Libertas Academica
FREEDOM TO RESEARCH

Open Access: Full open access to this and thousands of other papers at http://www.la-press.com.

Cancer Informatics

Supplementary Issue: Network and Pathway Analysis of Cancer Susceptibility (A)

Network Analysis of Circular Permutations in Multidomain Proteins Reveals Functional Linkages for Uncharacterized Proteins

Donald Adjeroh[1], Yue Jiang[2], Bing-Hua Jiang[3] and Jie Lin[2]

[1]Computer Science and Electrical Engineering, West Virginia University, Morgantown, WV, USA. [2]Faculty of Software, Fujian Normal University, Fuzhou, Fujian, China. [3]Pathology, Anatomy and Cell Biology, Thomas Jefferson University, Philadelphia, PA, USA.

ABSTRACT: Various studies have implicated different multidomain proteins in cancer. However, there has been little or no detailed study on the role of circular multidomain proteins in the general problem of cancer or on specific cancer types. This work represents an initial attempt at investigating the potential for predicting linkages between known cancer-associated proteins with uncharacterized or hypothetical multidomain proteins, based primarily on circular permutation (CP) relationships. First, we propose an efficient algorithm for rapid identification of both exact and approximate CPs in multidomain proteins. Using the circular relations identified, we construct networks between multidomain proteins, based on which we perform functional annotation of multidomain proteins. We then extend the method to construct subnetworks for selected cancer subtypes, and performed prediction of potential linkages between uncharacterized multidomain proteins and the selected cancer types. We include practical results showing the performance of the proposed methods.

KEYWORDS: circular patterns, multidomain proteins, cancer, functional annotation

SUPPLEMENT: Network and Pathway Analysis of Cancer Susceptibility (A)

CITATION: Adjeroh et al. Network Analysis of Circular Permutations in Multidomain Proteins Reveals Functional Linkages for Uncharacterized Proteins. *Cancer Informatics* 2014:13(S5) 109–124 doi: 10.4137/CIN.S14059.

RECEIVED: July 8, 2014. **RESUBMITTED:** September 23, 2014. **ACCEPTED FOR PUBLICATION:** September 24, 2014.

ACADEMIC EDITOR: JT Efird, Editor in Chief

TYPE: Methodology

FUNDING: This work was support in part by grants from the US National Science Foundation (#IIS-1236983), the National Natural Science Foundation of China (#61472082), the Natural Science Foundation of Fujian Province of China (#2014J01220), and the US National Institutes of Health (#R01ES020868, #R21CA175975). The authors confirm that the funder had no influence over the study design, content of the article, or selection of this journal.

COMPETING INTERESTS: Authors disclose no potential conflicts of interest.

CORRESPONDENCE: linjie891@163.com

Paper subject to independent expert blind peer review by minimum of two reviewers. All editorial decisions made by independent academic editor. Upon submission manuscript was subject to anti-plagiarism scanning. Prior to publication all authors have given signed confirmation of agreement to article publication and compliance with all applicable ethical and legal requirements, including the accuracy of author and contributor information, disclosure of competing interests and funding sources, compliance with ethical requirements relating to human and animal study participants, and compliance with any copyright requirements of third parties. This journal is a member of the Committee on Publication Ethics (COPE).

Introduction

Given the complex nature of multidomain proteins, it comes as no surprise that they will be involved in very complicated diseases such as cancer. Various studies have implicated different multidomain proteins in cancer. Examples here include the BRICHOS superfamily,[1,2] and the BCL-2 family.[3–5] However, to our knowledge, there has been little or no detailed study on the role of circular permutations (CPs) in multidomain proteins in cancer. Yet, circular proteins and CPs in proteins are becoming of increasing interest, especially given their role in the structure, function, folding, and stability of proteins.[6,7] In a circular (or cyclic) protein, the traditional N- and C-termini are joined, resulting in a protein sequence with neither a beginning nor an end. The cyclotides is a typical example of a naturally occurring family of cyclic proteins in the plant kingdom. Cyclotides are known to play a major role and provide important functions in terms of plant defense against insects and other pathogens.[7–9] Their cyclic structure is known to be an important factor in their unusual stability.[7] Other common examples of cyclic proteins are the bacteriocins, small antimicrobial peptides with 30–70 residues produced by bacteria,[10–12] cyclosporins found in fungi,[13] and the primate rhesus θ-defensin-1[14] with antibacterial properties for the immune system of macaque monkeys.

A CP involves the modification of a protein, first by joining the N- and C-termini to form a circular protein,

and then creating a new N- and C-termini by splitting the circular protein at a different location. Thus, the new sequence formed will be a circularly permuted version of the original sequence. The earliest observation of naturally occurring CPs in proteins was reported by Cunningham et al,[15,16] who showed that the amino acids of the protein concanavalin A (con A) was a CP of another homologous protein, lectin favin. Lindqvist and Schneider[17] listed several other example proteins with CPs, such as bacterial β-glucanases, α-1,3 and α-1,6 glucansynthesizing glucosyltransferases, transaldolose, the C2 domain, and saposins with a structure similar to the bacteriocins.[10,11] CPs in DNA methyltransferases were earlier studied by Jeltsch and Bujnicki.[18,19] Since then, various other CPs have been found in a diverse family of proteins, involved in a diverse array of functions.

Block rearrangements based on domains are common in protein evolution and adaptation.[6,20,21] Thus, CPs can also occur at the block level, in terms of protein domains, rather than just protein sequences. Weiner et al.[22] argued that the conservation of catalytic centers and structural elements in artificial permutations that maintain the same function as the original sequence suggests that CPs are more likely to be block-based at the level of functional domains, rather than at the level of amino acid sequences. Thus, they proposed an algorithm for detecting domain-level CPs in multidomain proteins.[22,23] Han et al.[24] reported that multidomain proteins occupy >50% of all proteomes, with eukaryote proteomes containing a higher proportion of multidomain proteins than prokaryote proteomes. The preponderance of multidomain proteins in complete genomes,[25–27] and the rate at which complete genomes of several organisms are being sequenced, provides another important motivation for a deeper study of CPs in multidomain proteins.

Figure 1 shows examples of multidomain zinc finger protein sequences that are related by CPs, along with their domain block structures. In this figure, one protein (ZNF146) appears as an exact CP inside the other (ZNF680). Also, both proteins form a pair of matching 1-approximate CP. We note that, without considering CPs, these matches cannot be found using standard exact or approximate pattern matching.

There is still a debate on the origins, evolution, and prevalence of naturally occurring CPs in proteins. Various mechanisms have been suggested[22] based on evolutionary genetic events, such as duplication and deletion,[18] fusion/fission events,[22] and "cut-and-paste" mechanism[19] involving plasmids. Others have proposed post-translational modifications.[16] Craik[7] described other possible mechanisms. Further, the complete role of circularization in proteins is not yet fully understood.[7] However, circular proteins have been known to be involved in several important functions, such as plant defense against insects and other pathogens,[7–9] providing stability,[7] and support of antibacterial activities for the immune system in macaques monkeys.[14] Cyclization was suggested to be critical for certain activities of the cyclic proteins, as engineered acyclic permutants of naturally occurring proteins with the same general structure were shown to exhibit loss of hemolytic activity.[29] The C2 domains (which are topologically distinct from Synaptogamin I but related by CPs)[30] are known to be involved in signaling and transduction in eukaryotes,[17] and thus could play a role in certain cancers. The WD-Repeat protein (WIPI protein family) is implicated in various human cancers, such as skin, kidney, and pancreatic cancers. The WIPI family contains beta-propellers with ring structures, which are stabilized by CPs.[31] The PDZ domain is another multidomain family that is involved in cancer.[32] Folding and misfolding of CP variants of the PDZ domain and the impact on the stability of their structure and function were studied by Hultqvist et al and Ivarsson et al.[33,34] Chemically synthesized retrocyclin, a defensin-like molecule, was found to possess possible anti-HIV properties.[8,35]

Given the growing importance of cyclization and CPs in proteins, there is a need for efficient algorithms for their detection and analysis. Further, the preponderance of multidomain proteins, coupled with the prevalence of CPs in such proteins underline the importance of considering multidomain proteins in such an algorithmic study. For block-based multidomain proteins for instance, there are key challenges posed by the specific nature of the domain sequences, such as the very large alphabets involved, and the variability in sequence lengths (in ProDom, the multidomain protein database,[28] sequence lengths vary from as small as 2 domains, to as large as 568, with an alphabet size of almost 2 million). Most of the available algorithms for detecting CPs are still relatively slow, often running in times that are quadratic or cubic with respect

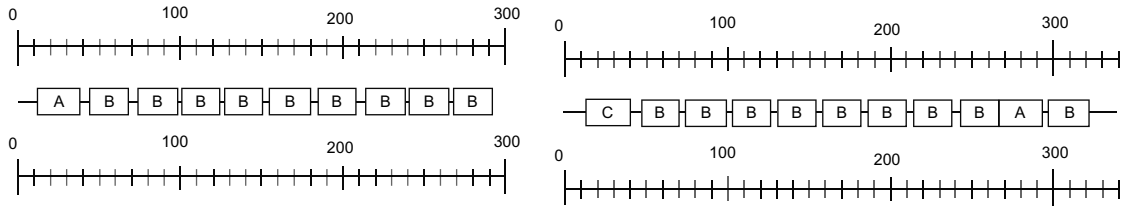

Figure 1. Example of multidomain proteins that are related by CP. Multidomain protein Q5RFP4 (Zinc finger (ZNF146) from *Pongo abelii*) with domain block sequence *ABBBBBBBBB* occurs as an exact CP inside Q8NC79 (ZNF680, *Homo sapiens*) with domain sequence *CBBBBBBBBAB*. Notice also that both proteins form a *k*-approximate CP (with $k = 1$). Codes inside the blocks denote protein domain IDs as used in the protein domain database (ProDom).[28] Key: A:PD057131, B:PD000003, C:PD915601. Schematic for linear domain block structures generated from the ProDom website.

to the total length of the sequences in the database. With such algorithms, an all-against-all search of possible CPs of a protein contained within other proteins becomes almost infeasible, even with multiple processors. The exponential growth in the size of available genomic datasets and the rapidly increasing rate at which complete genomes are being sequenced imply an urgent need for improved algorithms for whole-genome analysis of cyclic permutations in proteins. Such algorithms should be robust and efficient on both the direct protein sequences and on block-based multidomain representations with vastly increased alphabet sizes. They should be able to support sophisticated searches and comparisons, such as the all-against-all CP problem.

In this paper, we first propose algorithms for rapid detection of CPs in multidomain proteins, suitable for scanning large genomic databases for all-against-all circular pattern matches. Building on the results, we study networks of multidomain proteins constructed based on their shared CPs. Using this network, we investigate a method for functional annotation of uncharacterized multidomain proteins. We then extend the method to study potential association of some unknown multidomain proteins with certain types of cancer.

