IDENTIFICAÇÃO ESTRUTURAL EM REDES DE PROTEÍNAS

Tópicos Especiais em Redes Complexas II

Professor: Daniel Ratton Figueiredo

Aluno: Vitor Borges Coutinho da Silva

Artigos

- Comparative Analysis of Protein Networks: Hard Problems, Practical Solutions – Nir Atias e Roded Sharan

- Global alignment of multiple protein interaction networks with application to functional orthology detection – Rohit Singh, Jinbo Xu e Bonnie Berger

Motivação

- Decifrar o funcionamento das células
  - Proteínas interagem para formar o maquinário das células e transmitir sinais
  - Modificação de uma proteína tem efeitos drásticos em células
    - Doenças

- Técnicas automatizadas para medir interações entre proteínas - PPIs
  - Dados de baixa qualidade ➔ critério de conservação entre espécies
    - focar nas partes mais confiáveis da informação
    - inferir funções biológicas similares entre espécies ➔ transferência de conhecimento

Objetivo

- Critério de conservação entre espécies exige comparação das redes das espécies

- Descrever métodos usados para análise comparativa de redes

Redes PPI

- Grafo G=(V,E)
  - V ➔ conjunto de proteínas
  - Existência de uma aresta ➔ interação entre proteínas existe

Comparação de redes

- Dados: 2 ou mais redes e informações das sequências das proteínas

- Objetivo: identificar similaridades entre as redes
  - Local
  - Global
  - Assumindo que existem semelhanças entre redes
    - Ancestral comum

Similaridade de vértice

- Pontuação de similaridade de duas sequências ➔ p
  - Chance de observar essa similaridade de sequência aleatoriamente
  - Valor mais alto ➔ distância evolucionária menor
    - Maior chance de ter função similar

Similaridade de interação

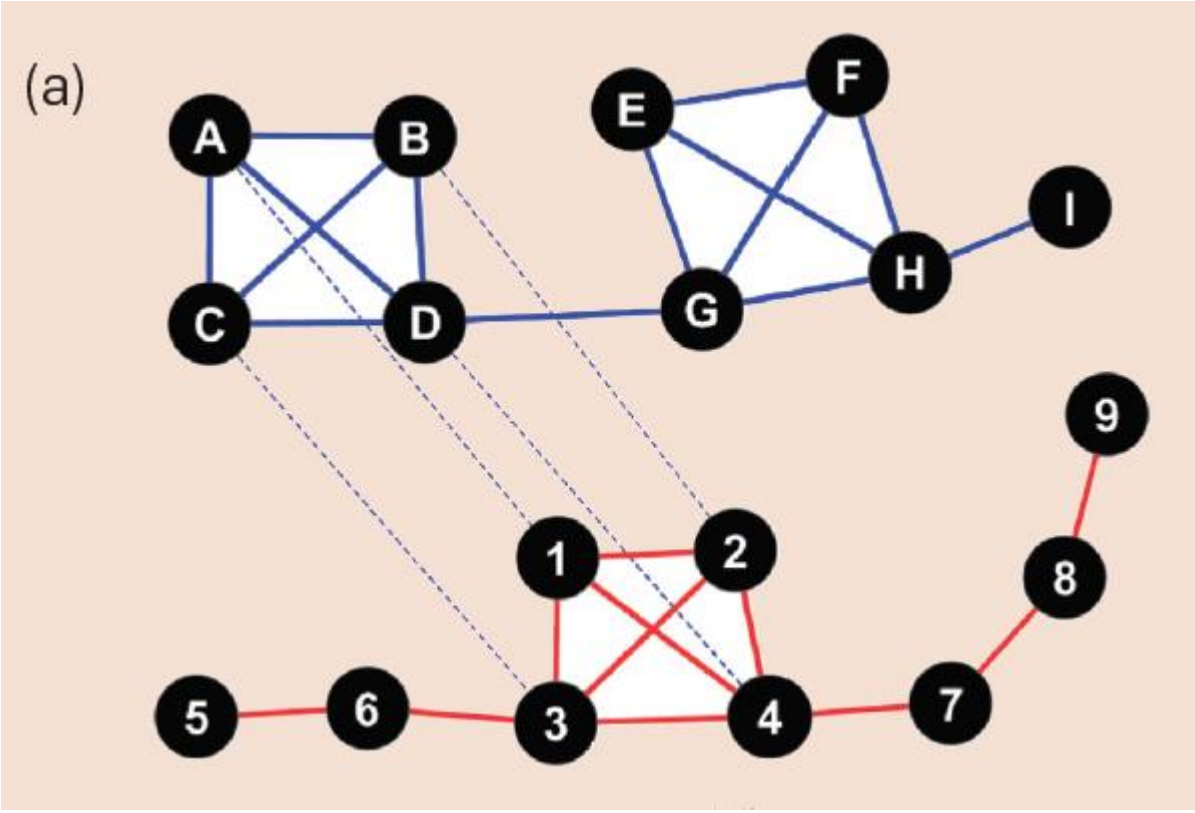
- Contar número de interações conservadas entre redes
  - Interação (x,y) conservada se em outra espécie a interação (x’,y’) existe

