Methods xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Methods

journal homepage: www.elsevier.com/locate/ymeth

Proteome compression via protein domain compositions

Morihiro Hayashida *, Peiying Ruan, Tatsuya Akutsu

Bioinformatics Center, Institute for Chemical Research, Kyoto University, Gokasho, Uji, Kyoto 611-0011, Japan

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Grammar-based compression
Protein domain composition
Integer linear programming

ABSTRACT

In this paper, we study domain compositions of proteins via compression of whole proteins in an organism for the sake of obtaining the entropy that the individual contains. We suppose that a protein is a multiset of domains. Since gene duplication and fusion have occurred through evolutionary processes, the same domains and the same compositions of domains appear in multiple proteins, which enables us to compress a proteome by using references to proteins for duplicated and fused proteins. Such a network with references to at most two proteins is modeled as a directed hypergraph. We propose a heuristic approach by combining the Edmonds algorithm and an integer linear programming, and apply our procedure to 14 proteomes of *Dictyostelium discoideum*, *Escherichia coli*, *Saccharomyces cerevisiae*, *Schizosaccharomyces pombe*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Arabidopsis thaliana*, *Oryza sativa*, *Danio rerio*, *Xenopus laevis*, *Gallus gallus*, *Mus musculus*, *Pan troglodytes*, and *Homo sapiens*. The compressed size using both of duplication and fusion was smaller than that using only duplication, which suggests the importance of fusion events in evolution of a proteome.

1. Introduction

A living individual is considered to be an open non-equilibrium system from the viewpoint of statistical mechanics. In an isolated system, the entropy increases according to the second law of thermodynamics. On the other hand, in an open system, a dissipative structure is constructed, and it reaches a reproducible steady state [1]. The DNA base sequences in an individual are one kind of information to be maintained under non-equilibrium environments, whereas the sequences have been mutated and substituted through evolutionary processes. If random mutation and substitution of bases were always allowed from one generation to another, then the resulted sequences would be completely random. It can be considered that the case corresponds to the isolated system in statistical mechanics, and the entropy of the sequence is maximized.

There are several studies to compress DNA and protein sequences, which might be useful to study the entropy of these sequences. It is known that DNA sequences include abundant repetition and palindromes. Grumbach and Tahi [2] developed the first compression method specified to DNAs, called biocompress-2, which is a lossless algorithm using Lempel and Ziv's approaches [3,4], and tries to detect repeats and palindromes in DNA sequences. Rivals et al. [5] developed the Cfact algorithm, which uses suffix trees, and tries to detect the longest exact matching repeat. Chen et al. [6] developed the DNACompress algorithm, which tries to detect approximate repeats using some efficient method. Willems et al. [7] developed the context-tree weighting (CTW) method, which was defined as a suffix tree with edges weighted by some occurrence probability. Matsumoto et al. [8] proposed combination methods with CTW for DNA and protein sequences, respectively. Cao et al. [9] proposed an expert model (XM) based on statistical properties and repetition within sequences, and their method outperformed all other DNA and protein sequence compressors. Zhu et al. [10] proposed an approximate repeat vector (ARV) model forming a reference codebook for compression of DNA sequences, and developed an adaptive particle swarm optimization-based memetic algorithm (POMA) to maximize the cover rate and minimize some distance of the code vectors on the sequences. Kuruppu et al. [11] proposed COMRAD (COMpression using RedundAncy of Dna) by adapting an existing compression algorithm, RAY [12], to DNA sequences using some knowledge about alphabet size and sequence evolution. Their method outperformed RLCSA [13] and RLZ [14] for several organisms, and was effective in long-range repetition detection. Compression of sequences in multiple organisms of the same species has been also studied as genomic repositories are rapidly growing. For the purpose, most methods compress such sequences using difference from reference sequences [15,16]. Unlike these compression methods, we deal with protein domain compositions instead of amino acid sequences. Many proteins contain domains, which are known as functional and structural units in proteins [17]. In addition, the same domain can be included in multiple

* Corresponding author.
E-mail address: morihiro@kuicr.kyoto-u.ac.jp (M. Hayashida).

http://dx.doi.org/10.1016/j.ymeth.2014.01.012
1046-2023/

distinct kinds of proteins. Furthermore, we make use of compression for understanding of evolution by measuring entropies of proteomes, not for saving memory and disk space. We regard a protein as a multiset of domains, and compress sets of proteins. As far as we know, this is the first study to compress a proteome using such domain compositions.