Background and Related Work

Basic notations. Let $T = \alpha\beta\gamma$, for some strings α, β, and γ (α and γ could be empty). The string β is called a *substring* of T, α is called a *prefix* of T, while γ is called a *suffix* of T. We will also use $t_i = T[i]$ to denote the i-th symbol in T. We let $P = P[1\ldots m]$ be the pattern string that needs to be found in T. Let *SeqDB* be a sequence database with Z sequences. The total number of symbols in *SeqDB* is N. Let *SeqDB*[i] be the i-th sequence in *SeqDB*, where $0 < i \leq Z$. Let m_i be the length of *SeqDB*[i]. Let $N = \sum_{i=1}^{Z}(m_i)$ be total number of symbols in *SeqDB*. The average number of symbols per sequence in *SeqDB* is $m_a = N/Z$. Let k be the allowed error for an approximate match.

Circular pattern matching. Computing similarity (or dissimilarity) between two strings is an important problem in general sequence analysis,[36–38] pattern recognition,[39,40] and biology.[41,42] The major computational tool used to study CPs is based on solutions to the circular pattern matching (CPM) problem.

Two strings are CPs of each other if one can be transformed to the other through a sequence of circular shifts. A circular shift is a mapping f: $\Sigma^* \times [0, r-1] \rightarrow \Sigma^*$, $f^t(c_1\ldots c_r) = c_{t+1}\ldots c_r c_1\ldots c_t$, where $0 \leq t \leq r-1$ and r is the length of string $c_1c_2\ldots c_r$. Thus, $f^0(c_1\ldots c_r)$ corresponds to the original string. The CPM problem is to find all occurrences of the pattern P and/or its circular shifts in the text T. Let $[s]$ be a set of circular shifts of string s, then $[s] = \{f^i(s) | 0 \leq i \leq |s| - 1\}$. Given two circular strings s_1 and s_2, the circular edit distance between s_1 and s_2, is the minimum number of edit operations needed to transform one member of $[s_1]$ to a member of $[s_2]$. This is defined as $ED_c(s_1, s_2) = \min\{ED(f^i(s_1), f^j(s_2)) | 0 \leq i \leq |s_1| - 1$ and $0 \leq j \leq |s_2| - 1\}$, where $ED(A, B)$ is the standard edit distance between strings A and B. Thus, the dissimilarity between the two strings in a circular shift is a function of the circular edit distance between them.

Algorithms for the CPM problem have been proposed for the exact CPM problem,[36,43,44] and for the k-approximate CPM (ACPM) problem.[45–49] More specifically, given an m-length circular pattern P and an n-length text T, the approximate CPM (ACPM) problem seeks to find k-approximate matches between circular pattern $[P]$ and text T. The naïve method for the ACPM problem is to use each of the circular strings $f^t(P)$ to calculate the edit distance between T and $f^t(P)$, $0 \leq t \leq m-1$. Thus, there are m steps to run the dynamic programming procedure. The time complexity of a naïve algorithm to compute ED($[P]$,T) is $O(m^2n)$, where $m = |P|$, $n = |T|$. Maes[45] published a "divide and conquer" algorithm to compute ED($[P]$, T) in $O(mn \log m)$.

CPM problems in protein sequences. A number of studies have been reported on algorithms for detecting CPs for protein sequences.[41,42,50] The first method[51] used the dot matrix and human visualization to identify circular relationships between pairs of protein sequences. Altschul et al[52] used a dictionary method to find short fragments common to the protein sequence pairs and used human visualization to report the best local matches. Uliel et al.[53,54] introduced a method to detect CPs in protein sequences using global alignment.[55] They gave an $O(n^3)$ time algorithm to find all the locations in T that match a CP of P. They also proposed a faster greedy algorithm that requires $O(n^2)$ time, but which could miss some valid CPs in the text T. Weiner et al.[22,23] proposed another greedy method that runs in $O(n^2)$ time. They focused on circular multidomain proteins, where the alphabet is now composed of the protein domain blocks, rather than traditional protein symbols. Thus, $|\Sigma|$ could be quite large, in the order of 20^q, where q is the length of the domain blocks. This was the first application of the CPM problem in studying multidomain proteins. However, they did not consider the problems posed by the expanded alphabet. The methods of Uliel et al.[53,54] and Weiner et al.[22,23] each required an $O(mn)$ space.

More fundamentally, both groups[22,23,53,54] that have studied CPM in protein sequences have focused on whole sequence comparison with another whole sequence. In their experiments, they have to group the protein sequences based on their specified lengths, and used the dissimilarity in lengths for initial pruning. These methods ignored the fact that a shorter circular protein sequence could be part of the functional region of a much larger multidomain protein. This, however, could be a key consideration in function prediction for multidomain proteins. Further, as with the more theoretical algorithms for the ACPM problem,[45–47,56] the methods for protein sequences[22,23,53,54] also only considered the existential version of the ACPM problem (ie, simply report **true** or **false** on whether P and T are CPs). None of the CPM methods described have considered the more challenging enumerative

version of the ACPM problem (ie, given P and T, find every substring of T that forms a CP with P). Solution to this variant is mandatory for our goal of studying potential functional linkages between multidomain proteins and some unknown proteins.

Other recent work on CPs have studied structure alignments for circular proteins.[57–61] Our focus is on rapid and efficient search for CPs, rather than on alignments. We address the enumerative version of the ACPM problem, and use the results to study functional associations between multidomain proteins. We also apply our results to the problem of predicting cancer-related multidomain proteins. Our circular pattern detection method is based on a very different approach, using indexing on suffix arrays.

Materials and Methods

Datasets. The major sources of data used are the protein domains in the ProDom database, protein annotation in the gene ontology (GO) database, and information on proteins with known association with cancer.

Protein domain database. Most proteins consist of several domains. The same protein domain may occur in many related proteins. Our experiments were performed using multidomain proteins in ProDom,[28] a database of known protein domains. Each domain is represented as a unique symbol, thus a multidomain protein is viewed as a sequence of such symbols. The length of the domain representation is generally much smaller than the original protein sequence, but the size of alphabets has increased drastically. Pagel et al.[62] constructed a protein domain interaction network using data from ProDom.

GO database. The GO project (http://www.geneontology.org/) provides a description and annotation of genes and protein products in different databases including the known functions of the genes. Currently, the GO Consortium includes many databases such as GeneDB (http://www.genedb.org/), UniProtKB-Gene Ontology Annotation (UniProtKB-GOA) (http://www.ebi.ac.uk/GOA/), and FlayDB (http://flybase.bio.indiana.edu/). The ProDom database provides the Accession Number for the parent protein of each domain. The Accession Number is also provided for UniProtKB-GOA. This establishes a connection between entities in ProDom and their corresponding entities in GO. Thus, we can use this relationship to obtain the GO terms used to describe the protein function.

Cancer Protein Datasets. The Cancer Resource[63] is a database of proteins known to be associated with cancer. The database contains information on 25 general cancer categories. For our experiments, we downloaded and analyzed protein data on five cancer categories, namely, bone, colon, lung, skin, and breast. The Cancer Resource dataset is available at on the web (http://bioinf-data.charite.de/cancerresource/). To verify some of the novel cancer-related proteins predicted by our algorithm, we performed literature search using PubMed, and also checked for entries in the publicly available *Atlas of Genetics and Cytogenetics in Oncology and Haematology* (http://atlasgeneticsoncology.org).

Algorithms for CPM. In this section, we present our algorithms for the ACPM problem. First, we introduce APM-VIA-LIS (**Algorithm 0**), a generic approximate pattern matching algorithm that uses longest increasing subsequences (LIS) to find an approximate match of a pattern P in text T. The algorithm does not handle CPM. Next, we propose algorithms for the ACPM problem and analyze their complexity. We will start with a simple greedy algorithm and then consider a suffix-array-based q-gram algorithm for the ACPM problem. The LIS method for pattern matching will be used in these algorithms. When we use this algorithm to solve the ACPM problem, we have to consider all the circular shifts $f^t(P)$, $0 \leq t \leq m-1$, to match the text T.

The LIS method utilizes the LIS algorithm[36,64] to calculate the longest common subsequence (LCS)[36,65] between two sequences. The verification process checks whether the edit distance between these two sequences is less than k. When we calculate LIS and LCS, each matched symbol will occur in the LCS. The algorithm uses a mapping table (*mapTable*) that stores the positions in P of each symbol in the alphabet in decreasing order.

For each matched symbol, we obtain its positions of occurrence in the two sequences. We can use these positions to check the number of edit operations between two matched symbols. Thus, the algorithm reports the edit distance between these two sequences. The time complexity for this algorithm is $O(\frac{mn}{|\Sigma|}\log m)$. When $|\Sigma| \geq m$, as in the case for multidomain proteins, the time complexity will be $O(n \log m)$.

Algorithm 0: Generic approximate pattern matching using LIS

APM-VIA-LIS(T, P, k)

```
1 Build the mapping table mapTable
2 seq ← NULL, n ← |T|
3 for (i ← 1 to n) do
4  seq ← seq ○ mapTable[T[i]]
5 end for
6 Generate LIS from seq
7 Calculate LCS between T and P using the LIS
8 if verify(LCS,k) is true then
9  return matching string
10 end if
```

Algorithm 1: Greedy ACPM Algorithm. Algorithm ACPM-greedy **(Algorithm 1)** compares any two sequences for possible ACPM using Algorithm APM-VIA-LIS (Algorithm 0). First, the algorithm will choose two sequences from the database, one is considered as text T and the other as a circular pattern P. The algorithm executes two steps. The first step creates a new pattern from P, viz: $PP \leftarrow P[1\ldots m] \circ P[1\ldots m-1]$ where "○" is the concatenation operator. The second step calculates the LCS between PP and T and returns the LCS string *lcs*. This procedure is performed in line 5. This step also verifies

the approximate pattern matching with parameter k, using Algorithm APM-via-LIS.

This algorithm is simple, but greedy (suboptimal): it finds only one occurrence of the pattern, and may not detect all the circular patterns that occur in the text. If there is more than one LCS in T, this method could miss some matches.

Algorithm 1: ACPM with Greedy Algorithm

ACPM-greedy(*SeqDB*, *Z*, *k*)

```
1 for (i ← 1 to Z) do
2  for (j ← 1 to Z) do
3   P ← SeqDB[i], m ← |P|, PP ← P[1…m] ○ P[1…m − 1]
4   T ← SeqDB[j], n ← |T|
5   APM-via-LIS(T, PP, k)
6  end for
7 end for
```

Time complexity analysis. For the complexity analysis, we need to consider two cases: (1) For the case of searching for one sequence against a group of sequences (loop from line 2 to line 6), the time complexity is $O(\sum_{i=1}^{Z} n_i \log m) = O(N \log m)$, where $N = \sum_{i=1}^{Z} n_i$ is the total length of all sequences used, Z is the number of sequences, and n_i is the length of the i-th sequence in *SeqDB*. (2) For the case of searching for a CP among a group of sequences (loop from line 1 to line 7), the time complexity is $O(ZN \log m_m))$, where m_m is the length of the longest sequence. The final time complexity is $O(N^2 \log m_m)$, since $Z = O(N)$. In our experiments with multidomain proteins using Prodom[28], $N \approx 7.3Z$.