Tipos de busca por similaridade

- Alinhamento local de rede
  - Definida uma função para medir a similaridade de subredes de espécies distintas
  - Função definida para favorecer certos tipos de estrutura para guiar o algoritmo de busca
  - Alinhamentos podem ser inconsistentes
    - Uma proteína pode ser mapeada de forma diferente em outro mapeamento

Tipos de busca por similaridade

- Alinhamento local de rede



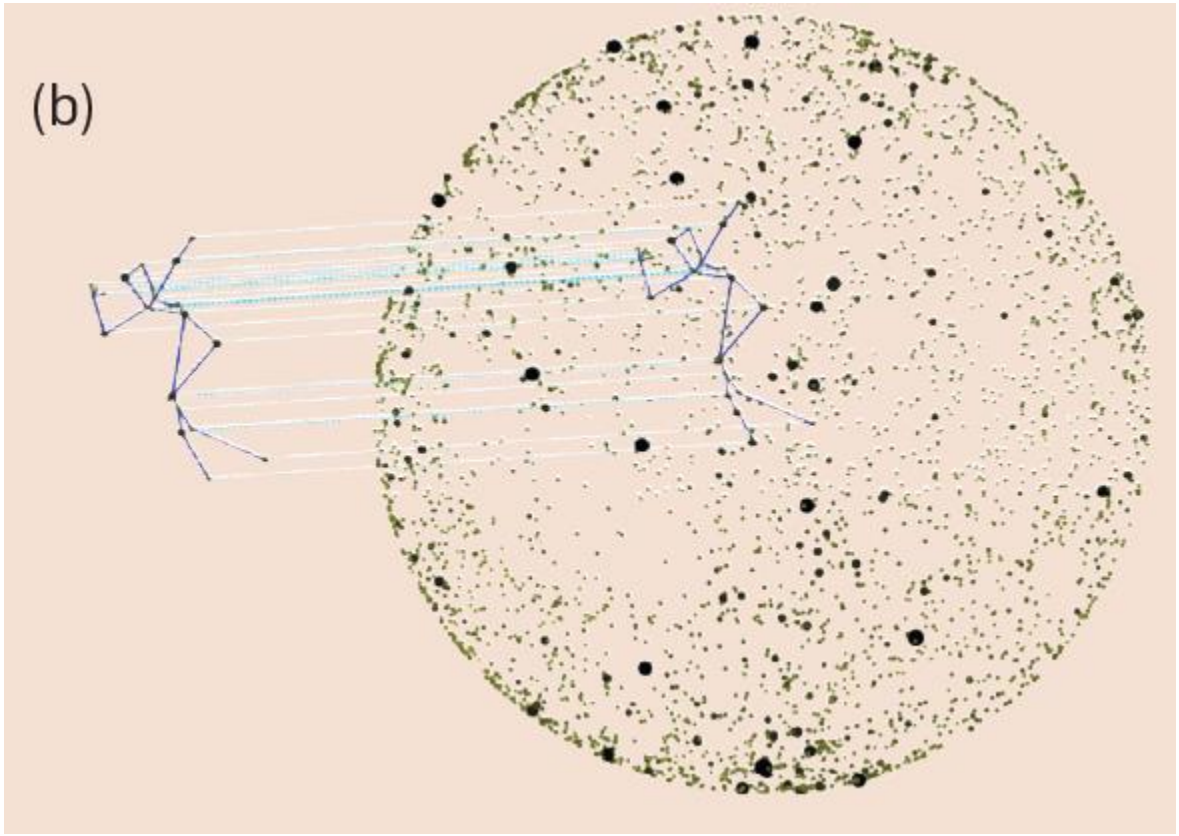
Tipos de busca por similaridade

- Network querying
