Molecular BioSystems

Dynamic Article Links

Cite this: *Mol. BioSyst.*, 2012, **8**, 1396–1404

www.rsc.org/molecularbiosystems

METHOD

Can simple codon pair usage predict protein–protein interaction?†

Yuan Zhou,‡[a] Ying-Si Zhou,‡[a] Fei He,§[a] Jiangning Song*[bc] and Ziding Zhang*[a]

Received 17th October 2011, Accepted 12th February 2012
DOI: 10.1039/c2mb05427b

Deciphering functional interactions between proteins is one of the great challenges in biology. Sequence-based homology-free encoding schemes have been increasingly applied to develop promising protein–protein interaction (PPI) predictors by means of statistical or machine learning methods. Here we analyze the relationship between codon pair usage and PPIs in yeast. We show that codon pair usage of interacting protein pairs differs significantly from randomly expected. This motivates the development of a novel approach for predicting PPIs, with codon pair frequency difference as input to a Support Vector Machine predictor, termed as CCPPI. 10-fold cross-validation tests based on yeast PPI datasets with balanced positive-to-negative ratios indicate that CCPPI performs better than other sequence-based encoding schemes. Moreover, it ranks the best when tested on an unbalanced large-scale dataset. Although CCPPI is subjected to high false positive rates like many PPI predictors, statistical analyses of the predicted true positives confirm that the success of CCPPI is partly ascribed to its capability to capture proteomic co-expression and functional similarities between interacting protein pairs. Our findings suggest that codon pairs of interacting protein pairs evolve in a coordinated manner and consequently they provide additional information beyond amino acids-based encoding schemes. CCPPI has been made freely available at: http://protein.cau.edu.cn/ccppi.

Introduction

Protein–protein interactions (PPIs) provide important insights into protein function and cell organization.[1] Using high-throughput experimental techniques like yeast two-hybrid screening[2] and tandem-affinity purification coupled with mass spectrometry,[3] miniatures of the interactomes of a few model organisms have been revealed so far. However, these experimental methods are relatively expensive and labor intensive, while suffering from insufficient coverage. Consequently, it is greatly desired to develop computational approaches to predict PPIs.[4] One of the most validated PPI prediction methods is interolog-based, which transfers interaction annotation from a protein pair in a species to the orthologous protein pairs in other species.[5,6] Nevertheless, as this method relies on interaction data from related organisms, it does not perform well in distal organisms. To address this, a common complementary method was developed based on domain–domain interactions,[7,8] which relies on known interacting domains[9,10] and has a higher false positive rate.[11] Other methods usually take advantage of the observed evolutionary or functional relationship of interacting proteins to predict PPIs. For example, the phylogenetic profile methods predict PPIs among co-occurring protein pairs from different genomes;[12] the co-expression approaches use the expression profile similarity to detect PPIs;[13,14] while function-based methods examine the functional similarity of a query protein pair to judge whether they interact or not.[15] In general, these kinds of methods can discover functionally associated protein pairs, but not necessarily the physical interactions between proteins.[8,16]

The strong dependency on evolutionary or functional information of the aforementioned methods has led to a plethora of simple sequence-based PPI prediction methods. These methods aim to predict physical PPIs based on short sequence unit frequency,[17–21] rather than homology. For example, the conjoint triad (CT) encoding scheme[20] proposed by Shen *et al.* based on the calculation of tri-peptide frequencies was shown to achieve good results in the human PPI dataset. Using auto covariance (AC) of physicochemical features derived from spaced amino acid pairs, Guo *et al.*[18] achieved acceptable performance on the yeast (*Saccharomyces cerevisiae*) dataset. The related PPI prediction methods and reported performance are summarized in Table S1 (ESI†). Most of these sequence-based studies exploited amino acid-centric encoding schemes to develop PPI predictors, as proteins rather than DNAs or RNAs are the primary components of PPIs.

[a] *State Key Laboratory of Agrobiotechnology, College of Biological Sciences, China Agricultural University, Beijing 100193, China. E-mail: zidingzhang@cau.edu.cn*
[b] *National Engineering Laboratory for Industrial Enzymes and Key Laboratory of Systems Microbial Biotechnology, Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences, Tianjin 300308, China. E-mail: jiangning.song@monash.edu*
[c] *Department of Biochemistry and Molecular Biology, Faculty of Medicine, Monash University, Melbourne, VIC 3800, Australia*
† Electronic supplementary information (ESI) available: Tables S1–S6 and Fig. S1–S7. See DOI: 10.1039/c2mb05427b
‡ These authors contributed equally to the work.
§ Current address: Developmental Biology Program, Sloan-Kettering Institute, New York, NY 10065, USA.

The central dogma of molecular biology[22] defines the direction of bio-information flow, but does not promise perfect transferring efficiency or identity. It is well established that the usage of synonymous codons is a factor that affects the expression level of proteins in microorganisms and is inter-species biased.[23] This bias has been shown to correlate with the abundance of different tRNA species,[24] and exerts a strong influence over translation rates in various organisms.[25] Furthermore, synonymous codon usage has also been suggested to influence protein folding under certain circumstances. For example, synonymous substitutions of rare codons into more frequent codons in a fatty acid binding protein expressed in *Escherichia coli* could induce its misfolding.[26] Such phenomena are interpreted as a reflection of the translation rate control that facilitates co-translational protein folding.[27] Moreover, a recent large-scale study has identified synonymous codons that have appreciable preferences towards different secondary structure types and different residue positions in protein structures, which are significantly distinct from the amino acids they encode.[28] Taken together, these results suggest that synonymous codons contain important information that is not represented by amino acid sequences. Similar to codon usage, codon pair usage is also biased,[29] thereby influencing translation efficiency[30] and fidelity.[31] Recently, virus attenuation resulting from the alteration of codon pair usage of poliovirus capsid protein without the change of codon usage was reported.[32] This indicates that codon pair usage could carry information different from that carried by codon usage.