Algorithm 2: ACPM with q-grams and suffix array. The q-gram approach[38,66] is a two-phase method that can be used to reveal all approximate patterns. The first phase is the *hypothesis phase*, which determines all potential matching positions using only q-length (q-gram) substrings of P and T. In the second phase, the *verification phase*, the algorithm verifies each potential matching position to report the correct matches. The basis of the q-gram approach is the fact that for any two strings that are approximate matches, there must exist some exact matching sub-region(s) between them. Being a filtration approach, the hypothesis generation phase is typically fast, while the verification stage is typically slower. However, verification will be applied only to a few locations corresponding to potential matching regions in the text. Thus, the overall computational cost will depend on the number of hypothesis generated. We use the suffix array data structure for rapid hypothesis generation, and then verify each hypothesis using the generic APM-via-LIS algorithm.

Figure 2 shows the number of hypotheses generated for different q values, using the ProDom database.[28] Here, we used $N = 10^6$. We notice that when q increases, the number of hypotheses will decrease very rapidly. At around $q \geq 3$, the number of hypothesis will typically reduce to $O(N)$.

Algorithm ACPM-qgram (**Algorithm 2**) shows the process. Lines 1–7 denote the preprocessing stage. This stage

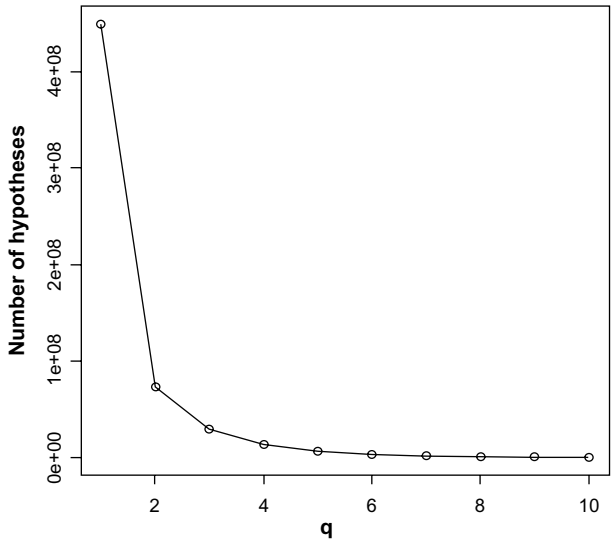

Figure 2. Variation of the number of hypotheses generated using q-grams.

constructs a long concatenated sequence, *seq*, using all the sequences so far encountered in *SeqDB*. It also builds an auxiliary array *pos*. This array is used to maintain the relationship between positions in *seq* and *SeqDB*. Line 8 constructs the suffix array for the concatenated sequence.

Lines 9–24 use a loop to generate all the hypotheses for the q-gram method using the longest common prefix *LCP* array. Lines 11–13 determine the candidate matching positions that have the same q-gram prefix. Line 14 considers each pair of candidate positions obtained with the current q-gram for verification.

Lines 15–22 perform the verification. We use the ACPM-via-LIS algorithm to verify the approximate patterns. Constructing the circular pattern is the same as in the previous algorithm. We enumerate the m circular patterns from a sequence one by one. We construct *subT* from the second sequence T as follows. Assume the q-gram occurs in position y, so let *subT* be the substring of T which includes $T[y \ldots y + q - 1]$ and the length is $(m + k)$. So text will be $T[y - m - k + q \ldots y + q - 1]$, $T[y - m - k + q + 1 \ldots y + q]$, … $T[y \ldots y + m + k - 1]$. There are $(m + k - q)$ number of such substrings.

Algorithm 2: ACPM with q-grams and Suffix Array

ACPM-qgram(*SeqDB*, *N*, *Z*, *q*, *k*)

```
1 seq ← NULL, pos ← NULL, s ← 1
2 for (i ← 1 to Z) do
3  seq ← seq ○ SeqDB[i]
4  for (j ← 1 to m_i) do
5   pos[s] ← i, s ← s + 1
6  end for
7 end for
8 <SA,lcp> ← BuildSA(seq)
9 for (i ← 1 to N) do
10 Candidates ← {}
```

```
11 do while (lcp[i] ≥ q)
12   Candidates ← Candidates ∪ {i}, i ← i + 1
13 end do
14 for each Pair {x, y} ∈ Candidates do
15   P ← SeqDB[pos[SA[x]]], m ← |P|
16   T ← SeqDB[pos[SA[y]]], n ← |T|
17   for (j ← min(1, y − m − k + q) to y + m + k − q) do
18     subT ← T[j...j + m + k − 1]
19     for (h ← 1 to m) do
20       APM-via-LIS(subT, f^h (P), k)
21     end for
22   end for
23 end for
24 end for
```

Complexity analysis. The required suffix array for the entire database can be constructed in $O(N)$ time and space, using any of several linear-time linear-space suffix array construction algorithms.[38,67–69] After suffix array construction, hypothesis generation is performed in $O(m \log |\Sigma|)$ for each m-length pattern, or a total time of $O(N \log |\Sigma|)$ for all the database sequences. The time complexity for the LIS algorithm to verify one pattern vs one substrings from the text that includes one matched q-gram is $O((m + k - q) \times m \log m)$. Since $k \leq O(m)$ and $q \leq O(m)$, the time complexity is $O(m^2 \log m)$. Each pair in the same group has $O(m)$ circular pattern operations, thus the time complexity for verifying each pair is $O((m^2 \log m) \times m) = O(m^3 \log m)$. There are r groups and group i has n_i elements and there are $\sum_{i=1}^{r} n_i^2$ pairs. The total complexity is $O(m^3 \log m \times \sum_{i=1}^{r} n_i^2)$. The worst case occurs when $r = 1$ with time complexity of $O(N^2 m^3 \log m)$. For the average case, m is the average length of the sequences. Then, the time complexity will be in $O(Nm_a^3 \log(m_a))$, where $m_a = \frac{N}{Z}$.

Comparison with other ACPM algorithms. The time complexity of our ACPM-QGRAM algorithm is $O(m_m^3 N^2 \log m_m)$ in the worst case, where m_m is the length of the longest sequence in the database. On average, the time complexity is in $O(m_a^3 N^2 / |\Sigma|^q)$, where m_a is average sequence length, N is the total length of the sequences, and $q = \lfloor \frac{m}{k+1} \rfloor$. When q increases, $O(N^2/|\Sigma|^q)$ will be reduced to $O(N)$, since typically, $|\Sigma|^q > O(N)$. Comparing the ACPM-QGRAM with other CPM algorithms, our q-gram algorithm does not fare very well when m is large (eg, $m = O(N)$). In this case, the Maes' algorithm[45] will produce a better performance of $O(N^2 m_m^2 \log m_m)$ in the worst case. However, the performance is very competitive on average. When $\frac{m}{k+1}$ increases and m is not very large, the ACPM-QGRAM algorithm will run in $O(m_a^3 N \log m_a)$, where $m_a = \frac{N}{Z}$. This can be treated as a constant ($\frac{N}{Z} \approx 7.3$ for the case of multidomain proteins in our dataset). Therefore, under such conditions, the ACPM-QGRAM algorithm is a linear-time algorithm on average. This can be compared with Maes' algorithm, which runs in $O(m_a^2 N^2 \log m_a)$ on average, or in $O(N^2)$ if we assume that m_a is a constant, when compared with N. Thus, for the average case, the proposed ACPM-QGRAM algorithm is better than the other algorithms previously proposed for the ACPM problem.

Multidomain protein networks using circular patterns. We explore the use of our proposed CPM algorithms on the problem of analyzing multidomain protein sequences. Based on the circular patterns found by our algorithms, we construct a multidomain protein network by connecting different multidomain proteins that are found to be associated by some matching circular patterns. We note that Pagel et al.[62] introduced a tool for analyzing potential relationships between proteins using the protein domain network. This network was based on protein domain interaction networks. They built a web resource to explore Domain Interaction MAp (DIMA). In this network, the nodes are the protein domains and the edges are the interactions between two protein domains. In our work, network formation is based primarily on cyclic relationships between multidomain proteins.

More specifically, we construct a directed graph showing a relationship network among the multidomain proteins. Each node (vertex) in the network represents a multidomain protein, while an edge between two nodes represents a circular relationship between the nodes. For a given node, we define the *in-edges* and *out-edges* as follows. If a protein sequence P_1 is a circular pattern in protein sequence P_2, then there is an out-edge from P_1 to P_2. Conversely, there is an in-edge from P_2 to P_1.

Protein function prediction. The network described above provides an important framework for studying potential functional linkages between the multidomain proteins in our dataset. First, Table 1 provides an intuition on how an analysis of the network of CP relations could expose potential associations between multidomain proteins. The table shows the protein functions for the Top 20 highest degree proteins. We notice that 15 of the 20 proteins have exactly the same functions. Four of the proteins (ranked 3, 4, 9, and 12, respectively) do not have entries in the GO database. Protein Q5NU40 (rank 14) has a record in GO database, but no function has been assigned to it in GO. Thus, the functions of these five proteins are not yet known. It is expected that some of these five proteins with no known function are likely to have the same or similar functions as the other 15 proteins.

We use the z-score as a measure of significance of the functional relatedness between two proteins. For a given random variable x, the z-score is defined as: $z = \frac{x - \mu_x}{\sigma_x}$, where μ_x is the mean and σ_x is the standard deviation. Then, protein function prediction is performed in two steps using the z-scores. First, we rank the nodes in the network (the proteins) based on their z-scores for in-degree and out-degree. We then identify the proteins that have degree z-scores above a threshold, or those that are ranked within the top K_1 nodes for further analysis in the second stage of function prediction. In the second stage, we predict the function(s) for each protein that is selected in the first stage. To predict the function for a protein, say P_A using the network, we first enumerate all the proteins in

Table 1. Top 20 highest degree proteins with GO function.

RANK	COUNT	AC NUMBER	GO DESCRIPTION
1	23353	Q7VMZ1	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
2	23344	Q9CPC5	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
3	23338	Q3EG14	Protein not found in GO
4	20508	Q33HH1	Protein not found in GO
5	20446	Q47AY9	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
6	20446	Q4UQ62	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
7	20446	Q8P4K7	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
8	20446	Q8PG73	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
9	20415	Q426Q5	Protein not found in GO
10	20398	Q3BNR9	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
11	20393	Q73PA3	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
12	20273	Q50XK7	Protein not found in GO
13	20246	Q66C16	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
14	20244	Q5NU40	No function in GO
15	20244	O32748	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
16	20199	Q30UI1	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
17	20177	Q5NU41	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
18	20150	Q3MAZ4	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
19	20133	Q5VLQ9	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity
20	20118	Q6NEY3	nucleotide binding; ATP binding; ATPase activity; nucleoside-triphosphatase activity

the respective in-edge and out-edge sets for protein P_A. Using GO, we identify the functions for each protein in the two sets. We then compute the normalized scores (again using z-scores) for the occurrence frequencies for each function identified. We then assign the function for protein P_A as the functional with a z-score above a specified threshold, or the functions ranked within the Top K_2 functions.