  - Padrão procurado é conhecido de uma espécie bem estudada
  - Rede a ser buscada é de uma espécie pouco conhecida

  - Mapeamento pode ser exato (isomorfismo) ou não

Tipos de busca por similaridade

- Network querying

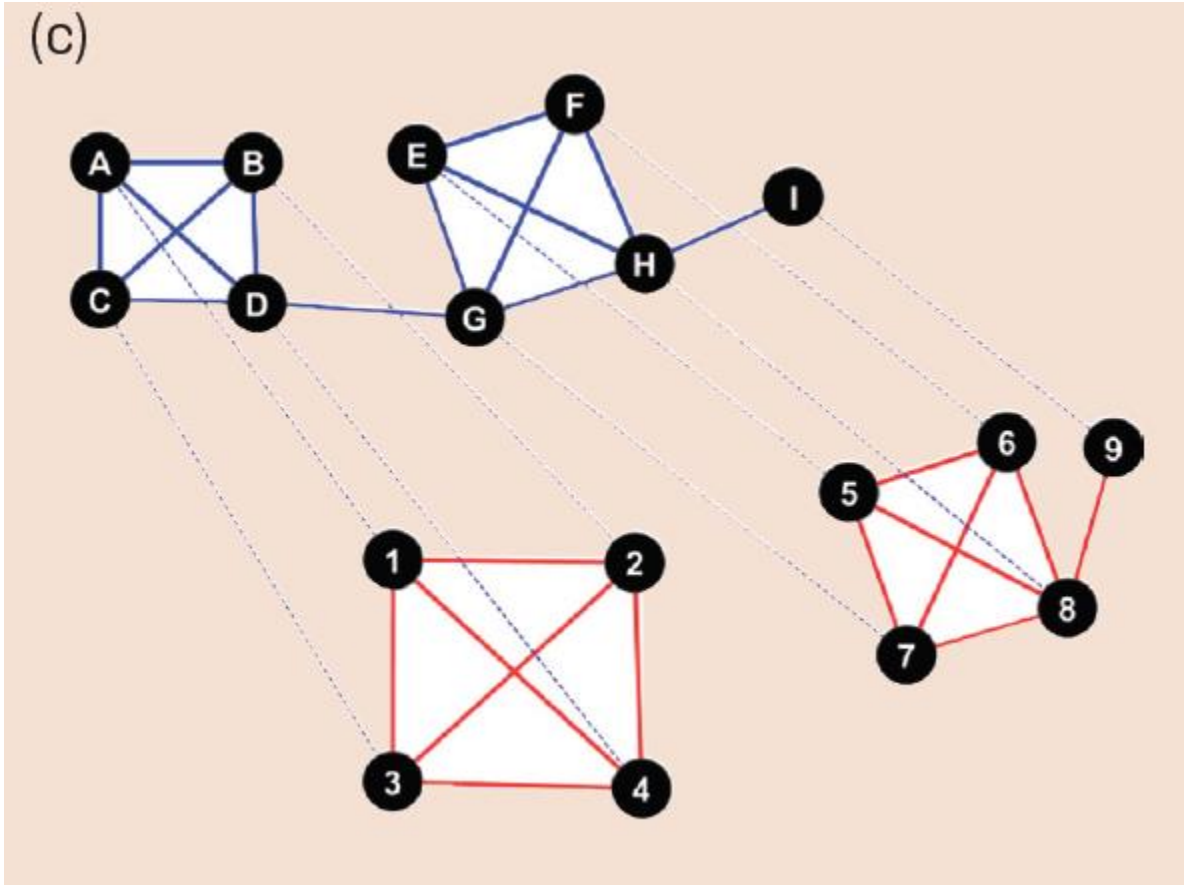


Tipos de busca por similaridade

- Alinhamento global de rede ➔ maximizando similaridade
  - Forma mais simples realiza mapeamento 1 – 1 entre vértices de espécies diferentes
  - Forma mais complexa realiza mapeamento muitos – muitos