Several studies have been done for evolution of protein domains [18,19]. Gene duplication can occur when an mRNA is retrotranscribed to cDNA and randomly inserted into the genome [20]. As a result, the protein can be generated also from the duplicated gene in a different chromosome, and has evolved independently from the original one. It is known that unequal crossing-over induces another gene duplication [20]. If positions of hybridization in crossing-over are not the same between two strands, genes in the strand can be also duplicated. Fusion and fission of genes are evolutionary events that two or more genes in an organism are connected and compose a gene in a descendent organism, and, in contrast, that a gene is split into multiple genes, respectively. Kummerfeld and Teichmann applied their maximum parsimony method to several completely sequenced genomes, and reported that the number of fusion events is about fourfold larger than that of fission events [21]. The number of total domains and the number of domain families in a protein follow power-law and exponential distributions in many organisms, respectively. Nacher et al. [22] proposed evolutionary models including gene duplication, fusion, and internal duplication events to explain both distributions. It should be noted that the internal duplication event is also known as tandem repeats within a gene, and is not different from the usual external gene duplication event [23]. Thus, compressing a proteome is considered to be possible because genes and domains have been duplicated through evolutionary processes. We make use of gene duplication and fusion events, generate a directed hypergraph with weighted hyperedges from a proteome, and try to find the minimum spanning hypertree, where each vertex corresponds to a protein, and an edge weight represents the compressed cost for a protein using some proteins. However, Brejová et al. [24] showed that the problem of finding the minimum directed spanning hypertree in the hypergraph is NP-hard even if each hyperedge has at most three vertices. In addition, they proposed an integer linear programming (ILP) formulation and applied it to the problem of maximizing some likelihood function for detecting signals in DNA, which are short subsequences located near functional sites. Although ILP outputs the exact optimum, if we consider the gene fusion events that correspond to hyperedges with size three, the execution time is too long to obtain the solution because the number of pairs of proteins in an organism is large. Hence, we propose a greedy method to reduce the number of hyperedges for compressing a proteome. Our method first finds the optimum solution for a graph with usual edges. To this end, it minimizes a cost function, which means the compressed size, and is based on the cost function proposed by Adler and Mitzenmacher [25] for compressing web graphs. After that, hyperedges with size three are added to the solution in some heuristic way. We apply our proposed method to 14 organisms of *Dictyostelium discoideum*, *Escherichia coli*, *Saccharomyces cerevisiae*, *Schizosaccharomyces pombe*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Arabidopsis thaliana*, *Oryza sativa*, *Danio rerio*, *Xenopus laevis*, *Gallus gallus*, *Mus musculus*, *Pan troglodytes*, and *Homo sapiens*. The results suggest that the same domain would be frequently utilized in higher organisms.

2. Method

In this section, we formulate our problem for compressing a proteome, briefly review the integer linear programming (ILP)-based method for minimum spanning directed hypertree problems [24], and describe our proposed heuristic method.

2.1. Problem formulation

Let $\mathcal{P}$ and $\mathcal{D}$ be the set of proteins and domains in a given proteome, respectively. Each protein $P_i(\in \mathcal{P})$ consists of several domains in $\mathcal{D}$, and is supposed to represent a multiset. For instance, if P_i consists of two D_1s and one D_2, then $P_i = \{D_1, D_1, D_2\}$. We define the cost representing a protein using only domains by

$$cost(P_i) = \lceil \log |\mathcal{D}| \rceil \cdot |P_i|, \tag{1}$$

where $|S|$ denotes the number of elements in the (multi) set S, and $\lceil x \rceil$ is the minimum integer no smaller than x. Then, a proteome without compression is represented with size $\sum_{P_i \in \mathcal{P}} cost(P_i)$.

