Relation between Graphs

Yangjing Long

Max Planck Intitute for Math. in the Sciences

Joint work with Jan Hubička, Jürgen Jost, Peter F. Stadler and Ling Yang

WGTC2012, Kaohsiung

Motivation

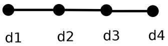


Figure: Domain interaction graph

Relation: containing relation

$$\{d_2, d_3 \in p_1; d_4 \in p_2; d_1, d_3 \in p_3; d_1 \in p_4\}$$

Motivation

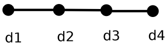


Figure: Domain interaction graph

Relation: containing relation

$$\{d_2, d_3 \in p_1; d_4 \in p_2; d_1, d_3 \in p_3; d_1 \in p_4\}$$

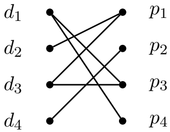
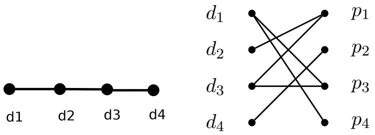


Figure: Relation

We can define a protein interaction graph by any two proteins have interaction iff there are two domains which belong to them respectively.



We can define a protein interaction graph by any two proteins have interaction iff there are two domains which belong to them respectively.

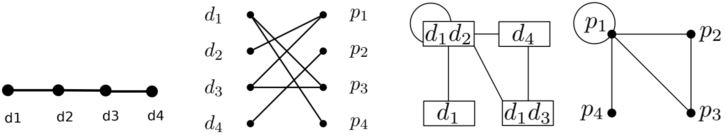
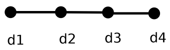


Figure: From Domain Interaction Network to Protein Interaction Network

Find relation between graphs



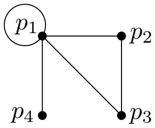


Figure: Domain interaction network Protein Interaction Network

relation R =

$$\{(d_2, p_1), (d_3, p_1), (d_4, p_2), (d_1, p_3), (d_3, p_3), (d_1, p_4)\}$$

Basic Definitions

Definition

The (binary) relation R between two sets A and B is a subset of $A \times B$.

Basic Definitions

Definition

The (binary) relation R between two sets A and B is a subset of $A \times B$.

Definition

Given a graph $G = (V_G, E_G)$, a finite set B and a binary relation $R \subset V_G \times B$, we define a new graph $H = G * R$ whose vertex set is B and for arbitrary $u, v \in B$, $(u, v) \in E_H$ iff one can find $x, y \in V_G$, such that $(x, u), (y, v) \in R$ and $(x, y) \in E_G$.

Basic Definitions

Definition

The (binary) relation R between two sets A and B is a subset of $A \times B$.

Definition

Given a graph $G = (V_G, E_G)$, a finite set B and a binary relation $R \subset V_G \times B$, we define a new graph $H = G * R$ whose vertex set is B and for arbitrary $u, v \in B$, $(u, v) \in E_H$ iff one can find $x, y \in V_G$, such that $(x, u), (y, v) \in R$ and $(x, y) \in E_G$.

Definition

Given two graphs G and H, if there exists a $R \subset V_G \times V_H$, such that $G * R = H$, we say R is a relation from G to H.

Definition

Graph homomorphism $G \to H$ is a mapping $f\colon V_G \to V_H$ such that $(x, y) \in E_G$ implies $(f(x), f(y)) \in E_H$.

Definition

Graph homomorphism $G \to H$ is a mapping $f\colon V_G \to V_H$ such that $(x, y) \in E_G$ implies $(f(x), f(y)) \in E_H$.

Definition

Graph multihomomorphism is a mapping $f : V_G \to 2^{V_H} \setminus \emptyset$ such that $(x, y) \in E_G$ implies $(u, v) \in E_H$ for every $u \in f(x)$ and $v \in f(x)$.

Definition

Graph homomorphism $G \to H$ is a mapping $f \colon V_G \to V_H$ such that $(x, y) \in E_G$ implies $(f(x), f(y)) \in E_H$.

Definition

Graph multihomomorphism is a mapping $f : V_G \to 2^{V_H} \setminus \emptyset$ such that $(x, y) \in E_G$ implies $(u, v) \in E_H$ for every $u \in f(x)$ and $v \in f(x)$.

Definition

Surjective multihomomorphism is a multihomomorphism such that pre-image of every vertex in H is non-empty and for every edge (u, v) in H we can find an edge (x, y) in G satisfying $u \in f(x)$, $v \in f(y)$.

Relation VS Surjective multihomomorphism

$R \subseteq V_G \times V_H$ $f : V_G \to 2^{V_H} \setminus \emptyset$

Relation VS **Surjective multihomomorphism**

$R \subseteq V_G \times V_H$ $\quad$ $f : V_G \to 2^{V_H} \setminus \emptyset$

Definition

R is **full** if for every u, there is $(u, v) \in R$.