Tipos de busca por similaridade

- Alinhamento global de rede ➔ maximizando similaridade



Métodos

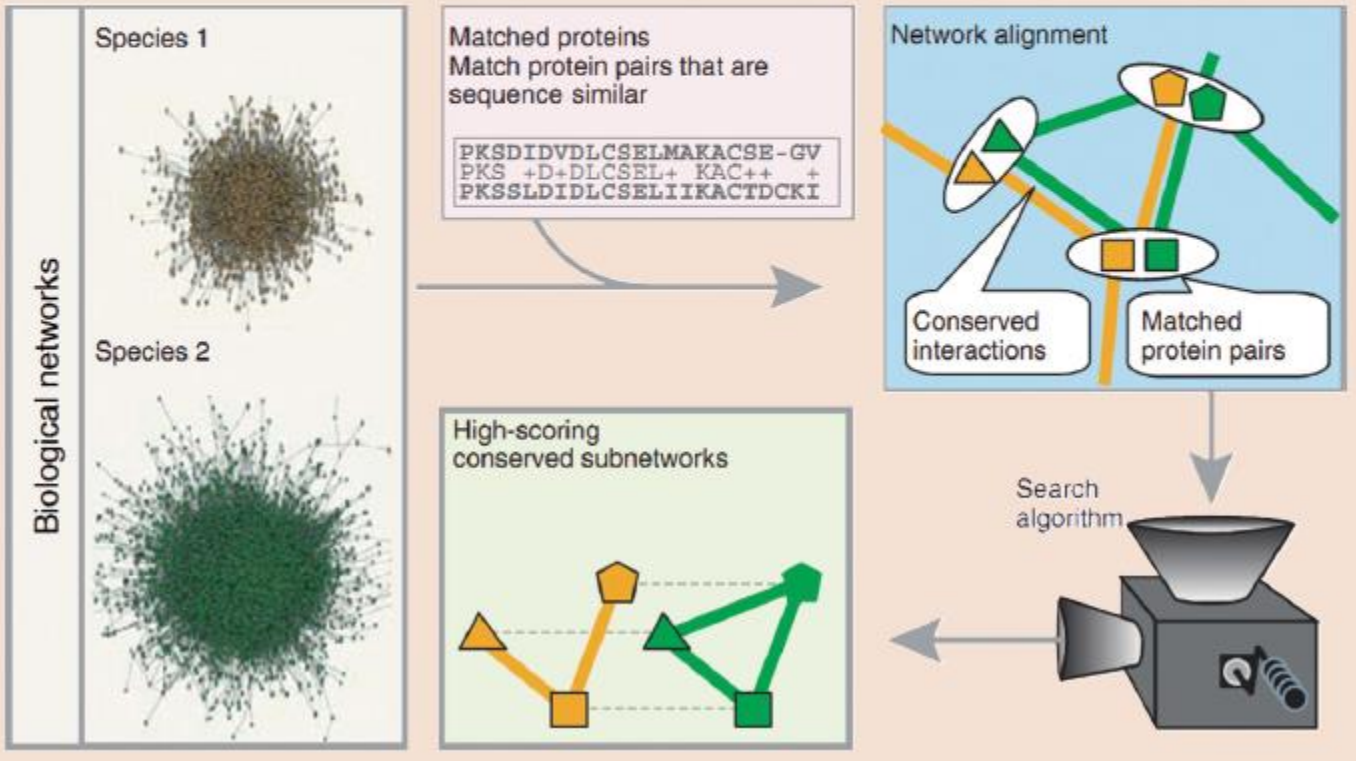
- Heurísticos (- Garantias)
  - NetworkBLAST
  - IsoRank
- Exatos (- Velocidade)
  - QPath
  - Torque