Physically interacting or functionally associated protein pairs have been recently demonstrated to have similar codon usage bias.[33,34] Based on large-scale datasets, Najafabadi and Salavati showed that codon frequency, as an indicator of coding sequence co-evolution among interacting protein pairs,[35] can be effectively used to predict protein interactions (see Table S1, ESI†) in a recent report.[19] Inspired by their pioneering work, in this study, we take a further step to investigate the ability of codon pair usage to predict PPIs. Our analyses show that codon pair usage of interacting protein pairs is also significantly different from that of random protein pairs. We consequently build a codon pair usage-based PPI prediction method termed as CCPPI (Codon Combination-based Protein–Protein Interaction predictor) under the Support Vector Machine (SVM) framework. We show that the performance of CCPPI compares favorably with several popular sequence-based encoding schemes through extensive benchmark tests. We provide possible explanations for why CCPPI can predict PPIs by comparing and analyzing the predicted true positives resulting from different encoding schemes. Moreover, we also discuss the applicability of CCPPI to the prediction of the fruit fly (*Drosophila melanogaster*) interactome. Our CCPPI approach, when integrated with traditional prediction methods, is anticipated to be further useful for improving the performance and coverage of PPI prediction.

Materials and methods

Sequence encoding schemes

Our analyses were mainly based on the yeast interactome data, which were believed to be of relatively high coverage.[36,37] We exploited features from protein sequences or gene coding sequences. Protein sequences and the corresponding coding sequences of yeast were downloaded from SGD database (http://www.yeastgenome.org/). The difference in a feature[19] between a pair of proteins can be simply calculated as:

$$d(x) = Z \times |f_i(x) - f_j(x)| \quad (1)$$

where $f_i(x)$ and $f_j(x)$ stand for the values of feature x of proteins i and j, respectively. For codon pair frequencies, $f(x)$ is the total number of the codon pair x in the coding sequence of a protein divided by the length of the protein. Z is a scaling factor used to avoid a feature of small quantity, which was set to 100. The frequencies of amino acids, amino acid pairs and codons were calculated in the same manner.

A simple extension of the above $f(x)$ is the inverted distance weighting. To calculate the inverted distance weighted (IDW) frequency of a codon pair x that is composed of two codons p and q, we summed up the inverted linear distance between the codons p and q. Then, $f(x)$ was modified as:

$$f(x) = Z \times \frac{\sum_{k=1}^{m(x)} \frac{1}{\mathrm{dst}(pq)+1}}{n} \quad (2)$$

where $m(x)$ is the total number of codon pairs x, dst(pq) is the linear distance between codons p and q in the corresponding protein sequence; Z and n are the scaling factor and protein length, respectively. For example, $f(AAATTT)$ of the sequence "AAACCCGGGAAATTT" is calculated as $100 \times (1/4 + 1/1)/5 = 25$. To simplify the calculation, codon pairs with dst(pq) > 8 were disregarded. The IDW amino acid pair frequencies were calculated in the same fashion.

Two previously published encoding schemes were compared with our encoding scheme in this study. The first one is the CT encoding[20] where 20 amino acids were firstly classified into seven classes according to their dipoles and volumes. Then the total number of different tripeptides (triads) in a protein sequence was counted. The numbers of tripeptides composed of amino acids belonging to the same class, *e.g.* ADR and VEK, were added up separately. Finally, a total of $7 \times 7 \times 7 = 343$-dimensional features could be expected. These features were then normalized as:

$$f_{\mathrm{normalized}}(x) = \frac{f(x) - \min\{f(1), f(2), ..., f(343)\}}{\max\{f(1), f(2), ..., f(343)\}} \quad (3)$$

The second one is the AC encoding[18] that considers the auto covariance in terms of physicochemical properties between two residues spaced with a certain number of residues in a pair of proteins. See the original paper[18] for details. It is worth mentioning that the above two sequence encoding schemes concatenated feature vectors for a pair of proteins, instead of calculating the differences between them.

Yeast testing datasets

SVM predictors trained with codon pair frequency differences and other encodings were extensively tested by 10-fold cross-validation tests using three kinds of combined datasets of 4156 DIP positives and the equal number of non-interacting protein pairs. The first kind of datasets that contains randomly selected non-interacting protein pairs as negatives

are termed as "DIP + Random". The second kind ("DIP + RSS Negative") contains "RSS Negative" without known similar functions or subcellular localizations. The index termed as RSS value[15] was proposed by Wu *et al.* to measure the similarity between Gene Ontology annotations of two proteins. The Gene Ontology[38] annotations of yeast proteins were downloaded from http://www.geneontology.org/ (version April 2011). Since Gene Ontology contains three types of annotations, *i.e.* Biological Process, Molecular Function and Cellular Component, there are three RSS values for each pair of proteins. An RSS value ranges from 0 to 1, with a higher RSS value corresponding to a stronger association. Details for calculating the RSS values can be found in the article.[15] The difference between our and Wu *et al.*'s methods is that the latter excluded several Cellular Component annotations from the calculation, while we did not. The "RSS Negative" datasets were randomly selected protein pairs whose RSS (Biological Process) and RSS (Cellular Component) were less than 0.4.[17] With respect to the third kind of datasets ("DIP + Homogeneous"), the negatives were generated by randomly rewiring the DIP positives.[20] For a more comprehensive benchmarking, all cross-validation tests were repeated five times by randomly sampling different negative datasets. We also tested CCPPI in the aforementioned three types of datasets after removing redundant proteins by the CD-HIT tool[39] using a 40% protein sequence identity cutoff.

The large-scale independent benchmarking test was performed as described below. All predictors were trained on a joint dataset of DIP positives, MIPS complex positives (available in the supporting materials of ref. 40) and the equal number of randomly selected negative pairs. This training dataset is called "DIP + MIPS + Random". The testing dataset is composed of the BIOGRID interaction dataset[36] and 0.9 million randomly selected non-interacting protein pairs. The BIOGRID dataset was downloaded from the BIOGRID database (http://thebiogrid.org) and only physical interactions between yeast proteins were retained. In particular, we filtered out all protein pairs in the testing dataset that appeared in the training dataset. For the MIPS positives and BIOGRID positives, interactions from ribosomal protein complexes were discarded, which was suggested in ref. 19. The training and testing datasets for this large-scale benchmarking test are available at http://protein.cau.edu.cn/ccppi/download.html.

It is noteworthy that proteins with less than 30 amino acids were removed from all datasets, as requested by the AC encoding.[41] In addition, we also removed duplicated interactions and self-interactions in all of the datasets used.

SVM implementation and performance assessment

SVM training, testing and 10-fold cross-validation experiments were implemented using the LIBSVM package.[42] All SVM models were constructed with the radial basis function (RBF) kernel. Unless otherwise stated, the parameter c was preliminarily optimized to 10 and the other SVM parameters were set to their default values. All the three encoding schemes (*i.e.* CCPPI, CT encoding and AC encoding) perform better with $c = 10$ than the default c.