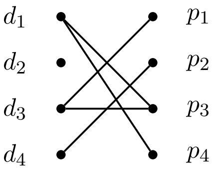


Figure: Non-full relation

Relation VS **Surjective multihomomorphism**

$R \subseteq V_G \times V_H$ $\quad$ $f : V_G \to 2^{V_H} \setminus \emptyset$

Definition

R is **full** if for every u, there is $(u, v) \in R$.

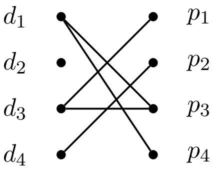


Figure: Non-full relation

If R is full and $G * R = H$, then R is a surjective multihomomorphism.

Relation VS Homomorphism

- If there is a surjective homomorphism $f : G \to H$, then $G * R = H$ for R corresponding to f.
- If $G * R = H$ and R is full, then there is a homomorphism $f : G \to H$.

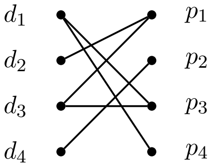


Figure: Homomorphism

Relation VS Homomorphism

- If there is a surjective homomorphism $f : G \to H$, then $G * R = H$ for R corresponding to f.
- If $G * R = H$ and R is full, then there is a homomorphism $f : G \to H$.

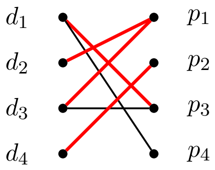


Figure: Homomorphism

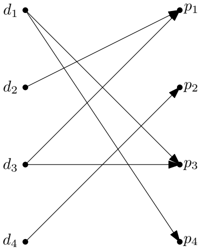
Compose and Decompose

Graph relations compose, i.e., $(G * R) * S = G * (R \circ S)$.

Compose and Decompose

Graph relations compose, i.e., $(G * R) * S = G * (R \circ S)$.

Graph relation R can be decomposed to a relation R_D duplicating vertices and a relation R_C contracting vertices.



Compose and Decompose

Graph relations compose, i.e., $(G * R) * S = G * (R \circ S)$.

Graph relation R can be decomposed to a relation R_D duplicating vertices and a relation R_C contracting vertices.

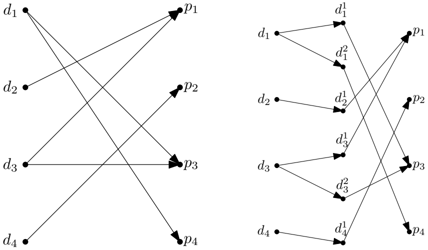
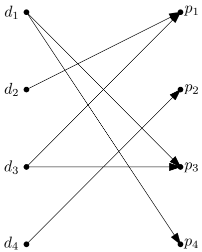
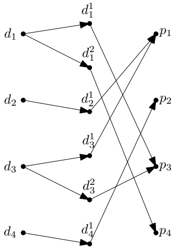
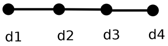
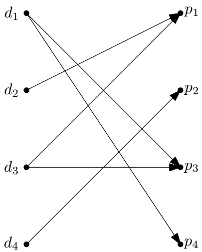


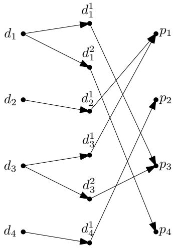
Figure: Decomposition

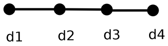

d_1
d_2
d_3
d_4
p_1
p_2
p_3
p_4

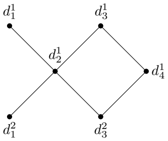

d_1
d_2
d_3
d_4
d_1^1
d_1^2
d_2^1
d_3^1
d_3^2
d_4^1
p_1
p_2
p_3
p_4

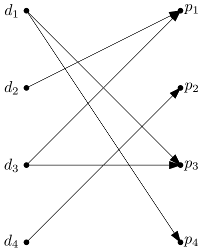

d1
d2
d3
d4

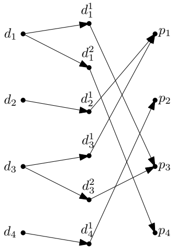

d_1
d_2
d_3
d_4
p_1
p_2
p_3
p_4

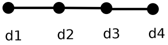

d_1
d_2
d_3
d_4
d_1^1
d_1^2
d_2^1
d_3^1
d_3^2
d_4^1
p_1
p_2
p_3
p_4

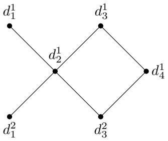

d1
d2
d3
d4

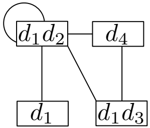

d_1^1
d_3^1
d_2^1
d_4^1
d_1^2
d_3^2

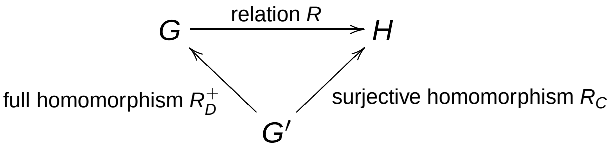

d_1
d_2
d_3
d_4
p_1
p_2
p_3
p_4


d_1
d_2
d_3
d_4
d_1^1
d_1^2
d_2^1
d_3^1
d_3^2
d_4^1
p_1
p_2
p_3
p_4


d1
d2
d3
d4


d_1^1
d_3^1
d_2^1
d_4^1
d_1^2
d_3^2


d_1d_2
d_4
d_1
d_1d_3



Full homomorphism: preserve both edge and non-edge

Surjective homomorphism: contraction

Let $\leq_{\mathrm{P}}^{\mathrm{Tur}}$ indicate polynomial time Turing reduction.
Fix a graph H,

