S15: Results for WCA applied to the *Pseudomonas aeruginosa* genome. (a) The 4th axis shows a clear nonlinear relationship with the GC content, which is captured by the 1st axis. (b) The codon weights of the 4th axis are dominated by AGA (9th codon) and TTA (61st codon). (c) For the 3rd axis, the standard deviation is much larger for genes with low GC content than for the bulk of the genes (ribosomal proteins have high values because this axis captures the expression bias).

the data).

CA/RSCU in CodonW: Analysis of synonymous codon usage?

The algorithm most commonly used to analyze codon usage is correspondence analysis on relative synonymous codon usage (CA/RSCU), as implemented in the program CodonW (available from URL http://codonw.sourceforge.net/culong.html). There are, however, two problems with the algorithm used in CodonW.

First, CodonW includes the first codon of a gene in its analysis. However, many prokaryotes use alternative start codons, such as UUG, which in the context of a start codon codes for methionine (or rather N-formylmethionine). CodonW mistakenly treats this codon as a codon for leucine. SCUMBLE, as well as our implementations of WCA and CA/RSCU, ignore the start codon.

Second, when a gene does not contain a given amino acid, CodonW sets the RSCU values for the corresponding codons to zero. This, however, means that the RSCU values contain information on whether or not a gene contains a given amino acid, which means the results can depend (in a very crude way) on the amino acid content of genes. To eliminate this problem in our implementation of CA/RSCU, and ensure that the results depend only on synonymous codon usage, we set the RSCU values for codons encoding a missing amino acid to the averages of the RSCU values for the same codons in all genes that do

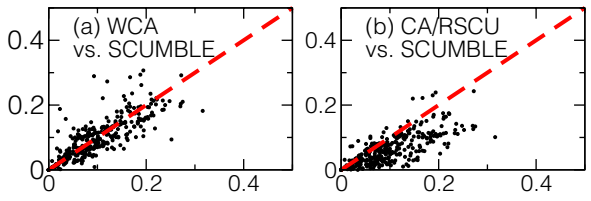

FIG. S16: Maximal square correlations between β_i / WCA$_i$ / RSCU$_i$ and a ribosomal indicator (1 for identified ribosomal genes, 0 for all other genes). The correlations for WCA [(a) and (c)] or CA/RSCU [(b) and (d)] are plotted against the correlations for SCUMBLE for all the prokaryote genomes studied.

contain the amino acid. This ensures that missing amino acids contribute to neither the eigenvalue calculations nor the gene scores.

More prokaryote results

There are several different ways to compare the results from two different algorithms. If we were to compare correlations between β_i / WCA$_i$ / RSCU$_i$ and the genes' GC3 or GT3 values, then SCUMBLE would be at a clear disadvantage, since the nonlinear relationships between the βs and codon usage frequencies in SCUMBLE would yield lower correlations than the linear relationships of WCA and CA/RSCU, even if the underlying biases were identical. When comparing preference functions/codon weights, however, none of the algorithms should be at significant disadvantage. Figure S8 shows that not only does SCUMBLE perform better than WCA and CA/RSCU on average, but there are barely any genomes for which WCA or CA/RSCU outperforms SCUMBLE. The performance difference is clearly greater for GT content than for GC content, suggesting that SCUMBLE is relatively better at uncovering weaker biases. In both cases, CA/RSCU is the weakest method of the three.

To evaluate the algorithms' performance on discovering expression-related codon bias, we compare the correlations between β_i / WCA$_i$ / RSCU$_i$ and the ribosomal indicator, which is 1 for ribosomal genes and 0 for all other genes. As shown in Fig. S16, SCUMBLE again beats CA/RSCU easily, but there is no clear winner between SCUMBLE and WCA; each method outperforms the other significantly for a number of genomes (the average correlation is slightly higher for SCUMBLE).

Figure S17 shows the results for *Bacillus subtilis*. The first trend for *B. subtilis* corresponds to GC bias [Fig. S17(a)]: the square correlation between β_1 and the GC3 value is 0.889, and the preference function correlation is $r_{\rm P}^2(E_1, E_{\rm GC}) = 0.760$. The second trend corresponds to expression level [Fig. S17(b)]: Ribosomal genes have average β_2 values of 1.33, whereas the genomic average is zero. This is only about half the strength of ribosomal gene bias in yeast, but is among the stronger biases for ribosomal genes in prokaryotes, although extremely fast-growing prokaryotes such as *Clostridium perfringens* and *Vibrio parahaemolyticus* have significantly stronger biases (Table S1). β_3 corresponds well to GT bias [$r_{\rm P}^2(E_3, E_{\rm GT}) = 0.769$] and exhibits significant strand asymmetry [Fig. S17(c,d)]. These observations agree with previous studies of codon usage in *B. subtilis* and of the strength of selected codon usage bias in various bacteria [8].