Four performance measures based on the default SVM cutoff value (*i.e.* zero) were introduced using the following definitions:

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \quad (4)$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (5)$$

$$\text{sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (6)$$

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP}+\text{FP}) \times (\text{TP}+\text{FN}) \times (\text{TN}+\text{FP}) \times (\text{TN}+\text{FN})}} \quad (7)$$

where TP, FP, TN, FN represent the number of true positives, false positives, true negatives and false negatives, respectively. MCC, the Matthew's Correlation Coefficient, is a comprehensive indicator of a predictor's performance.

In order to evaluate the performance on the large-scale benchmark dataset, the Receiver Operating Characteristic (ROC) curves were generated by plotting the true positive rate (*i.e.* sensitivity) as a function of the false positive rate (*i.e.* 1-specificity). The specificity is defined as:

$$\text{specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (8)$$

We manipulated the SVM cutoff value to change the specificity level. The overall performance is thus quantified by the Area Under Curve (AUC) value.

Results and discussion

Non-random codon pair frequency distribution among interacting protein pairs

We compared codon pair frequency differences between 4380 interacting protein pairs from the DIP database[37] and randomly selected protein pairs which are 19-fold larger than the former. In total, there are $61 \times 61 = 3721$ codon pairs under investigation. Compared with randomly selected protein pairs, 1551 out of 3721 codon pairs in the interacting protein pairs were observed to have significantly similar frequencies (Welch's t-test followed by Benjamini–Hochberg correction, $p < 0.05$; Fig. 1). At the same significance level, the frequencies of 619 codon pairs in interacting protein pairs tend to be dissimilar. Moreover, there is a considerable fraction (41.7%) of codon pairs that do not have any significant difference. In contrast, 57 out of 61 codons in the interacting protein pairs show similar frequencies ($p < 0.05$; Fig. 1), which is consistent with previous observations based on a different dataset.[19]

Although the potential amount of informative codon pairs seemed to be large, it could be expected that non-random usage of these codon pairs in PPIs was a result of non-random codon usage and non-random amino acid pair usage. To test this, we generated 1000 permutated sequence sets where only the synonymous codons in each coding sequence were shuffled. Therefore, the codon pair usage was altered but the codon usage and amino acid pair usage remained unchanged

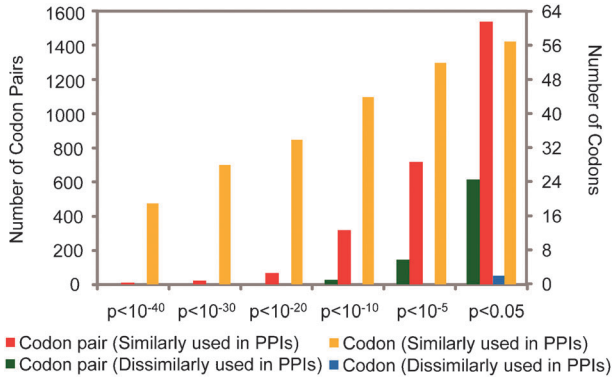

Fig. 1 Comparison of codon frequency differences and codon pair frequency differences between the PPIs and random ones. The columns indicate the total numbers of codons or codon pairs whose frequencies are shown to be either significantly more similar or dissimilar between interacting protein pairs in comparison with randomly selected non-interacting ones. In other words, a corrected *p*-value was calculated for each codon or codon pair to describe whether it was informative in discriminating interacting protein pairs and non-interacting protein pairs. The *x* axis lists the corrected *p*-value cutoff under which one codon/codon pair was treated as significantly similarly or dissimilarly used in PPIs, while the *y* axis shows the total number of codons or codon pairs that meets each individual cutoff.

(see Methods, ESI†). Among at least 950 out of 1000 permutated sets, there were 198 codon pairs still shown to be more similarly used in PPIs in comparison with protein pairs with permutated coding sequences and 85 codon pairs still shown to be dissimilarly used in PPIs (*i.e.* empirical $p < 0.05$). These informative codon pairs are more likely to be independent of codon usage and amino acid pair usage, indicating that the non-random usage of codon pairs is beyond the combination of non-random codon usage and non-random amino acid pair usage. Therefore, a predictor based on codon pair frequency differences may perform better in distinguishing interacting protein pairs from random protein pairs, which we will examine in the following sections.

CCPPI: codon pair usage as the encoding feature under the SVM framework

For a pair of proteins, a feature vector consisting of 3721 codon pair frequency differences between them was constructed. Considering that SVM is suitable for dealing with such high dimensional feature vectors, we used the codon pair frequency differences as the encoding scheme and developed an SVM-based PPI predictor called CCPPI.

Based on "DIP + Random" datasets, we compared the performance of CCPPI and the other sequence-based encoding schemes through 10-fold cross-validation tests under the same SVM framework, for a fair comparison of the performance of CCPPI and other different encoding schemes. These encoding schemes include amino acid frequency differences, amino acid pair frequency differences and codon frequency differences. The performances of these encoding schemes are summarized in Table 1. It can be seen that codon-derived encodings outperformed the corresponding amino acid-derived encodings. That is to say, the codon frequency difference encoding outperformed the amino acid frequency difference encoding and CCPPI achieved a better performance than the amino acid pair frequency difference encoding. Moreover, CCPPI also significantly outperformed the codon frequency difference-based encoding with approximately 8% accuracy increase under the SVM framework (Table 1). We noted that the predictor based on the codon frequency difference encoding had been initially established under the Naïve Bayes framework.[19] This Naïve Bayes predictor could only reach an accuracy of 61.0 ± 0.3% (MCC = 0.222 ± 0.005) in the "DIP + Random" datasets using 10-fold cross-validation tests. Similarly, codon pair frequency difference encoding under the Naïve Bayes framework performed worse (accuracy = 66.8 ± 0.3%, MCC = 0.336 ± 0.006) than that under the SVM framework (Table 1), indicating that the SVM framework would be a more favorable choice to construct codon/codon pair information-based predictors. In summary, these results suggest that codon pair usage is more informative for PPI prediction.