NetworkBLAST

- Método para alinhamento local de redes
  - Pareamento de subredes entre duas ou mais redes
- Método cria um grafo de alinhamento de rede
  - Nó ➔ conjunto de proteínas sequencialmente similares (um de cada espécie)
  - Aresta ➔ interações conservadas
    - Definida para permitir interação indireta através de vizinho em comum
- Subrede no grafo de alinhamento ➔ potencial pareamento das subredes das espécies

NetworkBLAST

Figure 2. The NetworkBLAST local network alignment algorithm. Given two input networks, a network alignment graph is constructed. Nodes in this graph correspond to pairs of sequence-similar proteins, one from each species, and edges correspond to conserved interactions. A search algorithm identifies highly similar subnetworks that follow a prespecified interaction pattern. Adapted from Sharan and Ideker.[30]



NetworkBLAST

- Pontuação de subredes
  - Função da densidade das subredes das espécies vs a chance de essas subredes surgirem aleatoriamente
- Identificação de subredes com maiores pontuações
  - Inicia com semente de 4 vértices
  - Busca local maximizando pontuação
  - Subrede com número máximo de nós
  - Algoritmo guloso alcança resultado próximo ao ótimo em 75% dos casos
- Vantagem ➔ Muito mais rápido que ILP
- Desvantagem ➔ Grafo de alinhamento cresce exponencialmente com K (número de espécies comparadas)

IsoRank

- Método para alinhamento global de redes
  - Duas ou mais redes
- Transitivo
  - Se A1➔A2 e A2➔A3; Então A1➔A3
- Entrada: Duas ou mais redes PPI e similaridade entre nós
- Objetivo: mapeamento entre redes de entrada que maximiza
  - 1 - o tamanho do grafo comum as espécies
  - 2 - a similaridade de sequência agregada dos nós mapeados

- Constrói problema de autovalor cuja solução retorna um mapeamento entre os nós

IsoRank

- Algoritmo em duas etapas:
  - Associa pontuação de similaridade funcional para cada possível casamento entre nós da rede
  - Constrói mapeamento extraindo casamentos mutuamente consistentes de alta pontuação

IsoRank

- Calcular a pontuação de similaridade funcional
  - Nós i e j são bom casamento se suas sequências são similares e se seus vizinhos N(i) e N(j) formam bons casamentos (Recursão)
  - Grafos sem peso

$$R = \Sigma R_{ij} = \sum_{u \in N(i)} \sum_{v \in N(j)} \frac{1}{|N(u)||N(v)|} R_{uv} \quad i \in V_1, j \in V_2, \qquad [1]$$

  - Grafos pesados

$$R_{ij} = \sum_{u \in N(i)} \sum_{v \in N(j)} \frac{w(i,u)w(j,v)}{\sum_{r \in N(u)} w(r,u) \sum_{q \in N(v)} w(q,v)} R_{uv}$$

$$i \in V_1, j \in V_2, \qquad [2]$$

  - Matricial $R = AR,$ where

$$A[i,j][u,v] = \begin{cases} \dfrac{1}{|N(u)||N(v)|} & \text{if } (i,u) \in E_1,\ (j,v) \in E_2. \\ 0 & \text{otherwise} \end{cases}$$

IsoRank

- Cálculo de R ➔ método da potência

$$R(k+1) \leftarrow AR(k)/|AR(k)|$$

- Incluindo similaridade de sequência B ➔ E = B/|B|

- Combinação convexa das duas similaridades

$$R = \alpha AR + (1-\alpha)E, \qquad 0 \leq \alpha \leq 1, \text{ or}$$
$$R = (\alpha A + (1-\alpha)E1^{T})R.$$

- Convergência

$$O(\log(1/1\text{-}\alpha))$$

IsoRank

- Extrair mapeamentos de R
  - Identificar pares de nós que tem pontuação alta garantindo transitividade
  - Um-para-Um
  - Muitos-para-Muitos

- Mapeamento Muitos-para-Muitos
  - Conjunto de nós de todas as redes é particionado
  - Cada partição corresponde a um conjunto de nós mapeados entre si
    - 0, 1 ou mais nós de cada espécie
    - Desempenham a mesma função nas diferentes espécies

IsoRank

- Algoritmo de Mapeamento
  - Restrições
    - Cada nó tem alta pontuação R com a maioria dos nós do conjunto
    - Não existem nós com essa propriedade fora do conjunto
    - Existe um número máximo de nós permitidos no conjunto por espécie
  - Crie um grafo K-partido a partir de R

IsoRank

- Algoritmo de Mapeamento

While the k-partite graph H has any edges remaining:

1. Select the edge (i, j) with the highest score (let i be from G_1 and j from G_2). Initialize a new match-set with i and j as its initial members.