Experiments and Results

Setup. We performed experiments using the results of the proposed algorithms to study CPs in multidomain proteins, looking for potential CP relationships between pairs of such proteins. We use the results of the proposed algorithms to study potential functions linkages between uncharacterized or unknown proteins and known multidomain proteins. In the experiments, each domain is a symbol in the alphabet. Thus, $|\Sigma|$ is quite large, and often in $O(N)$. Example, for ProDom,[28] $|\Sigma| \approx 1.99 \times 10^6$, $Z = 973{,}686$, and $N = 7{,}075{,}729$. Thus, total number of protein pairs $= 9.48 \times 10^{11}$ and $m_a \approx 7.3$. We observed $m_m = 568$.

The experiments were performed using a DELL PC, with 4 × 2.67 GHz CPU, and 8G memory, running Ubuntu 10.10 Linux operating system. All programs were compiled using gcc.

Speed and completeness. We ran the three proposed algorithms on the ProDom dataset and use the results to analyze the relationship between multidomain proteins. The ACPM-qgram algorithm was executed using two different parameter settings, namely $q = 1$ and $q = 2$. When $q = 1$, the result is complete. When $q = 2$, the result is suboptimal (incomplete). We use the complete results as a benchmark to compare with the results from the greedy algorithms. In all experiments, we set the error parameter $k = 1$.

ACPM-greedy algorithm is faster than the other ACPM algorithms, but the result has low accuracy (around 50%). Algorithm ACPM-LIS was the slowest algorithm. Figure 3 shows the practical time required by these three algorithms, where the q-gram algorithm has two instances, $q = 1$ and $q = 2$. A comparison of the outputs of the algorithms provides some insight in their overall performance. There are 29,625,738 relations in the complete result. ACPM-greedy only found 15,075,729 relations (51%) of the total. ACPM-qgram algorithm with parameter $q = 2$ found 29,345,380 relations (99% of the total).

We also implemented a hybrid algorithm where an exact CPM (ECPM) algorithm[48] was applied first and then followed by the ACPM-qgram algorithm with parameter $q = 1$. We obtain the complete result, while the running time was reduced from 41 to 14 hours. Table 2 shows a breakdown of all the CPs found by the proposed algorithm, using the ProDom dataset. The non-CP matches correspond to matches using the original patterns, without circular shifts. That is, matches using shift function $f^t()$, with parameter $t = 0$. We can see that

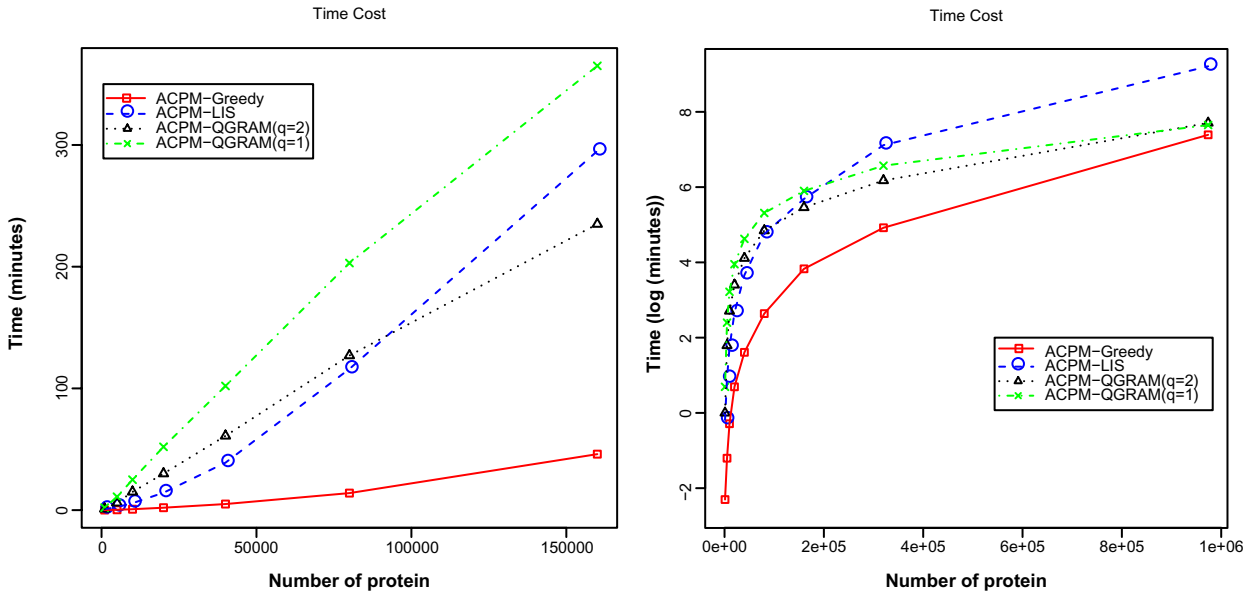

Figure 3. Execution time for the proposed ACPM algorithms.

the CPs are much more predominant when compared with the non-CP-matches.

Statistics of CP Network in ProDom. First, we investigate the nature of the multidomain protein network formed between protein that share some CPs in the ProDom dataset. Figure 4 shows the log plot of the degree distribution for the network. Figure 4A shows the degree distribution for all vertices in the network, while Figure 4B shows the degree distribution of the Top 100 highest degree nodes.

The results show the power and significance of our basic approach, addressing the all-against-all variant of the ACPM problem. Each protein sequence is not only used as a pattern to search against the other protein sequences, but also used as text to be searched on using the other protein sequences in the database. Of the 973,686 multidomain proteins remaining in our dataset after preprocessing, 799,044 (85%) contain at least one other protein sequence as a circular pattern; 424,888 protein sequences (43.6%) were found to be a pattern in some other protein sequences; 374,279 protein sequences (38.4%) have both out-edges and in-edges. About 50,609 protein sequences (5.2%) only have out-edges; while 424,765 protein sequences (43.6%) only have in-edges. The average degree of this graph was 23, with an average out-degree of 46 and an average in-degree of 24.5. We note that traditional ACPM algorithms, such as those of Weiner et al and Uliel et al,[22,23,53,54] which do not consider the all-against-all problem, will find CPs for only the 374,279 sequences that have both in-edges and out-edges.

Figure 5A shows the number of directly connected pairs in the Top K highest degree proteins, with K = 10, 20,..., 1000. Let the Top K highest degree proteins be vertices of a subgraph, the number of directly connected pairs is the number of edges. We define a ratio ρ_K as follows: $\rho_K = \frac{\#\ of\ total\ edges}{\#\ of\ edges\ in\ Top{-}K\ complete\ subgraph} = \frac{\#\ of\ observed\ edges}{\frac{1}{2}\times K\times(K-1)}$. Figure 5(B) shows the ratio ρ_K for the Top K proteins. When K is <460, the ratio ρ_K stays stable at around 0.5. When K is >460, the ratio ρ_K starts to decrease. Thus, in this graph, the top 460 highest degree proteins have higher relations.

Predicting functions for uncharacterized proteins. We first tested the function prediction using nine multidomain proteins in the ProDom dataset, with known functions in GO. Table 3 shows the prediction results on nine sample multidomain proteins using the union of the functions from the proteins in the in-edge and out-edge sets at different thresholds on the z-scores. Table 4 shows the equivalent results using the intersection. Expectedly, using the intersection led to more precision, but with less recall (notice the many missed prediction, with many empty cells in the table).

We conducted a larger experiment to predict the protein functions in the Top 500 highest degree proteins in our network. Of these, 156 proteins were not found in the GO database. Thus, performance analysis was based on the remaining 344 multidomain proteins that have function annotation in GO. Prediction performance was measured in terms of *precision*, *recall* and the *F-measure*, where *FP* is the number of false positive; *FN* is the number of false negative; *TP* is the number of true positive. These were computed as follows: $recall = \frac{TP}{TP+FN}$; $precision = \frac{TP}{TP+FP}$; $F\text{-}measure = 2\times\frac{recall\times precision}{recall+precison}$.

Table 2. Circular patterns found in the ProDom database.

MATCH TYPE	CP-MATCHES	NON-CP MATCHES	TOTAL
exact PM	1706800	2626323	4333123
1-approx PM	24679013	613602	25292615
Total	26385813	3239925	29625738

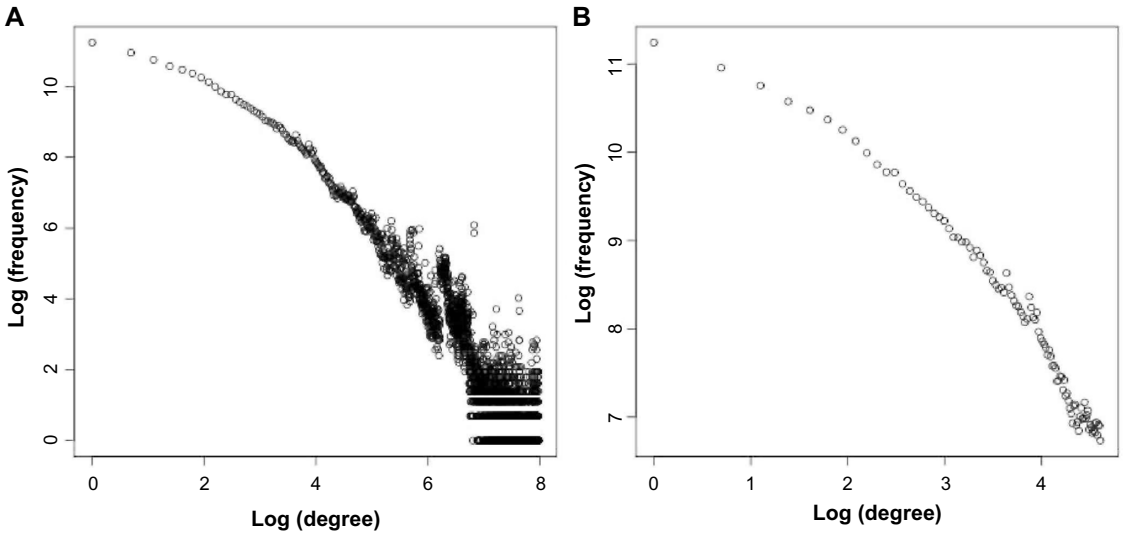

Figure 4. Degree distributions in the network of multidomain proteins constructed based on the circular patterns they contain. (**A**) Log degree distribution. (**B**) Log degree distribution for Top-100 degree nodes.