Spaced amino acid pair information has been previously shown to be useful for improving prediction accuracy.[18] Therefore, inverted distance weighting (IDW) was introduced to extend amino acid pair frequency or codon pair frequency in a spaced pair encoding fashion. Indeed, such extensions improved the prediction accuracy of amino acid pair frequency-based predictors by 3% (achieving a level nearly comparable to CCPPI), but did not lead to a better performance for codon pair-based predictors (Table 1). Spaced amino acid pairs can reflect patterns of local amino acid distribution throughout the protein sequence[18] and plausibly the residue context of the protein interaction interface. In contrast, spaced codon pairs represented in this form seem to lack straightforward biologically meaningful information for PPI prediction. As a simple extension of codon pairs that accounts for more distal neighboring codons, the major advantage of IDW is that it does not generate higher dimensional feature vectors. On the other hand, IDW cannot

Table 1 Performance of different sequence encoding schemes evaluated by 10-fold cross-validation tests

Encoding scheme	Accuracy (%)	Precision (%)	Sensitivity (%)	MCC
Amino acid frequency difference	60.6 ± 0.5	61.1 ± 0.5	58.1 ± 0.9	0.211 ± 0.011
Amino acid pair frequency difference	70.3 ± 0.4	68.7 ± 0.5	74.4 ± 0.7	0.407 ± 0.008
Codon frequency difference	66.4 ± 0.5	65.8 ± 0.4	68.3 ± 0.9	0.329 ± 0.010
Codon pair frequency difference (CCPPI)	74.8 ± 0.4	73.3 ± 0.5	78.1 ± 0.5	0.498 ± 0.008
IDW amino acid pair frequency difference	73.3 ± 0.3	74.5 ± 0.3	70.9 ± 0.4	0.466 ± 0.007
IDW codon pair frequency difference	74.3 ± 0.4	73.3 ± 0.2	76.6 ± 0.9	0.488 ± 0.008
CT encoding	68.6 ± 0.7	68.0 ± 0.7	70.3 ± 0.7	0.372 ± 0.013
AC encoding	63.3 ± 0.5	62.4 ± 0.4	66.7 ± 1.3	0.266 ± 0.010

All the 10-fold cross-validation tests were repeated five times by selecting different negative samples. The results are expressed as mean ± standard deviation. The predictors were trained with the preliminarily optimized parameters.

completely represent the information of spaced codon pairs, due to its arbitrary weighting scheme. This suggests that more sophisticated extensions of codon pair frequencies are desirable.

Comparison of the performance of CCPPI with other encoding schemes

We further compared CCPPI with another two popular encoding schemes that were frequently cited in the literature, namely the CT[20] and AC[18] encoding schemes. Both encodings were implemented using the SVM-based predictors (Table S1, ESI†). We first evaluated the prediction performance of these two encoding schemes (not the methods) by 10-fold cross-validation tests on the aforementioned "DIP + Random" datasets. As shown in Table 1, the accuracies for these two encodings are about 5–10% lower compared with CCPPI. The corresponding MCC values of these two encodings were 0.12 and 0.23 lower than CCPPI, respectively.

It should be noted that cross-validation tests on such balanced datasets often overestimate the performance of PPI predictors. This is due to the nature of the extreme unbalance between interacting protein pairs and non-interacting protein pairs.[43] Therefore, we further examined the performance of CCPPI and the other two encoding schemes on a large-scale benchmark dataset. As an addition to the training dataset for all three encodings, the MIPS protein complex dataset was introduced for a more sufficient training (resulting in the "DIP + MIPS + Random" dataset). CCPPI predictor trained on this dataset is accessible at http://protein.cau.edu.cn/ccppi/. The benchmark testing dataset combines 48 993 positive protein pairs from the BIOGRID database[36] and 0.9 million randomly selected negative protein pairs, reaching a high positive-to-negative ratio of 1 : 18.

The performance of CCPPI and the other two encoding schemes on the benchmark dataset was characterized by ROC curves, as shown in Fig. 2. CCPPI achieved an AUC value of 0.695, outperforming CT and AC encoding schemes (0.616 and 0.577). More importantly, at the 90% specificity level, CCPPI correctly predicted 14 634 interactions from the BIOGRID dataset, while CT and AC encodings predicted 10 652 and 7294 interactions, respectively. The prediction consistency among the different encoding schemes was illustrated using the Venn diagrams (Fig. 3). The predicted true positives by the three predictors at the 90% specificity level show moderate overlap only, which indicates the potentiality of being integrated as a more powerful predictor, irrespective of computational burdens. In other words, a meta or consensus approach[44] can potentially be developed to make a better prediction by integrating the prediction results of all three encoding schemes. Besides, we also retrained other encodings presented in Table 1 in the same way to test them on this large-scale benchmark dataset. Codon pair-based encodings also performed the best, and CCPPI performed especially well at the specificity levels higher than 90% (Fig. S1, ESI†).

We noted that above performance of these sequence encodings was estimated using preliminarily optimized SVM parameters. To confirm our finding, we also performed parameter optimization through 10-fold cross-validation tests on a "DIP + Random" dataset (see Table S2, ESI†, for the listed optimized parameters).

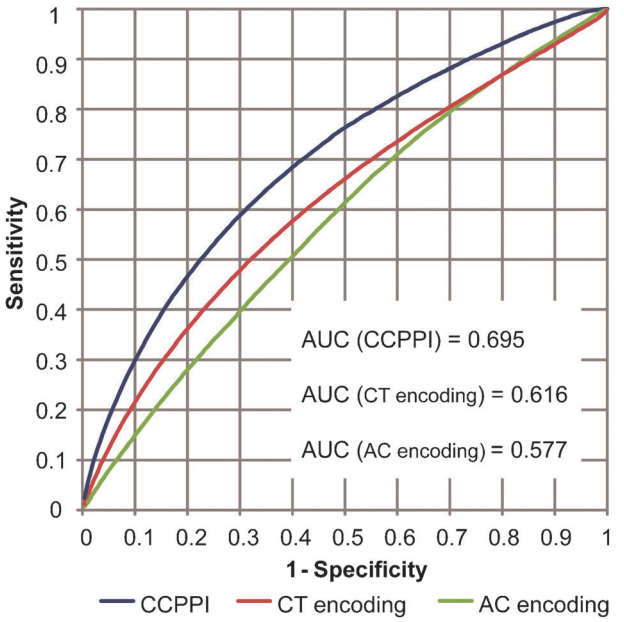

Fig. 2 The ROC curves illustrating the overall performance of the CCPPI and the other two encoding schemes by using the large-scale testing dataset composed of the BIOGRID positives and 0.9 million randomly selected negatives. The predictors were trained with the preliminarily optimized parameters.