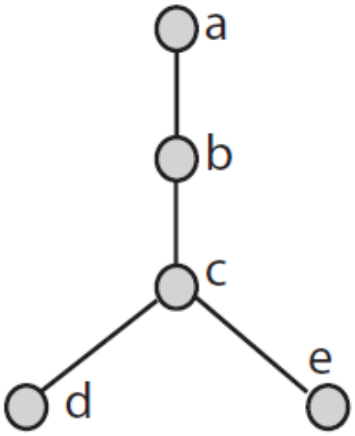
2. In every other species $(G_3, \ldots, G_k)$, if a node l exists such that (i) R_{il} and R_{jl} are the highest scores between l and any node in G_1 and G_2, respectively and, (ii) the scores $R_{il} \geq \beta_1 R_{ij}$, and $R_{jl} \geq \beta_1 R_{ij}$, add it to the set. This set of nodes forms the primary match-set; it has at most one node from each species.

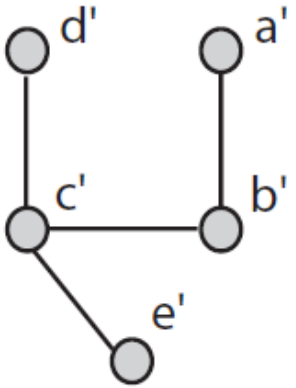
3. Add upto r-1 nodes from different parts of the graph to the primary match-set. Suppose u (from G_x) is in the primary match-set. Then, a node v (from G_x) is added to the set if $R_{vw} \geq \beta_2 R_{uw}$ for each node w (w≠u) in the primary set.

4. Remove from H all of the nodes in this match-set and their edges.

IsoRank

- Exemplo





R

	a'	b'	c'	d'	e'
a	0.0312		0.0937		
b		0.1250		0.0625	0.0625
c	0.0937		0.2812		
d		0.0625		0.0312	0.0312
e		0.0625		0.0312	0.0312

$$R_{aa'} = \frac{1}{4} R_{bb'}$$

$$R_{bb'} = \frac{1}{3} R_{ac'} + \frac{1}{3} R_{a'c} + R_{aa'} + \frac{1}{9} R_{cc'}$$

$$R_{dd'} = \frac{1}{9} R_{cc'}$$

$$R_{cc'} = \frac{1}{4} R_{bb'} + \frac{1}{2} R_{be'} + \frac{1}{2} R_{bd'} + \frac{1}{2} R_{eb'} + \frac{1}{2} R_{db'} + R_{ee'} + R_{ed'} + R_{de'} + R_{dd'}$$

Métodos

- Heurísticos (- Garantias)
  - NetworkBLAST
  - IsoRank
- Exatos (- Velocidade)
  - QPath
  - Torque

QPath

- Utilizado no caso de realizar network querying para subredes pequenas (5 – 15 vértices)

- Tipo de color coding
  - Pintar aleatoriamente os vértices da rede usando K cores distintas
    - Uma cor por vértice
  - Tarefa de encontrar um subgrafo simples se traduz em encontrar um subgrafo colorido
    - Colorido ➔ subgrafo onde K vértices apresentam as K cores distintas
  - Para encontrar subgrafo com alta probabilidade processo deve ser repetido várias vezes
    - Repinta e procura novamente
  - Probabilidade de grafo de tamanho K ser colorido: $k!/k^k > e^{-k}$
  - Número de iterações: e^k

QPath

- Estende o color coding para buscar redes pesadas suportando pareamentos inexatos
- Otimiza função de pontuação
  - Similaridade de sequência
  - Confiança na interação ➔ cada aresta no caminho pareado contribui com seu peso
  - Penalidade por inserção
  - Penalidade por remoção