Adler and Mitzenmacher [25] considered the cost generating a web page using another page as follows. A web page consists of links to other web pages. A new page is often created by copying some links from an existing page to itself. This is similar to gene duplication that a new gene is created by copying an existing gene. They used a 0–1 vector each element of which represents absence or presence of a web page linked from the existing page. The cost includes the size of the vector, the length representing the existing page, and new links not included in the existing one. For our purpose, the cost generating a protein P_i from another protein P_j by deleting and/or adding domains appropriately, $P_j \to P_i$, is defined by

$$cost(P_i, P_j) = \lceil \log |\mathcal{P}| \rceil + |P_j| + \lceil \log |\mathcal{D}| \rceil \cdot |P_i - P_j|, \tag{2}$$

where $|P_i - P_j|$ denotes the number of domains of P_i that are not included in P_j, and $|P_j|$ means the size of the 0–1 vector representing whether or not each domain in P_j is included in P_i. If the gene coding P_i is duplicated from that coding P_j, $cost(P_i, P_j)$ can be smaller than $cost(P_i)$ and costs for duplication and fusion from other genes. Fig. 1 illustrates the cost generating $P_4 = \{D_1, D_1, D_2, D_4, D_5\}$ from $P_1 = \{D_1, D_1, D_2, D_3\}$ in a proteome $\mathcal{P} = \{P_1, P_2, P_3, P_4\}$ with $\mathcal{D} = \{D_1, D_2, D_3, D_4, D_5\}$. The rectangle denotes a 0–1 vector representing absence (0) or presence (1) of domains of P_1 in P_4, two D_1s and D_2 of P_1 remain in P_4, and D_3 disappears in P_4. Furthermore, D_4 and D_5 are added. Thus, $cost(P_4, P_1) = \lceil \log 4 \rceil + 4 + \lceil \log 5 \rceil \cdot 2 = 12$.

In addition to gene duplication events, we consider gene fusion events that two different genes in an organism are fused into one gene in a descendent organism [21]. Then, we consider to generate a protein P_i using two proteins P_j and P_k. Since the number of combinations of three proteins is large for an actual proteome, for instance, the number is $\binom{1000}{3} = 166167000$ if $|\mathcal{P}| = 1000$, we consider only the case that P_i completely contains both proteins, that is, $P_j \cup P_k \subseteq P_i$. Thus, we define the cost generating P_i from P_j and P_k by adding domains appropriately, $P_j + P_k \to P_i$, by

$$cost(P_i, P_j, P_k) = 2 \cdot \lceil \log |\mathcal{P}| \rceil + \lceil \log |\mathcal{D}| \rceil \cdot |P_i - P_j - P_k|. \tag{3}$$

In this equation, the first term of the right-hand side means the length representing P_j and P_k, and the second term means the length representing domains newly appeared in P_i. It should be noted that the 0–1 vector in the case of gene duplication is not needed because all the domains in P_j and P_k are used in P_i.

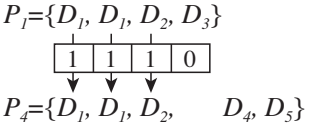

Fig. 1. Illustration of $cost(P_i, P_j)$ for $P_1 = \{D_1, D_1, D_2, D_3\}, P_4 = \{D_1, D_1, D_2, D_4, D_5\}$, $\mathcal{P} = \{P_1, P_2, P_3, P_4\}$, and $\mathcal{D} = \{D_1, D_2, D_3, D_4, D_5\}$. The rectangle denotes a 0–1 vector representing absence (0) or presence (1) of domains.