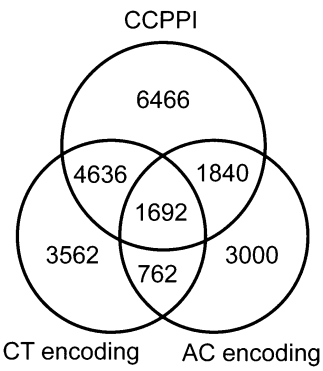

Fig. 3 Venn diagram showing the overlap of the predicted true positives by CCPPI and the other two encoding schemes at the 90% specificity level in the large-scale testing. The predictors were trained with the preliminarily optimized parameters.

All encodings were re-trained using the optimized parameters and the "DIP + MIPS + Random" dataset. As indicated by the repeated large-scale testing, CT encoding and AC encoding showed significant improvement after parameter optimization, but CCPPI performed slightly worse than those based on the preliminarily optimized parameters, possibly due to the over-fitting induced by optimization (Fig. S2, ESI†). Nonetheless, CCPPI still ranked the best in terms of the AUC value, with a comparable performance to the CT encoding. In addition, at least 25% of the true positives yielded from CCPPI at the 90% specificity level were predicted by neither of the other two encodings at the same specificity level (Fig. S3, ESI†). Indeed, a simple meta-predictor constructed by weighted summing of the decision values from the three encodings-based predictors could outperform any individual predictor (Fig. S2, ESI†).

Why CCPPI is more informative than the other encoding schemes in predicting PPIs?

To better understand why CCPPI performed better than the other two commonly used sequence encodings for PPI prediction, we elaborated on the exclusively predicted true positives (*i.e.* predicted true positives that could not be predicted by any other individual encoding at the 90% specificity level) by the three encodings. In particular, we interrogated four important factors that are presumably associated with protein–protein interactions, including (1) transcriptional co-expression, (2) proteomic co-expression, (3) functional similarity and (4) subcellular localization similarity (see Methods, ESI† for details).

As shown in Table 2, exclusively predicted true positives of CCPPI seemed to be enriched for transcriptional co-expressed proteins (Fisher's exact test, $p < 0.05$). We checked if such co-expression protein pairs were indeed overrepresented in predicted true positives of CCPPI in comparison with the whole BIOGRID dataset. It turned out that transcriptional co-expressed protein pairs were underrepresented for the other two encodings, rather than the overrepresentation for CCPPI (Table 2).

Unexpectedly, the difference in proteomic co-expression seemed ambiguous (*i.e.* CCPPI *vs.* AC encoding Fisher's exact test, $p < 0.05$, but CCPPI *vs.* CT encoding, $p > 0.05$; Table 2). In addition to the data quality, there are at least two other possible explanations for this. Firstly, the control mechanisms of protein expression are so complicated that the contribution of codon pair usage to proteomic co-expression is not pronounced. Secondly, both amino acid usage and codon pair usage contribute to proteomic co-expression and it is thus reasonable that no significant difference in their association with co-expression could be observed. We found that the predicted true positive protein pairs by both CCPPI and CT encodings at the 90% specificity level tend to be co-expressed at the proteomic level, in contrast to the whole BIOGRID dataset (Welch's *t*-test, $p < 0.05$; Table 2). This finding is also consistent with a recent observation that both codon usage and amino acid usage play important roles as determinants affecting protein abundance and translation.[45] Therefore, the second explanation seems to be more convincing. It has been proposed that similar usage of codon could help "synchronizing the translation of functionally associated protein pairs" (including but not limited to PPIs) across various eukaryotes.[35] We argue that codon pair usage may play a similar role in mediating correlated protein expression among PPIs.

We also tested the RSS values that reflect the functional or subcellular localization similarity between interacting protein pairs. Interestingly, exclusively predicted true positives by CCPPI showed significantly higher similarity than those of the other two encodings in terms of protein function (Welch's *t*-test, $p < 1 \times 10^{-3}$; Table 2) but not subcellular localization ($p > 0.05$, Table 2). Conversely, CCPPI performed especially well on the datasets where such a factor is more discernable, as discussed below.

Performance comparison based on different datasets

It has been previously suggested that the objective evaluation of PPI predictors should be ideally performed on multiple datasets with negatives generated in different ways,[17,18] because gold standard datasets of *bona fide* non-interactions are still under development.[46] From this perspective, RSS Negative, a type of negative datasets other than randomly selected ones, collected protein pairs without any known significant similarity of function or subcellular localization. On the "DIP + RSS Negative" datasets, CCPPI achieved accuracy as high as 90.2%, still outperforming the other two encoding schemes (Table S3, ESI†). This indicates that CCPPI captured important information contained in the subcellular localization, and especially functional similarity of interacting protein pairs.

Despite the above encouraging results, it is likely that there are cases where CCPPI and other predictors will perform poorly.[18] To explore this possibility, we tested these encodings on the "DIP + Homogeneous" datasets (see Materials and Methods for details). These datasets are more challenging in a sense that the protein repertories of positive protein pairs and negative pairs become somewhat homogenous, making it more difficult to differentiate between each other. As shown in Table S4 (ESI†), all of the encoding schemes including CCPPI were subjected to performance decline when evaluated by 10-fold cross-validation tests on these datasets. But CCPPI still achieved the highest accuracy and MCC value, indicating that it is more robust in dealing with this situation compared with the other two encoding schemes.

Table 2 Comparison of four different factors that presumably contribute to the true positive prediction of each encoding scheme

Encoding scheme	Transcriptional co-expression	Proteomic co-expression	Functional similarity	Subcellular localization similarity
(a) Factors of exclusively predicted true positives at the 90% specificity				
CCPPI	0.034	0.145	0.689	0.902
CT encoding	*0.022*	0.139	*0.640*	0.889
AC encoding	*0.021*	*0.086*	*0.666*	0.900
(b) Factors of all predicted true positives at the 90% specificity				
CCPPI	0.028	**0.152**	**0.681**	0.904
CT encoding	0.023	**0.138**	0.663	0.900
AC encoding	0.020	0.097	**0.678**	**0.906**
(c) Factors of the whole BIOGRID dataset				
BIOGRID	0.026	0.112	0.671	0.902

Transcriptional co-expression and proteomic co-expression of protein pairs were measured by the fraction of co-expressed proteins. The functional similarity and subcellular localization similarity were measured by RSS values which range from 0 to 1. (a) A factor of CT or AC encoding is shown in italic if it is significantly lower ($p < 0.05$) than that of CCPPI. (b) A factor is highlighted in bold if it is significantly larger ($p < 0.05$) than the average level of the BIOGRID dataset, which is presented in the part c of this table.