Using the *F-measure*, the union method at $z \geq 3$ produced the best results, with a highest *F-measure* of 0.84. Figure 6 shows the overall summary of the performance in function prediction using the proposed algorithms. The figure shows results for function prediction using the Top 10,000 degree proteins in the network. For each protein, we use the GO function annotations (where available) as the ground truth for the prediction results. Of the 10,000 proteins, 6,261 had GO annotations. Thus, the results shown are essentially based on these proteins. Perhaps, more importantly, the performance in function prediction for the 6,261 with GO terms implies that our proposed approach can be used for reliable annotation of the remaining 3,739 proteins that did not have GO annotations.

Predicting novel multidomain proteins associated with cancer. In this experiment, we use the proposed multidomain protein circular relationship network, and tailor the function prediction method described, to specifically focus on prediction of novel multidomain proteins with potential associations with cancer. Using data from the Cancer Resource dataset,[63] we selected five types of cancer (bone, colon, lung, skin, and breast cancers), and studied subnetworks involving multidomain proteins known to be associated with each type of cancer.

Construction of cancer subnetworks. For each cancer type, we construct a corresponding subnetwork using only the multidomain proteins identified in the Cancer Resource dataset. Thus, we determine the proteins that have circular

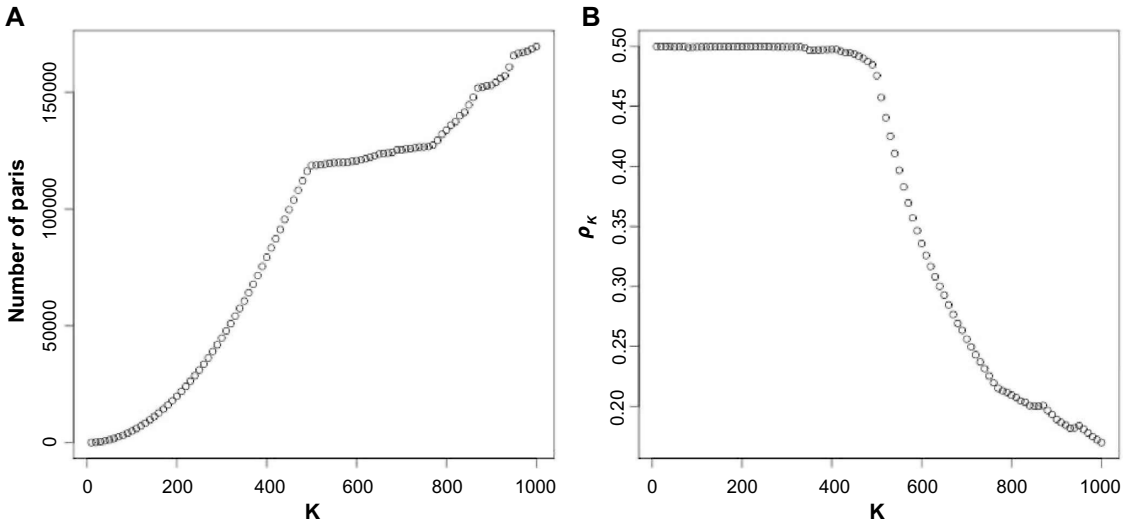

Figure 5. Number of directly connected pairs in Top *K* highest degree proteins. (**A**) Number of directly connected pairs. (**B**) The ratio ρ_K for increasing values of *K*.

Table 3. Predicted functions for nine sample multidomain proteins using the union of functions for known proteins in the in-edge and out-edge sets.

PROTEIN AC NUMBER	FUNCTION (GROUND TRUTH)	PREDICTED FUNCTION ($z \geq 3$)	PREDICTED FUNCTION ($z \geq 2$)	PREDICTED FUNCTION ($z \geq 1$)
Q7VMZ1	GO:0000166	GO:0000166	GO:0000166	GO:0000166
	GO:0005524	GO:0005524	GO:0005524	GO:0005215
	GO:0016887	GO:0016887	GO:0016887	GO:0005524
	GO:0017111	GO:0017111	GO:0017111	GO:0016787
			GO:0042626	GO:0016887
				GO:0017111
				GO:0042626
O32184	GO:0003824	GO:0003824	GO:0003824	GO:0003824
	GO:0005488	GO:0005488	GO:0005488	GO:0004316
	GO:0016491	GO:0016491	GO:0016491	GO:0005488
				GO:0016491
Q2Y7W6	GO:0000156	GO:0000155	GO:0000155	GO:0000155
		GO:0004871	GO:0004871	GO:0004871
Q33CH5	GO:0003723	GO:0003723	GO:0003723	GO:0003723
	GO:0003968	GO:0003968	GO:0003968	GO:0003968
Q30U32	GO:0000156	GO:0000156	GO:0000155	GO:0000155
		GO:0004871	GO:0000156	GO:0000156
			GO:0004871	GO:0004871
O93828	GO:0004585	GO:0004585	GO:0004585	GO:0004585
	GO:0016597	GO:0016597	GO:0016597	GO:0016597
	GO:0016740	GO:0016740	GO:0016740	GO:0016740
	GO:0016743	GO:0016743	GO:0016743	GO:0016743
Q30SN9	GO:0003824			GO:0003824
	GO:0004252			GO:0004252
				GO:0005515
Q2YTY7	GO:0003723			GO:0003723
	GO:0009982			GO:0009982
O78911	GO:0008137	GO:0008137	GO:0008137	GO:0008137
	GO:0016491	GO:0016491	GO:0016491	GO:0016491

relationship(s) with other proteins involved in the same cancer type. For instance, this yields 28 proteins for bone cancer, and 43 proteins for colon cancer. Using these known cancer proteins that are associated by CPs, we then search the larger circular relationship network with all the multidomain proteins in the ProDom database. We thus obtain a larger subnetwork, whereby nodes in the subnetwork are proteins with known association to a given cancer type, or those that are associated with these through a CP relationship. Figures 7 and 8 show the subnetworks for colon and skin cancers, respectively. The subnetworks for colon, lung, and breast cancers are available as supplementary material. Table 5 shows the summary statistics of the subnetworks from the circular relationship network, for each of the five cancer types.

Predicting cancer-related multidomain proteins. Using the cancer-type specific subnetworks, we can now predict which multidomain proteins are most likely to be associated with the given cancer type. This requires only a slight modification of the basic function prediction method described earlier in the Materials and Methods section.

We still use the notion that hubs (nodes with a higher connectivity) are more likely to be important in the subnets. That is, these nodes have more circular relationships in this cancer protein subnetwork.

However, rather than using simple connectivity based on node degree distributions, we measure the node *betweenness centrality*.[70,71] For a given node in a network, the betweenness centrality is defined as the number of shortest paths from each node to all other nodes that pass through the given node. Thus, the betweenness centrality measures both the local and global significance of a node in a given network. We therefore use the z-scores on the node betweeness centrality to rank the

Table 4. Predicted functions for nine sample multidomain proteins using the intersection of functions for known proteins in the in-edge and out-edge sets.

PROTEIN AC NUMBER	FUNCTION (GROUND TRUTH)	PREDICTED FUNCTION ($z \geq 3$)	PREDICTED FUNCTION ($z \geq 2$)	PREDICTED FUNCTION ($z \geq 1$)
Q7VMZ1	GO:0000166	GO:0000166	GO:0000166	GO:0000166
	GO:0005524	GO:0005524	GO:0005524	GO:0005524
	GO:0016887	GO:0016887	GO:0016887	GO:0016887
	GO:0017111	GO:0017111	GO:0017111	GO:0017111
O32184	GO:0003824			
	GO:0005488			
	GO:0016491			
Q2Y7W6	GO:0000156	GO:0004871	GO:0004871	GO:0000155
				GO:0004871
Q33CH5	GO:0003723		GO:0003723	GO:0003723
	GO:0003968		GO:0003968	GO:0003968
Q30U32	GO:0000156		GO:0004871	GO:0000155
				GO:0004871
O93828	GO:0004585			GO:0016597
	GO:0016597			GO:0016740
	GO:0016740			GO:0016743
	GO:0016743			
Q30SN9	GO:0003824			
	GO:0004252			
Q2YTY7	GO:0003723			
	GO:0009982			
O78911	GO:0008137	GO:0008137	GO:0008137	GO:0008137
	GO:0016491	GO:0016491	GO:0016491	GO:0016491

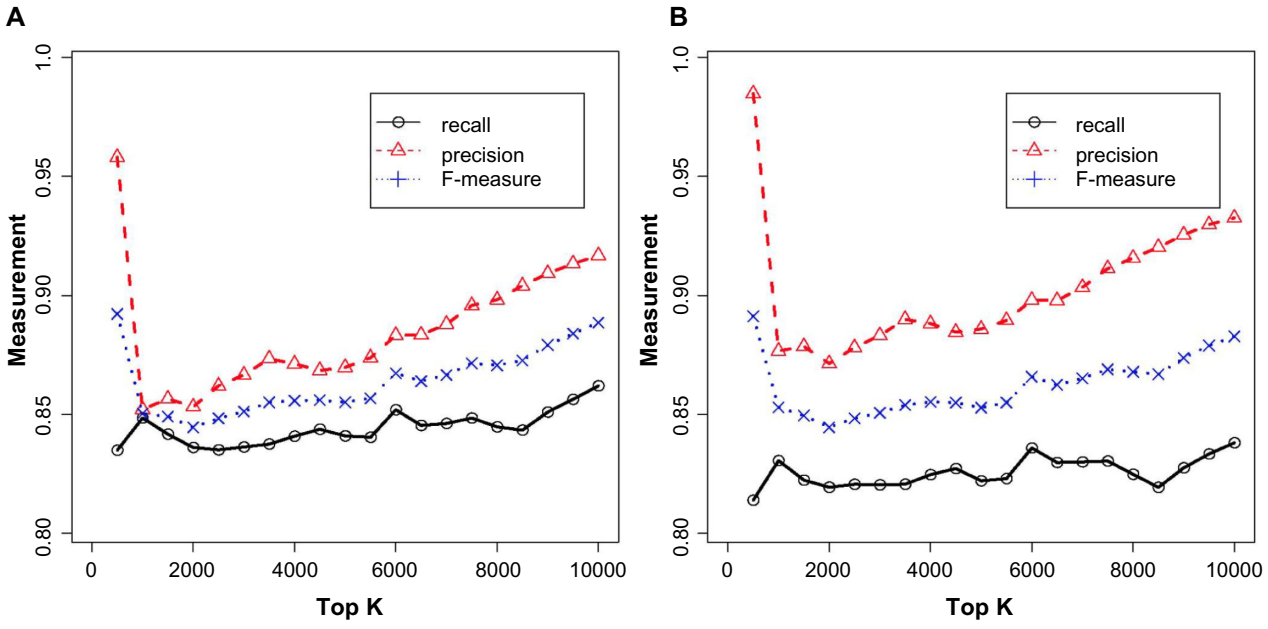

Figure 6. Performance in function prediction based on circular permutations for the top *k* highest degree proteins, with relationships defined based on circular permutations between pairs of multidomain proteins. (**A**) Using thresholds $z_1 \geq 3$, $z_2 \geq 0.5$. (**B**) using thresholds $z_1 \geq 3$, $z_2 \geq 1$.