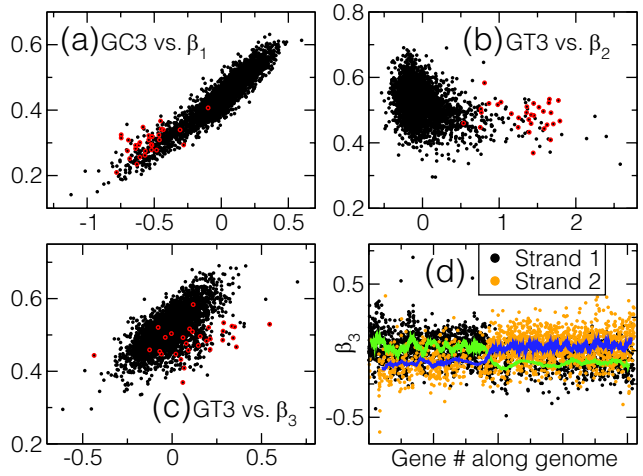

FIG. S17: A 4-trend model of *Bacillus subtilis*. (a)-(c) GC3 or GT3 plotted against the first, second and third offset for each gene. Genes for ribosomal proteins are circled in red. (d) β_3 plotted against the number of the gene along the genome, with genes on different strands in different colors. The green and blue lines are 50-point running averages for strand 1 and 2, respectively.

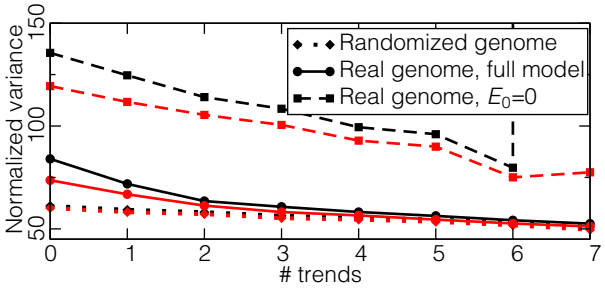

FIG. S18: Average (black) and median (red) normalized variance for named genes in *B. subtilis*, for models with up to 10 trends. Full models (circles, solid lines) and models with E_0 set to zero (squares, dashed lines) are compared to results for a randomized genome (diamonds, dotted lines).

As in budding yeast, the two first trends explain most of the excess variance. However, unlike for budding yeast, the models for *B. subtilis* with a moderate number of trends can not explain the overall codon bias of the genome when E_0 is set to zero (Fig. S18).

Borrelia burgdorferi has long been a prime example of

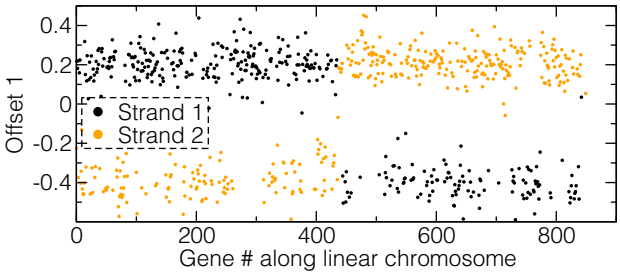

FIG. S19: A 2-trend model of *B. burgdorferi*: the first trend plotted against the gene number along the genome, with genes on different strands in different colors.

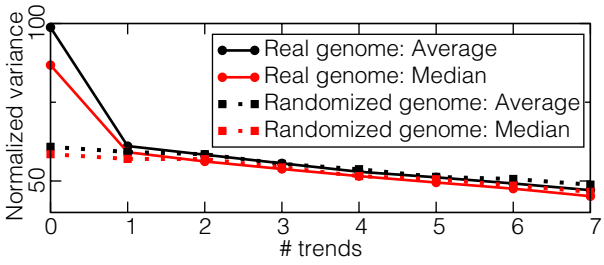

FIG. S20: Average and median normalized variance for models with up to 7 trends for *B. burgdorferi*.

a prokaryote with very strong strand asymmetry of codon usage [3]. The results from SCUMBLE support this: the first offset almost perfectly separates the genes on the leading and lagging strands (Supplemental Fig. S19), and the preference function E_1 is highly correlated with E_{GT}—$r^2_P(E_1, E_{GT}) = 0.856$. Even more significantly, the first trend seems to explain essentially all the codon usage variation in the *B. burgdorferi* genome: The excess variation for the model with 1 trend is reduced by 95% (leaving it on the order of the sampling error), and there is no evidence for excess variation for the models with 2 or more trends (Supplemental Fig. S20). The absence of competing biases would explain why it for *B. burgdorferi* is exceptionally easy to distinguish genes on the leading and lagging strands by their sequence [4].

Burkholderia mallei has a genomic GC content of 68% and an average GC3 value of 87%. Recently, it was claimed that the main source of codon usage variation in this organism is translational selection, and that all translationally optimal codons end in C or G [5]. To test this claim, we applied SCUMBLE to the *B. mallei* genome. The first temperature corresponds very well to GC content—$r^2_P(E_1, E_{GC}) = 0.828$—suggesting that the main source of codon usage variation is mutational bias, not translational selection. The second temperature, on the other hand, is completely unrelated to GC content, but has high values for all the ribosomal proteins, indicating that it corresponds to expression level (Fig. S21). CA/RSCU, on the other hand, yields little to no signal for the ribosomal proteins.