Although CCPPI was ranked the best when tested on any type of the negative datasets, the resulting accuracies diverged considerably. RSS Negative is a good type of negative datasets as it has the least overlap with *bona fide* interactions. However, it is also biased towards specific negative protein pairs whose function and subcellular localization are dissimilar. Therefore, benchmarking on this type of datasets may result in an overestimated performance, especially for a predictor that is sensitive to functional similarity like CCPPI or AC encoding. By excluding the proteins out of the positive datasets, the homogenous negative datasets can serve as rigorous benchmarks. As a consequence, however, only a limited fraction of the proteome (*i.e.* proteins that present in PPI datasets) can be examined. In this study, as a compromise, the randomly selected negatives, which are neither of the highest quality nor strict enough by themselves, were ultimately chosen for benchmarking purposes, due to their neutral bias and relative high coverage.

We noted that in addition to the selection of negative datasets, the presence of similar proteins may also result in a performance overestimation. We have repeatedly performed cross-validation tests using the aforementioned three types of datasets after removing redundant sequences (at 40% sequence identity cutoff). CCPPI showed 0.6–1.6% decline of accuracy among different types of datasets after such a filtering procedure (Table 3), indicating that sequence similarity is not a major factor that contributes to CCPPI's performance. Similarly, filtering redundant sequences did not alter the conclusion of the statistical analyses about codon/codon pair usage (Fig. S4, ESI†). However, we also found that the true positive prediction of CCPPI at the 90% specificity in the large-scale testing was enriched for interactions between paralogous protein pairs (sequence identity $>40\%$), in comparison with the whole BIOGRID dataset (0.64% *vs.* 0.33%, Fisher's exact test, $p < 0.05$). To alleviate this bias, we have made available an optional post-filter to exclude paralogous protein pairs (sequence identity $>40\%$) in our CCPPI prediction server.

Reasonable strategies to construct more powerful PPI predictors

From a more realistic point of view, due to the particularly low ratios between interacting and non-interacting protein pairs, CCPPI is expected to yield many more false positives than true positives even at the 90% specificity level. The presence of high false positive rate is reflected not only by the poor performance on testing datasets aimed at distinguishing interactions from non-interactions within the same group of proteins (Table S4, ESI†), but also by the poorer performance when tested on a false positive-prone dataset.[47] This has necessitated the integration of CCPPI with other PPI predictors.

Prediction methods like interolog, domain–domain interaction and phylogenetic profile could only predict protein interactions with known homologs or domains. In this context, integrating CCPPI with these methods is likely to achieve a better balance between the prediction coverage and false positive rate. This is exemplified by the PPIs in the fruit fly interactome. In this particular dataset,[48] there are 26 545 known interactions, out of which the interolog method predicted 985 interactions by transferring PPIs from yeast, worm (*Caenorhabditis elegans*), mouse and human orthologs. CCPPI trained on the "DIP + MIPS + Random" dataset was performed on the known fly interactions with a cutoff value of 0.39, which was estimated to have a 90% specificity according to the benchmarking on the large-scale yeast dataset. As a result, 4146 interactions were predicted by CCPPI. We estimated the specificity of each predictor using 150 000 randomly selected non-interacting fruit fly protein pairs. The interolog method showed fairly high specificity (99.9%), which is much higher than CCPPI (92.2%). Nonetheless, only 219 PPIs were predicted by both methods, indicating limited overlap of the two methods.

766 PPIs were predicted by the interolog method but not by CCPPI, presumably due to lack of correlation in codon pair usage in comparison with the 219 PPIs that could be predicted by both methods (Pearson's correlation coefficient, 0.13 *vs.* 0.21, $p < 1 \times 10^{-15}$). In contrast, there were 3927 PPIs that could only be predicted by CCPPI, which could be grouped into two categories. In the first category, the interolog method failed to predict 1177 interactions due to the limited availability of interaction data in other species.[36,37,49] That means, although orthologous protein pairs from other organisms could be identified, there was no experimental evidence showing that these protein pairs truly interact, and thus no interaction information could be transferred. In the second category, 1574 interactions did not have any convincible orthologs in other organisms. Therefore, they could not be predicted by the interolog method even if more PPI data of other species became available. In addition, we also compared CCPPI with another two homology-dependent methods, namely the domain–domain interaction[8,10] and phylogenetic profile methods[12] (see Methods for details, ESI†). As it can be seen in Table S5 (ESI†) and Fig. 4, CCPPI showed a limited overlap with other methods, though it had a higher false positive rate.

A typical example is related to the PPI prediction of FBgn0032789. According to the FlyBase annotations,[50] FBgn0032789 is an essential protein, which has no identifiable homolog from species other than fly species in the *Drosophila* genus. Therefore, all of the 31 PPIs that were involved in the current fruit fly interactome could not be discovered by the homology-dependent methods. However, CCPPI could correctly predict 15 PPIs

Table 3 Performance of CCPPI after filtering redundant sequences in the datasets

Datasets	Accuracy (%)	Precision (%)	Sensitivity (%)	MCC
DIP + Random	73.2 ± 0.3	71.3 ± 0.8	77.6 ± 1.0	0.466 ± 0.006
DIP + RSS Negative	89.6 ± 0.4	88.1 ± 0.6	91.7 ± 0.1	0.793 ± 0.008
DIP + Homogeneous	62.5 ± 0.3	64.4 ± 0.7	55.6 ± 1.3	0.253 ± 0.007

We removed redundant sequences using a 40% identity cutoff, which resulted in datasets containing 3460 interacting protein pairs and 3460 randomly selected non-interacting protein pairs. The performance was evaluated through the 10-fold cross-validation tests, which were repeated five times by selecting different negative samples. The results are expressed as mean ± standard deviation.