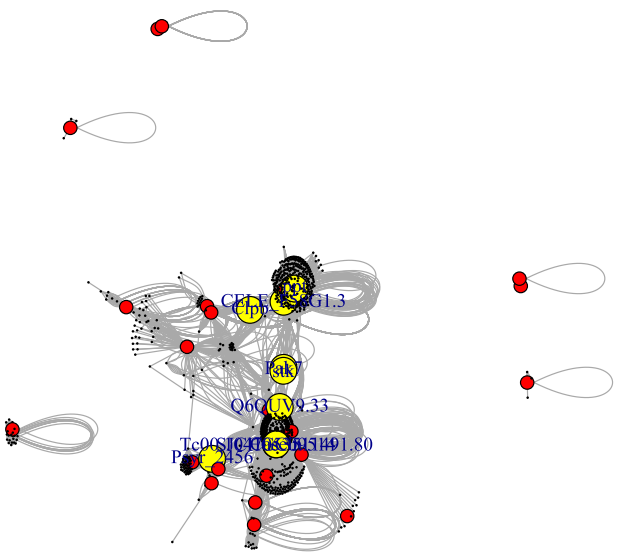

Figure 7. The subnetwork of colon cancer proteins. Red nodes denote known cancer proteins; yellow nodes are the proteins predicted to be associated with colon cancer.

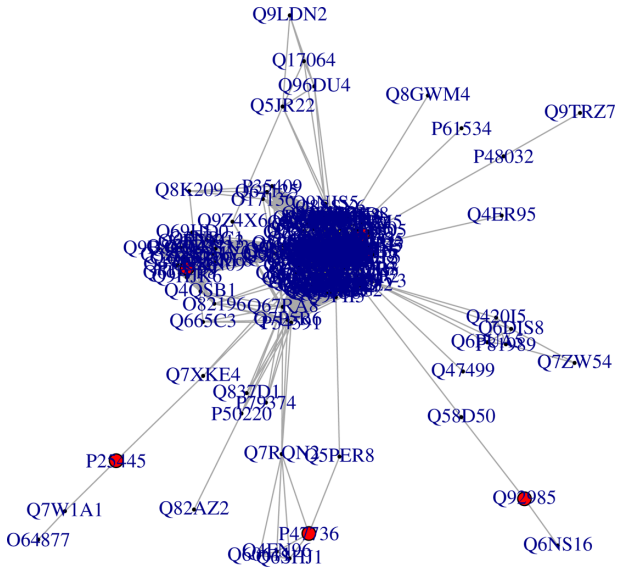

Figure 8. The subnetwork of skin cancer proteins, showing only the Top 200 nodes (ranked by betweenness centrality).

importance of each node in a given cancer subnetwork. As before, we can then choose a threshold on the z-scores, or on the Top K proteins in this ranking to determine the final list of proteins that are predicted to be associated with the specified type of cancer.

In this work, we consider the Top 10 multidomain proteins in the betweeness centrality ranking as predicted to be associated with the given cancer type, and then use literature search to further validate the predicted associations. See Figures 7 and 8, and figures in Supplementary Material. In the figures, red nodes denote known cancer proteins for the given cancer type, yellow nodes are the predicted proteins (those with largest betweenness centrality values in the subnetwork). Observe that the yellow nodes tend to have much more connections in the network, and they tend to be linking different regions in the network, showing their significance.

Table 6 lists the 10 proteins with the highest betweenness centrality measures in each of the five cancer subnetworks. For each cancer type, these proteins are the most important nodes, with respect to CP relationships, and usually have more connection with the cancer proteins.

Literature search. Given the cost and expense of wet-lab experimental verification, it is important to further narrow down the list of predicted proteins. For this purpose, we use literature search on PubMed to determine whether there has been previous publications on the predicted connection between the multidomain protein and the given cancer type. We also use information from the Atlas of Genetics and Cytogenetics in Oncology and Haematology (http://atlasgeneticsoncology.org). If a protein is already listed in the Atlas, we assume that it is known to be related with cancer. In some cases, the relationship may not be with the specific cancer type that our system predicted. Thus, for literature-based validation, we searched for the Top 10 predicted proteins for each cancer type, shown in Table 6. If the predicted association has not been previously reported in PubMed, or not in the Atlas, we take it to be a novel association found by our method.

From the table, we can observe that several of the Top 10 proteins for a given cancer type also appeared in the Top 10 for other cancer types (eg, PAK7 and stk-42 shared by bone and colon; HTK16, HTK98, pik3r1, and ced-2 shared by bone and lung). This is not completely unexpected, given that certain proteins are known to be implicated in multiple cancer types (see also the column labeled "In CT"). Lung cancer shared more circular proteins in the Top 10 with bone cancer. More significantly, some of the multidomain proteins in the Top 10 have previously been reported in PubMed or in the Atlas to be involved in some cancers, but not necessarily in the cancer type that was predicted by the proposed method. Of the Top 10 proteins predicted for each cancer type, we have

Table 5. Summary statistics of the circular relationship subnetworks for five cancer types (bone, colon, lung, skin, and breast).

	N_C	N_{CC}	N_{NC}	N_N	N_E
bone	130	28	3578	3708	66505
colon	168	43	3252	3420	55630
lung	131	38	3400	3531	53462
skin	254	117	13698	13952	905412
breast	441	143	11301	11742	713063

Abbreviations: N_C, Number of cancer proteins in subnetwork; N_{CC}, Number of cancer proteins that have circular relationship(s) with other cancer proteins; N_{NC}, Number of non-cancer proteins in subnetwork; N_N, Total number of nodes in subnetwork; N_E, Total number of edges.

Table 6. Top 10 proteins with the highest betweenness centrality values in each of the five cancer subnetworks (bone, colon, lung, skin, and breast cancer).

	ACCESSION			BC			IN	IN	IN	IN	
RANK	NO	SHORT NAME	PROTEIN NAME (UniProt)	(x10^6)	Z-SCORE	N_{CCP}	ATLAS	Pub1	Pub2	CT	U
Bone											
1	P53356	HTK16	Tyrosine-protein kinase	0.245	15.859	4	No	0	0	**L**	*
2	Q3TQJ7	PAK 7	Serine/threonine-protein kinase PAK 7	0.213	13.754	4	Yes	1	20	**C**	
3	Q7RYZ3	stk-42	Serine/threonine protein kinase-42	0.185	11.972	4	No	0	1	**C**	
4	Q5R8U2	DKFZp469F0413	Putative uncharacterized protein	0.167	10.781	5	No	0	0		**U**
5	O77440	HTK98	Tyrosine-protein kinase	0.159	10.211	20	No	0	0	**L**	
6	Q6GQ43	pik3r1	MGC80357 protein	0.159	10.211	18	Yes	0	76	**L**	
7	Q9N597	deleted		0.159	10.211	18	No	0	0	**L**	
8	Q9NHC3	ced-2	Cell death abnormality protein 2	0.159	10.211	18	No	0	1	**L**	
9	O62272	CELE_F58G1.3	Hypothetical protein	0.151	9.747	8	No	0	0		**U**
10	Q34QW6		Deleted (obsolete)	0.147	9.461	4	No	0	0		
Colon											
1	Q3TQJ7	PAK 7	Serine/threonine-protein kinase PAK 7	0.237	25.054	4	Yes	0	20	**B**	*
2	Q7RYZ3	stk-42	Serine/threonine protein kinase-42	0.149	15.727	4	No	0	1	**B**	
3	O62272		Serine/threonine-protein phosphatase	0.103	10.813	8	No	0	0		
4	O14428	ppt-1	Serine/threonine-protein phosphatase	0.094	9.882	8	No	0	2		
5	Q6QUV9			0.081	8.432	9	No	0	0		
6	Q4ZTM7		Short-chain dehydrogenase/reductase SDR	0.069	7.173	9	No	0	0		
7	Q3TXD4	Clpb	Putative uncharacterized protein	0.042	4.288	22	No	0	10		**U**
8	O77008		Casein kinase II alpha subunit	0.040	4.059	21	No	0	0		
9	Q4DHP2		Mitogen-activated protein kinase, putative	0.040	4.059	30	No	0	0		**U**
10	Q5DHJ0		SJCHGC09514 protein	0.040	4.059	30	No	0	0		
lung											
1	P42686	SRK1	Tyrosine-protein kinase isoform	0.343	21.250	4	No	0	1		
2	Q9IAX8	CYP2P1	Cytochrome P450 2P1	0.282	17.469	1	No	0	0		
3	P53356	HTK16	Tyrosine-protein kinase	0.192	11.835	4	No	0	0	**B**	
4	O77440	HTK98	Tyrosine-protein kinase	0.178	10.970	18	No	0	0	**B**	
5	Q6GQ43	pik3r1	MGC80357 protein	0.178	10.970	18	Yes	5	76	**B**	
6	Q9N597		Deleted (obsolete)	0.178	10.970	18	No	0	0	**B**	
7	Q9NHC3	ced-2	Cell death abnormality protein 2	0.178	10.970	18	No	0	1	**B**	
8	Q61125	Bdkrb1	B1 bradykinin receptor	0.138	8.487	1	Yes	1	2		
9	P35409		Probable glycoprotein hormone G-protein coupled receptor	0.135	8.258	1	No	0	0		**U**
10	O17136	srx-21	Protein SRX-21	0.135	8.258	1	No	0	0		
Breast											
1	P54591	yhcG	Uncharacterized ABC transporter ATP-binding protein	3.292	33.507	4	No	0	0		**U***
2	Q7SYD8	xpnpep2	Zgc:63528	1.958	19.849	4	No	0	1		
3	Q8DSW8	metS	Methionine--tRNA ligase	1.915	19.407	1	Yes	48	340		*
4	Q4TMZ8		Deleted (obsolete)	1.906	19.321	1	No	0	0		
5	Q3QD47		Deleted (obsolete)	1.839	18.628	1	No	0	0		
6	O74634	MSM1	Methionine--tRNA ligase, mitochondrial	1.802	18.252	1	No	0	0		*
7	Q9HMN5	srp54	Signal recognition particle 54 kDa protein	1.792	18.154	1	Yes	0	3		*
8	Q62ZT7	cysK	Cysteine synthase	1.629	16.484	1	No	0	0		
9	Q63KP6	ileS2	Isoleucine--tRNA ligase 2	1.594	16.125	2	No	0	0		

(*Continued*)

Table 6. (*Continued*)