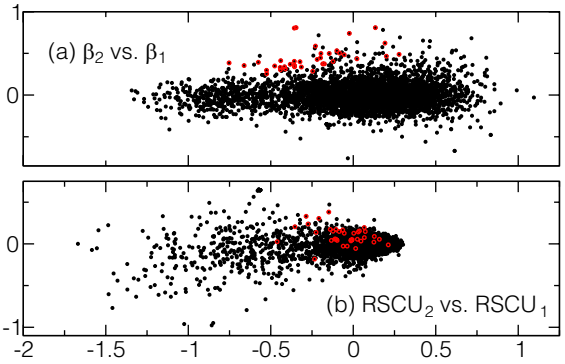

FIG. S21: Scatter plot of the first two axes from the two-trend model found by SCUMBLE (a) and CA/RSCU (b) for the genes of *Burkholderia mallei*. Genes for ribosomal proteins are circled in red.

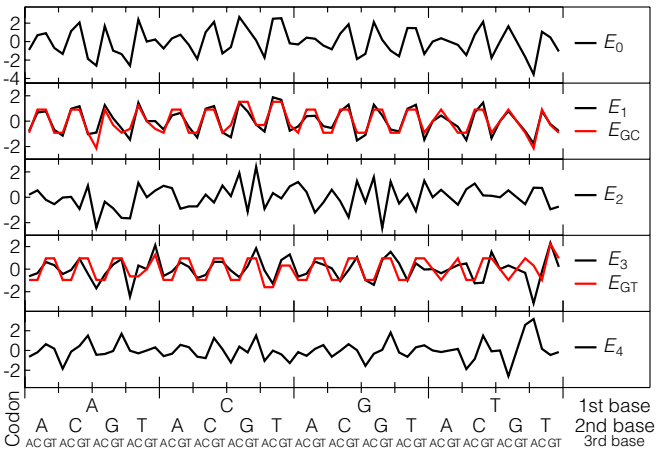

FIG. S22: Preference functions for the 4-trend model of *Burkholderia mallei*. 'Ideal' preference functions are shown in red.

Looking at the energies E_2, we find that many of the favored codons do not end in C or G (Fig. S22)—for instance, GAA is strongly favored over GAG for glutamate. Indeed, the ribosomal proteins have lower GC3 values than most proteins in *B. mallei*. We attribute the discrepancy with prior results to the prior authors' reliance on CAI—defined by ribosomal proteins—to estimate expression level: Even though C/G-ending codons are often not translationally optimal, many of them are nevertheless the most common codons in ribosomal proteins, due to the strong overall GC bias. CAI thus mistakenly identifies proteins with high GC3 values as highly expressed proteins. This problem has been noted before [7], although the issue is more clear in [5] than in the article discussed there, [6]: In [5], it is clear that the CAI was appropriately defined, using ribosomal genes (which are indeed highly expressed), but CAI is still a poor estimator for gene expression level.

While the ribosomal bias in *A. dehalogenans* is clear, we found a curious feature: While several of the genes ex-

pected to be highly expressed, such as chaperonin GroEL and translation EF-G, show similar bias as the ribosomal genes, translation EF-Tu has a highly significant opposite bias ($p < 10^{-6}$): the two near-identical genes have the 4th and 5th lowest values of β_2. This is quite unusual: in most other prokaryotes examined, EF-Tu shares the bias of the ribosomal genes.

[1] Perrière, G. and Thioulouse, J. (2002) Use and misuse of correspondence analysis in codon usage studies. *Nucleic Acids Res.*, **30**, 4548–4555.

[2] Charif, D., Thioulouse, J., Lobry, J.R. and Perriére, G. (2004) Online synonymous codon usage analyses with the ade4 and sequinR packages. *Bioinformatics*, **21**, 545–547.

[3] McInerney, J.O. (1998) Replicational and transcriptional selection on codon usage in *Borrelia burgdorferi*. *Proc. Nat. Acad. Sci.*, **95**, 10698–10703.

[4] Rocha, E.P.C., Danchin, A. and Viari, A. (1999) Universal replication biases in bacteria. *Mol. Microbiol.*, **32**, 11–16.

[5] Zhao, S., Zhang, Q., Chen, Z., Zhao, Y. and Zhong, J. (2007) The factors shaping synonymous codon usage in the genome of *Burkholderia mallei*. *J. Gen. Gen.*, **34**, 362–372.

[6] Gupta, S.K. and Ghosh, T. C. (2001) Gene expressivity is the main factor in dictating the codon usage variation among the genes in Pseudomonas aeruginosa. *Gene*, **273**, 63–70.

[7] Grocock, R.J. and Sharp, P.M. (2002) Synonymous codon usage in *Pseudomonas aeruginosa* PA01. *Gene*, **289**, 131–139.

[8] Sharp, P.M., Bailes, E., Grocock, R.J., Peden, J.F. and Sockett, R.E. (2005) Variation in the strength of selected codon usage bias among bacteria. **33**, 1141–1153.