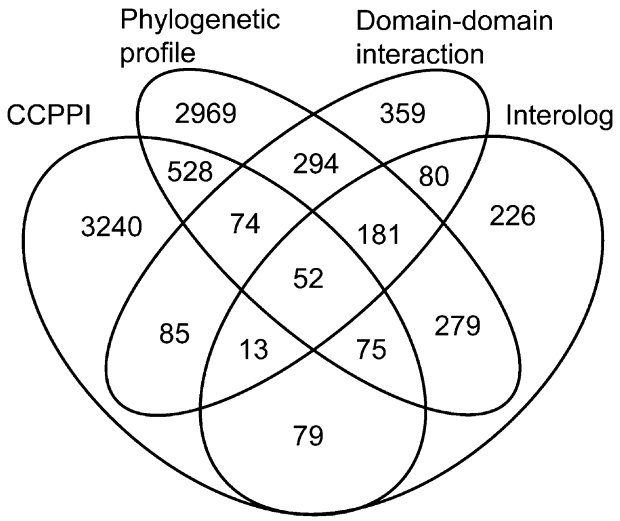

Fig. 4 Venn diagram showing the overlap of the predicted true positives from the fruit fly interactome, using CCPPI and homology-dependent methods. For CCPPI, a cutoff value corresponding to the 90% specificity level in the yeast large-scale testing was used. See Supplemental Methods and Table S5 (ESI†) for details of the other methods and the comprehensive performance evaluation, respectively.

it participates in, where mutations of four important partner proteins will induce severe growth or reproductive defects (FBgn0033988, FBgn0039385, FBgn0034523 and FBgn0030583). Interestingly, the latter three were proteins with unknown function. That is to say, CCPPI has identified essential PPIs where uncharacterized proteins were involved with relatively high decision values. These results illustrate the potential value of applying simple sequence encoding-based methods to identify novel interactions involving uncharacterized proteins or non-conserved proteins. We also observed a limited overlap of the predicted true positives between homology-dependent methods and the CT or AC encoding which was trained on the "DIP + MIPS + Random" dataset (Fig. S5 and S6, ESI†). This indicates the ability and potential of simple sequence encoding-based predictors to explore a unique niche in the interactome.

Conclusions

Our studies indicate that codon pair usage encodes additional important information that can be used to predict functionally or physically related protein partners. The developed codon pair based method CCPPI is capable of predicting protein–protein interactions, with a favorable or at least competitive performance in comparison with several well-known sequence-based encoding schemes. We would also like to point out that, like many of the existing PPI predictors, CCPPI suffers from a high false positive rate. CCPPI could partially extract information regarding the proteomic co-expression and functional similarity of interacting protein pairs, which is distinct from homology and difficult to be obtained by high-throughput experiments. We therefore propose that integration of CCPPI with other effective and complementary methods that are developed based on protein homology, such as the interolog approach, may be further helpful for enhancing the performance, coverage and reliability of PPI predictions.

Acknowledgements

We are grateful to the anonymous reviewers for their constructive and insightful comments. We thank Zhi-Gang Li, Ren-Xiang Yan, Zhen Chen, Xiao-Bao Dong and Xiao-Feng Wang at China Agricultural University for helpful discussions. We also appreciate Prof. Menglong Li and Dr Yanzhi Guo at Sichuan University for sharing their programs. JS is an NHMRC Peter Doherty Fellow and the Recipient of the Hundred Talents Program of CAS. This work was supported by the National Natural Science Foundation of China (31070259), the National Basic Research Program of China (2009CB918802), the Hundred Talents Program of the Chinese Academy of Sciences (CAS), the Knowledge Innovative Program of CAS (KSCX2-EW-G-8), Tianjin Municipal Science & Technology Commission (10ZCKFSY05600) and the National Health and Medical Research Council of Australia (NHMRC).

Notes and references

1 E. Pieroni, S. de la Fuente van Bentem, G. Mancosu, E. Capobianco, H. Hirt and A. de la Fuente, *Proteomics*, 2008, **8**, 799–816.
2 H. Yu, P. Braun, M. A. Yildirim, I. Lemmens, K. Venkatesan, J. Sahalie, T. Hirozane-Kishikawa, F. Gebreab, N. Li, N. Simonis, T. Hao, J. F. Rual, A. Dricot, A. Vazquez, R. R. Murray, C. Simon, L. Tardivo, S. Tam, N. Svrzikapa, C. Fan, A. S. de Smet, A. Motyl, M. E. Hudson, J. Park, X. Xin, M. E. Cusick, T. Moore, C. Boone, M. Snyder, F. P. Roth, A. L. Barabasi, J. Tavernier, D. E. Hill and M. Vidal, *Science*, 2008, **322**, 104–110.
3 A. C. Gavin, M. Bosche, R. Krause, P. Grandi, M. Marzioch, A. Bauer, J. Schultz, J. M. Rick, A. M. Michon, C. M. Cruciat, M. Remor, C. Hofert, M. Schelder, M. Brajenovic, H. Ruffner, A. Merino, K. Klein, M. Hudak, D. Dickson, T. Rudi, V. Gnau, A. Bauch, S. Bastuck, B. Huhse, C. Leutwein, M. A. Heurtier, R. R. Copley, A. Edelmann, E. Querfurth, V. Rybin, G. Drewes, M. Raida, T. Bouwmeester, P. Bork, B. Seraphin, B. Kuster, G. Neubauer and G. Superti-Furga, *Nature*, 2002, **415**, 141–147.
4 B. A. Shoemaker and A. R. Panchenko, *PLoS Comput. Biol.*, 2007, **3**, e43.
5 F. He, Y. Zhang, H. Chen, Z. Zhang and Y. L. Peng, *BMC Genomics*, 2008, **9**, 519.
6 L. R. Matthews, P. Vaglio, J. Reboul, H. Ge, B. P. Davis, J. Garrels, S. Vincent and M. Vidal, *Genome Res.*, 2001, **11**, 2120–2126.
7 W. K. Kim, J. Park and J. K. Suh, *Genome Inform.*, 2002, **13**, 42–50.
8 T. Y. Wang, F. He, Q. W. Hu and Z. Zhang, *Mol. Biosyst.*, 2011, **7**, 2278–2285.
9 C. Guda, B. R. King, L. R. Pal and P. Guda, *PLoS One*, 2009, **4**, e5096.
10 R. D. Finn, M. Marshall and A. Bateman, *Bioinformatics*, 2005, **21**, 410–412.
11 D. R. Rhodes, S. A. Tomlins, S. Varambally, V. Mahavisno, T. Barrette, S. Kalyana-Sundaram, D. Ghosh, A. Pandey and A. M. Chinnaiyan, *Nat. Biotechnol.*, 2005, **23**, 951–959.
12 J. Sun, J. Xu, Z. Liu, Q. Liu, A. Zhao, T. Shi and Y. Li, *Bioinformatics*, 2005, **21**, 3409–3415.
13 H. Ge, Z. Liu, G. M. Church and M. Vidal, *Nat. Genet.*, 2001, **29**, 482–486.
14 A. K. Ramani, Z. Li, G. T. Hart, M. W. Carlson, D. R. Boutz and E. M. Marcotte, *Mol. Syst. Biol.*, 2008, **4**, 180.
15 X. Wu, L. Zhu, J. Guo, D. Y. Zhang and K. Lin, *Nucleic Acids Res.*, 2006, **34**, 2137–2150.
16 L. J. Jensen, M. Kuhn, M. Stark, S. Chaffron, C. Creevey, J. Muller, T. Doerks, P. Julien, A. Roth, M. Simonovic, P. Bork and C. von Mering, *Nucleic Acids Res.*, 2009, **37**, D412–D416.
17 M. G. Shi, J. F. Xia, X. L. Li and D. S. Huang, *Amino Acids*, 2010, **38**, 891–899.