RANK	ACCESSION NO	SHORT NAME	PROTEIN NAME (UniProt)	BC (×10⁶)	BC Z-SCORE	N_{CCP}	IN ATLAS	IN Pub1	IN Pub2	IN CT	IN U
10	Q72D59	DVU_1070	Branched chain amino acid ABC transporter	1.566	15.838	1	No	0	0		
Skin											
1	Q5SMW4	P0568D10.9	Putative uncharacterized protein P0568D10.9	1.872	16.366	49	No	0	0		**U**
2	Q5Y2C4	CDC2	Cdc2 protein kinase	1.803	15.756	49	Yes	68	3177		
3	Q6XKY3	hog1	Mitogen-activated protein kinase hog1	1.803	15.756	49	No	0	32		
4	Q80YP0	Cdk3	Cyclin-dependent kinase 3 Protein kinase domain containing protein,	1.803	15.756	49	Yes	0	29		
5	Q53PY9	Os11g0150700	expressed	1.796	15.697	49	No	0	0		
6	Q54QD5	nek1	Probable serine/threonine-protein kinase nek1	1.796	15.697	49	Yes	0	8		**U**
7	Q5AI03	SPS1	Likely protein kinase	1.796	15.697	49	Yes	0	7		**U***
8	O04099	Bcpk1	Putative serine/threonine protein kinase	1.787	15.619	49	No	0	0		**U**
9	P51956	NEK3	Serine/threonine-protein kinase Nek3	1.787	15.619	49	Yes	0	6		
10	P51957	NEK4	Serine/threonine-protein kinase Nek4	1.787	15.619	49	Yes	0	4		

Abbreviations: BC, Betweeness centrality; N_{CCP}, No. of connected cancer proteins (of the given cancer type); In Pub1, number of times published in PubMed (with the indicated cancer type); In Pub2, number of times published in PubMed (with any cancer type); In CT, also found for cancer type (B, bone; C, colon; L, lung; R, breast; S, skin); U, Described as "unknown", "uncharacterized", "putative", "hypothetical", or "probable" in Uniprot.
Note: "*" indicates that the protein was in the list of known cancer proteins from the Cancer Resource dataset.

the following known associations (from Cancer Resource) or reported associations (in PubMed) with the specific cancer type predicted: bone 2, colon 1, lung 2, breast 4, and skin 2. This means that most of the predicted proteins have not yet been connected with the specific cancer predicted. The table also shows the predicted proteins that are described in Uniprot as uncharacterized, unknown, putative, probable, or hypothetical (denoted as U). Those with U* are assumed to be known, since they have already been included in the Cancer Resource dataset.[63] In fact, a good number of the predicted proteins have not been previously reported to be connected with any cancer type in the literature: bone 3, colon 6, lung 3, breast 5, and skin 3 (not including the deleted (obsolete) proteins). These proteins, along with those denoted with a U in the table, represent a reduced set of multidomain proteins that are most likely associated with the specified cancer types, and can thus be subjected to further wet-lab verification.

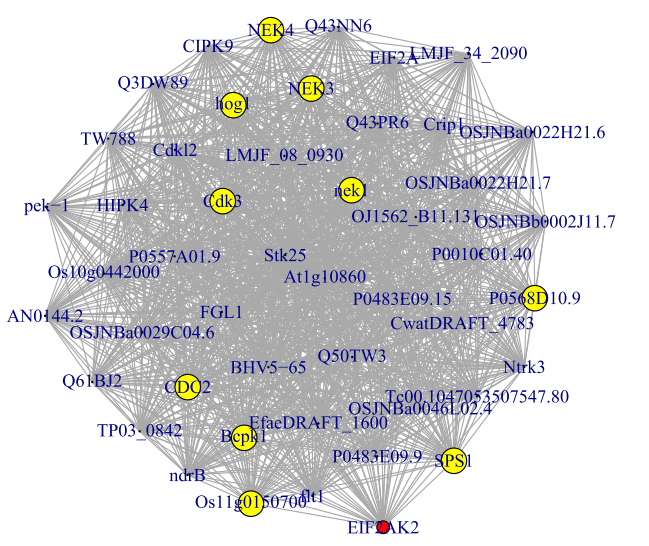

Figure 9. A 49-node dense subgraph at the center of the skin cancer subnetwork. See also Figure 8.

One striking observation from the table is the fact that all the Top 10 proteins for skin cancer have the same value of 49 for N_{CCP} – the number of connected cancer proteins. Their betweeness centrality values (and hence the z-scores) are also similar, except for the first one. This implies a potential clique or quasi-clique (dense subgraph) with about 49 nodes in the skin cancer subnetwork. In fact, upon closer investigation, we observed a large almost complete-connected component with 49 proteins. Figure 8 shows the subnetwork for skin cancer using only the Top 200 nodes (the full network is too large for display), while Figure 9 shows the subnetwork involving only the Top 49 nodes. The almost complete nature of the graph is clear, and the nodes all have about the same betweeness centrality value. These must be playing an important role in the skin cancer circular protein subnetwork. Of the 49 proteins in this dense subnetwork, two were in the original known protein set from Cancer Resource dataset, and more than half did not have GO function annotation, and many were characterized as unknown.

Discussion and Conclusion

We identify three major contributions of this work. First, we proposed an efficient algorithm for rapid identification of both exact and approximate CPs in multidomain proteins. By analyzing the computational complexity of the algorithms, we showed their superiority over current state of the art. We also presented results on the practical running time required by the

algorithms. Second, we showed how the circular relations can be used to construct a network between these proteins, based on which we perform functional annotation of multidomain proteins. This method showed a performance of about 0.81 precision and 0.88 recall, using known functions from GO on the Top 500 proteins. Third, we extended the method to construct subnetworks for selected cancer subtypes and performed prediction of the association between multidomain proteins and the selected cancer types. Our prediction based on the Top 10 proteins with the highest betweeness centrality measures contained many uncharacterized multidomain proteins that are likely to be associated with specific cancer types. Some of the multidomain proteins are predicted to be associated with more than one cancer type.

Of note is the observed 49-protein dense subgraph for the skin cancer subnetwork, which contains the top-ranking proteins predicted to be associated with skin cancer. We are not aware of any previous report of such a dense subgraph of multidomain proteins related by CPs, with known associations to skin cancer. Thus, we do not have a hard evidence on the practical relevance of the observed dense network to skin cancer. However, within this subgraph, we can identify groups of genes that are implicated in various functions that are relevant to cancer. For instance, we can observe several known cancer-related functional groups: Pro-oncogenes and growth-promoter genes[72] (Ntrk3, Stk25, flt1, Cdk3); genes involved in inducing angiogenesis[73] (EIF2AK2, Ntrk3, Stk25, flt1); genes for regulating oxidative stress[74] (Bcpk1, hog1); genes for regulating protein phosphorylation and dephosphorylation[75,76] (NEK3, Stk25, Cdkl2, Stk25, Cdk3, CIPK9, HIPK4, EIF2AK2); gene for regulating glucose level (SPS1); genes for regulating cell cycle, cell migration, and metastasis[76–79] (FGL1, flt1, nek1, CDC2, hog1, Cdk3, EIF2AK2); and genes for other functions. This long list of cancer-related genes in the 49-node dense subgraph gives us some confidence on the relevance of this network. We can also expect that some of the unknown/uncharacterized proteins in this 49-node dense network are likely to be implicated in some cancers, especially in skin cancer. It will be interesting to study this quasi-clique further, for any biological relevance to molecular studies of skin cancer in particular and of cancer in general.

We note that our prediction for associations with cancer is based primarily on information from Cancer Resource dataset.[63] Thus, a multidomain protein that is not in the list will be predicted as a potentially novel association with the given cancer type. This might explain why some of the predicted associations are already observed in the Atlas or in PubMed. Yet, this still gives some credence to the power of the proposed method: it can find important associations between cancer-related proteins. Verifying whether the association is novel or not can be performed easily.

Our approach is a computational method, which essentially generates hypotheses on potential functional associations for multidomain proteins. This provides an important mechanism needed to prune down the large number of possibilities for later biological verification of the predicted associations in the wet-laboratory.

Author Contributions

Conceived and designed the experiments: DA, JL, YJ. Analyzed the data: DA, JL, YJ, BHJ. Wrote the first draft of the manuscript: DA, JL. Contributed to the writing of the manuscript: DA, JL, YJ, BHJ. Agree with manuscript results and conclusions: DA, JL, YJ, BHJ. Jointly developed the structure and arguments for the paper: DA, JL, YJ, BHJ. Made critical revisions and approved final version: DA, JL, YJ, BHJ. All authors reviewed and approved of the final manuscript.

Supplementary Materials

Figure S1. The sub-network of bone cancer proteins.

Figure S2. The sub-network of lung cancer proteins.

Figure S3. The sub-network of breast cancer proteins, showing the top 200 nodes.

REFERENCES

1. Hedlund J, Johansson J, Persson B. BRICHOS – a superfamily of multidomain proteins with diverse functions. *BMC Res Notes*. 2009;2:180.
2. Willander H, Hermansson E, Johansson J, Presto J. BRICHOS domain associated with lung fibrosis, dementia and cancer a chaperone that prevents amyloid fibril formation? *FEBS J*. 2011;278:3893–904.
3. Fecker LF, Geilen CC, Tchernev G, et al. Loss of proapoptotic Bcl-2-related multidomain proteins in primary melanomas is associated with poor prognosis. *J Invest Dermatol*. 2006;126:1366–71.
4. Lessene G, Czabotar PE, Colman PM. BCL-2 family antagonists for cancer therapy. *Nat Rev Drug Discov*. 2008;7:989–1000.
5. Theodorakis P, Lomonosova E, Chinnadurai G. Critical requirement of BAX for manifestation of apoptosis induced by multiple stimuli in human epithelial cancer cells. *Cancer Res*. 2002;62:3373–6.
6. Vogel HJ, Chan DI. Circular proteins: ring around with NOESY. *Structure*. 2005;13:688–90.
7. Craik DJ. Circling the enemy: cyclic proteins in plant defence. *Trends Plant Sci*. 2009;14:328–35.
8. Craik DJ. Seamless proteins tie up their loose ends. *Science*. 2006;311:1563–4.
9. Barbeta BL, Marshall AT, Gillon AD, Craik DJ, Anderson MA. Plant cyclotides disrupt epithelial cells in the midgut of lepidopteran larvae. *Proc Natl Acad Sci USA*. 2008;105:1221–5.
10. Belkum MJ, Martin-Visscher LA, Vederas JC. Structure and genetics of circular bacteriocins. *Trends Microbiol*. 2011;19:411–8.
11. Montalbn-Lpez M, Snchez-Hidalgo M, Cebrin R, Maqueda M. Discovering the bacterial circular proteins: bacteriocins, cyanobactins, and pilins. *J Bioll Chem*. 2012;287:27007–13.
12. Cotter PD, Hill C, Ross RP. Bacterial lantibiotics: strategies to improve therapeutic potential. *Curr Protein Pept Sci*. 2005;6:61–75.
13. Kohli R, Walsh C. Enzymology of acyl chain macrocyclization in natural product biosynthesis. *Chem Commun*. 2003;3:297–307.
14. Tang YQ, Yuan J, Osapay G, et al. A cyclic antimicrobial peptide produced in primate leukocytes by the ligation of two truncated alpha-defensins. *Science*. 1999;286:498–502.
15. Hemperly JJ, Cunningham BA. Circular permutation of amino acid sequences among legume lectins. *Trends Biochem Sci*. 1983;8:100–2.
16. Cunningham BA, Hemperly JJ, Hopp TP, Edelman GM. Favin versus concanavalin A: circularly permuted amino acid sequences. *Proc Natl Acad Sci U S A*. 1979;76:3218–22.
17. Lindqvist Y, Schneider G. Circular permutations of natural protein sequences: structural evidence. *Curr Opin Struct Biol*. 1997;7(3):422–7.
18. Jeltsch A. Circular permutations in the molecular evolution of DNA methyltransferases. *J Mol Evol*. 1999;49(1):161–4.
19. Bujnicki J. Sequence permutations in the molecular evolution of DNA methyltransferases. *BMC Evol Biol*. 2002;2:3.
20. Heringa J, Taylor WR. Three-dimensional domain duplication, swapping and stealing. *Curr Opin Struct Biol*. 1997;7:416–21.