QPath

- Método de programação dinâmica para encontrar casamento com maior pontuação
- W ➔ pontuação do casamento ótimo dos i primeiros nós da consulta terminando no vértice v da rede

$$W(i, v, S, \theta_{\text{del}}) = \max_{u \in N(v)} \begin{cases} W(i-1, u, S \backslash \{c(v)\}, \theta_{\text{del}}) + w(u, v) + \sigma(q_i, v) & (u, v) \in E \\ W(i, u, S \backslash \{c(v)\}, \theta_{\text{del}}) + w(u, v) - c_{\text{ins}} & (u, v) \in E \\ W(i-1, v, S, \theta_{\text{del}} - 1) - c_{\text{del}} & 0 < \theta_{\text{del}} \leq N_{\text{del}} \end{cases}$$

The best scoring path is obtained using a standard dynamic programming back-tracking starting at

$$\underset{v \in V,\, S \subseteq \{1, \ldots, k+N_{\text{ins}}\},\, \theta_{\text{del}} \leq N_{\text{del}}}{\operatorname{argmax}} W(k, v, S, \theta_{\text{del}}).$$

Torque

- Método que utiliza programação linear inteira
- Utilizado no caso de realizar network querying sem conhecer topologia
  - Conhece os atores buscados mas não como eles interagem
- Objetivo ➔ encontrar subgrafo mapeado conexo

Torque

- O método resolve o problema modelando o subgrafo solução como uma rede de fluxos

- A rede de fluxos contém um ralo que escoa k-1 fluxos e k-1 fontes produzindo uma unidade de fluxo
- O fluxo total do sistema é preservado

Torque

- Variáveis
  - c_v é binária e indica se o vértice v participa da solução
  - e_{uv} é binária e indica se a aresta faz parte da solução
  - f_{uv} e f_{vu} indicam a magnitude e direção do fluxo percorrendo uma aresta
  - r_v é binária e marca o vértice ralo
  - g_{vq} é binária e para cada par de vértices sequencialmente similares da rede e da consulta (v,q) indica se v e q estão casados

Torque

- Requisitos
  - A solução deve abranger k nós
  - Só um nó pode servir como ralo
  - Uma aresta é parte da solução sse seus dois vértices são

$$\sum_{v \in V} c_v = k$$

$$\sum_{v \in V} r_v = 1$$

$$e_{vu} \leq \frac{1}{2} c_v + \frac{1}{2} c_u \quad \forall (v, u) \in E$$

Torque

- Restrições (Q ➔ conjunto de proteínas da consulta)
  - Uma proteína da rede só pode corresponder a uma da consulta
  - Todas as proteínas da consulta são casadas com uma da rede
  - Somente vértices da solução podem ser casados

$$\sum_{q \in \Phi(v)} g_{vq} \leq 1 \quad \forall v \in V$$

$$\sum_{v \in V} g_{vq} = 1 \quad \forall q \in Q$$

$$g_{vq} \leq c_v \quad \forall v \in V, q \in \Phi(v)$$

Torque

- Garantindo validade de fluxos
  - f_{uv} e f_{vu} concordam em magnitude e direção
  - Fluxos só passam por arestas que fazem parte da solução
  - Nós fontes geram fluxos consumidos pelo ralo

$$
\begin{aligned}
f_{vu} &= -f_{uv} && \forall (v,u) \in E \\
f_{vu}, f_{uv} &\leq (k-1)e_{vu} && \forall (v,u) \in E \\
\sum_{u \in N(v)} f_{vu} &= c_v - k \cdot r_v && \forall v \in V
\end{aligned}
$$

Torque

- As restrições garantem que as soluções são da forma de uma subrede conexa com exatamente k vértices sequencialmente similares a seus casamentos da consulta

- Assim o objetivo é maximizar o peso das arestas da subrede solução

$$\max \sum_{(u,v)\in E} w(u,v)e_{uv}$$

Conclusão

- Utilização da análise comparativa de redes permite:
  - Melhorias na acurácia de predição funcional ao acrescentar análise comparativa de redes na predição
  - Componentes comuns entre espécies maiores e com maiores taxas de acerto