18 Y. Guo, L. Yu, Z. Wen and M. Li, *Nucleic Acids Res*., 2008, **36**, 3025–3030.
19 H. S. Najafabadi and R. Salavati, *Genome Biol*., 2008, **9**, R87.
20 J. Shen, J. Zhang, X. Luo, W. Zhu, K. Yu, K. Chen, Y. Li and H. Jiang, *Proc. Natl. Acad. Sci. U. S. A*., 2007, **104**, 4337–4341.
21 S. Martin, D. Roe and J. L. Faulon, *Bioinformatics*, 2005, **21**, 218–226.
22 F. Crick, *Nature*, 1970, **227**, 561–563.
23 R. Grantham, C. Gautier, M. Gouy, M. Jacobzone and R. Mercier, *Nucleic Acids Res*., 1981, **9**, r43–r74.
24 M. Bulmer, *Nature*, 1987, **325**, 728–730.
25 C. Gustafsson, S. Govindarajan and J. Minshull, *Trends Biotechnol*., 2004, **22**, 346–353.
26 P. Cortazzo, C. Cervenansky, M. Marin, C. Reiss, R. Ehrlich and A. Deana, *Biochem. Biophys. Res. Commun*., 2002, **293**, 537–541.
27 F. Kepes, *J. Mol. Biol*., 1996, **262**, 77–86.
28 R. Saunders and C. M. Deane, *Nucleic Acids Res*., 2010, **38**, 6719–6728.
29 G. A. Gutman and G. W. Hatfield, *Proc. Natl. Acad. Sci. U. S. A*., 1989, **86**, 3699–3703.
30 B. Irwin, J. D. Heck and G. W. Hatfield, *J. Biol. Chem*., 1995, **270**, 22801–22806.
31 J. F. Curran, E. S. Poole, W. P. Tate and B. L. Gross, *Nucleic Acids Res*., 1995, **23**, 4104–4108.
32 J. R. Coleman, D. Papamichail, S. Skiena, B. Futcher, E. Wimmer and S. Mueller, *Science*, 2008, **320**, 1784–1787.
33 G. Lithwick and H. Margalit, *Nucleic Acids Res*., 2005, **33**, 1051–1057.
34 H. B. Fraser, A. E. Hirsh, D. P. Wall and M. B. Eisen, *Proc. Natl. Acad. Sci. U. S. A*., 2004, **101**, 9033–9038.
35 H. S. Najafabadi, H. Goodarzi and R. Salavati, *Nucleic Acids Res*., 2009, **37**, 7014–7023.
36 C. Stark, B. J. Breitkreutz, A. Chatr-Aryamontri, L. Boucher, R. Oughtred, M. S. Livstone, J. Nixon, K. Van Auken, X. Wang, X. Shi, T. Reguly, J. M. Rust, A. Winter, K. Dolinski and M. Tyers, *Nucleic Acids Res*., 2011, **39**, D698–D704.
37 I. Xenarios, L. Salwinski, X. J. Duan, P. Higney, S. M. Kim and D. Eisenberg, *Nucleic Acids Res*., 2002, **30**, 303–305.
38 M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin and G. Sherlock, *Nat. Genet*., 2000, **25**, 25–29.
39 W. Li and A. Godzik, *Bioinformatics*, 2006, **22**, 1658–1659.
40 R. Jansen, H. Yu, D. Greenbaum, Y. Kluger, N. J. Krogan, S. Chung, A. Emili, M. Snyder, J. F. Greenblatt and M. Gerstein, *Science*, 2003, **302**, 449–453.
41 Y. Guo, M. Li, X. Pu, G. Li, X. Guang, W. Xiong and J. Li, *BMC Res. Notes*, 2010, **3**, 145.
42 C. C. Chang and C. J. Lin, *ACM Trans. Intell. Syst. Technol*., 2011, **2**, 21–27.
43 Y. Park, *BMC Bioinf*., 2009, **10**, 419.
44 J. F. Xia, X. M. Zhao and D. S. Huang, *Amino Acids*, 2010, **39**, 1595–1599.
45 T. Tuller, M. Kupiec and E. Ruppin, *PLoS Comput. Biol*., 2007, **3**, e248.
46 P. Smialowski, P. Pagel, P. Wong, B. Brauner, I. Dunger, G. Fobo, G. Frishman, C. Montrone, T. Rattei, D. Frishman and A. Ruepp, *Nucleic Acids Res*., 2010, **38**, D540–D544.
47 J. Yu, M. Guo, C. J. Needham, Y. Huang, L. Cai and D. R. Westhead, *Bioinformatics*, 2010, **26**, 2610–2614.
48 A. M. Wiles, M. Doderer, J. Ruan, T. T. Gu, D. Ravi, B. Blackman and A. J. Bishop, *BMC Syst. Biol*., 2010, **4**, 36.
49 F. He, Y. Zhou and Z. Zhang, *Plant Physiol*., 2010, **153**, 1492–1505.
50 S. Tweedie, M. Ashburner, K. Falls, P. Leyland, P. McQuilton, S. Marygold, G. Millburn, D. Osumi-Sutherland, A. Schroeder, R. Seal and H. Zhang, *Nucleic Acids Res*., 2009, **37**, D555–D559.