21. Doolittle RF, Bork P. Evolutionarily mobile modules in proteins. *Sci Am*. 1993;269:50–6.
22. Weiner J, Bornberg-Bauer E. Evolution of circular permutations in multidomain proteins. *Mol Biol Evol*. 2006;23:734–43.
23. Weiner J, Thomas G, Bornberg-Bauer E. Rapid motif-based prediction of circular permutations in multi-domain proteins. *Bioinformatics*. 2005;21:932–7.
24. Han JH, Batey S, Nickson AA, Teichmann SA, Clarke J. The folding and evolution of multidomain proteins. *Nat Rev Mol Cell Biol*. 2007;8:319–30.
25. Apic G, Gough J, Teichmann SA. Domain combinations in archaeal, eubacterial and eukaryotic proteomes. *J Mol Biol*. 2001;310:311–25.
26. Ekman D, Bjorklund AK, Frey-Skott J, Elofsson A. Multi-domain proteins in the three kingdoms of life: orphan domains and other unassigned regions. *J Mol Biol*. 2005;348:231–43.
27. Grubber CW, Elliott AG, Daly NL, Craik D. Distribution and evolution of circular miniproteins in flowering plants. *Plant Cell*. 2008;20:2471–83.
28. Corpet F, Servant F, Gouzy J, Kahn D. ProDom and ProDom-CG: tools for protein domain analysis and whole genome comparisons. *Nucleic Acids Res*. 2000;28:267–9.
29. Daly NL, Craik DJ. Acyclic permutants of naturally occurring cyclic proteins. *J Biol Chem*. 2000;275(25):3218–22.
30. Essen LO, Perisic O, Cheung R, Katan M, Williams RL. Crystal structure of a mammalian phosphoinositide-specific phospholipase C*δ*. *Nature*. 1996;380:595–602.
31. Proikas-Cezanne T, Waddell S, Gaugel A, Frickey T, Lupas A, Nordheim A. WIPI-1a (WIPI49), a member of the novel 7-bladed WIPI protein family, is aberrantly expressed in human cancer and is linked to starvation-induced autophagy. *Oncogene*. 2004;23:9314–25.
32. Patra CR, Rupasinghe CN, Dutta SK, et al. Chemically modified peptides targeting the PDZ Domain of GIPC as a therapeutic approach for cancer. *ACS Chem Biol*. 2012;7:770–9.
33. Hultqvist G, Punekar AS, Morrone A, et al. Tolerance of protein folding to a circular permutation in a PDZ domain. *PLoS One*. 2012;7:e50055.
34. Ivarsson Y, Travaglini-Allocatelli C, Brunori M, Gianni S. Folding and misfolding in a naturally occurring circularly permuted PDZ domain. *J Biol Chem*. 2008;283:8954–60.
35. Cole AM, Hong T, Boo LM, et al. Retrocyclin: a primate peptide that protects cells from infection by T- and M-tropic strains of HIV-1. *Proc Natl Acad Sci U S A*. 2002;99:1813–8.
36. Gusfield D. *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*. Cambridge, UK: Cambridge University Press; 1997.
37. Smyth WF. *Computing Patterns in Strings*. US: Addison-Wesley; 2003.
38. Adjeroh D, Bell T, Mukherjee A. *The Burrows-Wheeler Transform: Data Compression, Suffix Arrays and Pattern Matching*. US: Springer-Verlag; 2008.
39. Tanimoto SL. Method for detecting structure in polygons. *Pattern Recognit*. 1981;13:389–94.
40. Sebastian TB, Klein PN, Kimia BB. On aligning curves. *IEEE Trans Pattern Anal Mach Intell*. 2003;25:116–25.
41. Jung J, Lee B. Circularly permuted proteins in the protein structure database. *Protein Sci*. 2001;10:1881–6.
42. Garcia-Vallve S, Rojas A, Palau J, Romeu A. Circular permutants in beta-glucosidases (family 3) within a predicted double-domain topology that includes a (beta/alpha)8-barrel. *Proteins*. 1998;31:214–23.
43. Booth KS. Lexicographically least circular substrings. *Inf Process Lett*. 1980;10:240–2.
44. Iliopoulos CS, Sohel RM. Indexing circular patterns. *WALCOM*. 2008;4921:46–57.
45. Maes M. On a cyclic string-to-string correction problem. *Inf Process Lett*. 1990;35:73–8.
46. Mollineda RA, Vidal E, Casacuberta FA. Windowed weighted approach for approximate cyclic string matching. In: ICPR '02 Proceedings of the 16th International Conference on Pattern Recognition (ICPR'02) Vol 4 (Washington, DC, USA):40188IEEE Computer Society 2002.
47. Mollineda RA, Vidal E, Casacuberta F. Cyclic sequence alignments: approximate versus optimal techniques. *Intern J Pattern Recognit Artif Intell*. 2002;16:291–9.
48. Lin J, Adjeroh DA. All-against-all circular pattern matching. *Comput J*. 2012;55:897–906.
49. Barton C, Iliopoulos CS, Pissis SP. Fast algorithms for approximate circular string matching. *Algorithms Mol Biol*. 2014;9:9.
50. Heinemann U, Hahn M. Circular permutation of polypeptide chains: implications for protein folding and stability. *Prog Biophys Mol Biol*. 1995;64:121–43.
51. Maizel JV Jr, Lenk RP. Enhanced graphic matrix analysis of nucleic acid and protein sequences. *Proc Natl Acad Sci USA*. 1981;78:7665–9.
52. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol*. 1990;215:403–10.
53. Uliel S, Fliess A, Amir A, Unger R. A simple algorithm for detecting circular permutations in proteins. *Bioinformatics*. 1999;15:930–6.
54. Uliel S, Fliess A, Unger R. Naturally occurring circular permutations in proteins. *Protein Eng*. 2001;14:533–42.
55. Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*. 1970;48:443–53.
56. Gregor J, Thomason MG. Dynamic programming alignment of sequences representing cyclic patterns. *IEEE Trans Pattern Anal Mach Intell*. 1993;15:129–35.
57. Dundas J, Binkowski TA, DasGupta B, Liang J. Topology independent protein structural alignment. *BMC Bioinformatics*. 2007;8:388.
58. Chen L, Wu L, Wang Y, Zhang S, Zhang X. Revealing divergent evolution, identifying circular permutations and detecting active-sites by protein structure comparison. *BMC Struct Biol*. 2006;6:18.
59. Binkowski TA, DasGupta B, Liang J. Order independent structural alignment of circularly permuted proteins. *Conf Proc IEEE Eng Med Biol Soc*. 2004;4:2781–4.
60. Vesterstrom J, Taylor WR. Flexible secondary structure based protein structure comparison applied to the detection of circular permutation. *J Comput Biol*. 2006;13(1):43–63.
61. Lo W, Lyu P. CPSARST: an efficient circular permutation search tool applied to the detection of novel protein structural relationships. *Genome Biol*. 2008;9(1):R11.
62. Pagel P, Oesterheld M, Stümpflen V, Frishman D. The DIMA web resource -exploring the protein domain network. *Bioinformatics*. 2006;22:997–8.
63. Ahmed J, Meinel T, Dunkel M, et al. CancerResource: a comprehensive database of cancer-relevant proteins and compound interactions supported by experimental knowledge. *Nucleic Acids Res*. 2011;39:D960–7.
64. Hunt JW, Szymanski TG. A fast algorithm for computing longest common subsequences. *Commun ACM*. 1977;20:350–3.
65. Cormen TH, Leiserson CE, Rivest RL. *Introduction to Algorithms*. Cambridge MA: MIT Press; 1990.
66. Jokinen P, Ukkonen E. Two algorithms for approximate string matching in static texts (extended abstract). In: Tarlecki A, ed. Mathematical Foundations of Computer Science 1991. Proceedings of the 16th International Symposium. 1991. Berlin, HD: Springer: 240–8
67. Kärkkäinen J, Sanders P, Burkhardt S. Linear work suffix array construction. *J ACM*. 2006;53:918–36.
68. Ko P, Aluru S. Space efficient linear time construction of suffix arrays. *J Discrete Algorithms*. 2005;3:143–56.
69. Adjeroh D, Nan F. Suffix-Sorting *via* Shannon-Fano-Elias Codes. *Algorithms*. 2010;3:145–67.
70. Freeman LC. A set of measures of centrality based on betweenness. *Sociometry*. 1977;40:35–41.
71. Newman M. *Networks: An Introduction*. USA: Oxford University Press; 2010.
72. Davies H, Hunter C, Smith R, et al. Somatic mutations of the protein kinase gene family in human lung cancer. *Cancer Res*. 2005;65:7591–5.
73. Georg B, Clauss M, Risau W. Coordinate expression of vascular endothelial growth factor receptor-1 (fit-1) and its ligand suggests a paracrine regulation of murine vascular development. *Dev Dyn*. 1995;204:228–39.
74. Bilsland E, Molin C, Swaminathan S, Ramne A, Sunnerhagen P. Rck1 and Rck2 MAPKAP kinases and the HOG pathway are required for oxidative stress resistance. *Mol Microbiol*. 2004;53:1743–56.
75. Jin W, Yun C, Hobbie A, Martin MJ, Sorensen PH, Kim SJ. Cellular transformation and activation of the phosphoinositide-3-kinase-Akt cascade by the ETV6-NTRK3 chimeric tyrosine kinase requires c-Src. *Cancer Res*. 2007;67:3192–200.
76. Malumbres M, Barbacid M. Cell cycle, CDKs and cancer: a changing paradigm. *Nat Rev Cancer*. 2009;9:153–67.
77. Zheng D, Cho YY, Lau AT, et al. Cyclin-dependent kinase 3-mediated activating transcription factor 1 phosphorylation enhances cell transformation. *Cancer Res*. 2008;68:7650–60.
78. Heuvel S, Harlow E. Distinct roles for cyclin-dependent kinases in cell cycle control. *Science*. 1993;262:2050–4.
79. Fry AM, O'Regan L, Sabir SR, Bayliss R. Cell cycle regulation by the NEK family of protein kinases. *J Cell Sci*. 2012;125:4423–33.