- *Homomorphism problem* $\textsc{Hom}(H)$
- *Full relation problem* $\textsc{Ful-Rel}(H)$
- *Surjective homomorphis problem* $\textsc{Sur-Hom}(H)$

Theorem

For finite H our relation problem sits in the following relationship:

$$\textsc{Hom}(H) \leq_P^{Tur} \textsc{Ful-Rel}(H) \leq_P^{Tur} \textsc{Sur-Hom}(H).$$

$G * R = G$

Theorem

All solutions of $G * R = G$ are automorphisms if and only if G has property that the neighborhoods of vertices do not contain each other.

In graph homomorphism, core of graph G is the smallest subgraph of G which G has homomorphism to.

Core of graph G is unique up to isomorphism, so we can it the core, denote it by G_{core}.

Example

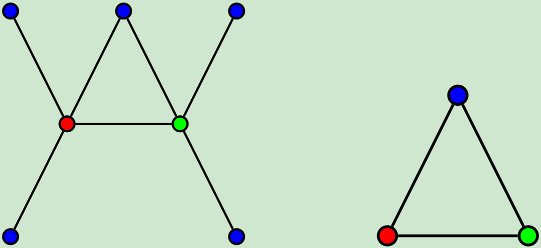


Figure: G and its core

Why core?

- G and G_{core} are homomorphically equivalent, i.e.,$G \to G_{\text{core}}$ and $G_{\text{core}} \to G$.
- Test whether G has homomorphism to H is equivalent to testing $G_{\text{core}} \to H_{\text{core}}$.
- Cores are the smallest graphs (in the number of vertices) in the class of homomorphism equivalence.
- Core is the smallest representative element in category theory.

Outline

Definition

Two graphs G and H are relationally equivalent, $G \backsim H$, if there are relations R and S such that $G * R = H$ and $H * S = G$.
If R and S are full, we write $G \backsim_f H$.

An R-core of a graph G, $G_{\text{R-core}}$, is the smallest graph in the same equivalence class of $\backsim_f$ as G.

Example

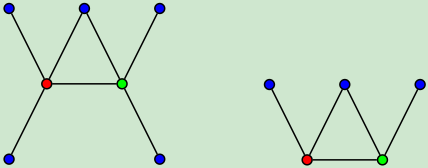


Figure: G and its R-core

R-core is unique up to isomorphism.

Proposition

$G_{\text{R-core}}$ is isomorphic to an induced subgraph of G.

If G is an R-core, then every relation R such that $G * R = G$ satisfies the Hall condition and thus contains a monomorphism.

Proposition

$G_{R\text{-core}}$ can be characterized by an algorithm, which removes all vertices $v \in G$ such that
(1) the neighborhood of v is union of neighborhood of some other vertices $v_1, v_2, \leq v_n$ and
(2) there is vertex u such that $N_G(v) \subseteq N_G(u)$.

Example

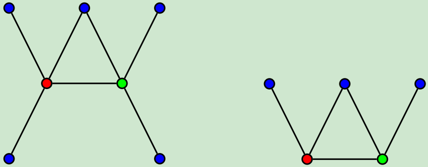


Figure: G and its R-core

Outline

Definition

A graph $G_{\text{R-reduced}}$ is R-reduced graph of G iff it is the smallest subgraph of G which has relation to G.

Proposition

a graph is R-reduced if and only if it is a graph core.

Example

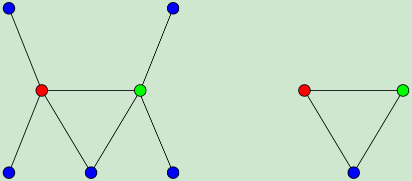


Figure: G and its R-reduced graph

Outline

1. Motivation
2. Homomorphisms and Relations
3. Computational Complexity
4. Core, Cocore, R-core
   - R-core
   - R-reduced graph
   - Cocore

There is always homomorphism from subgraph to graph.

There is always homomorphism from subgraph to graph.
This is not always the case with relations. We can reverse the notion of core to get. . .

There is always homomorphism from subgraph to graph.
This is not always the case with relations. We can reverse the notion of core to get. . .

Figure: From football to mathematics

Definition

A cocore of graph G is the smallest subgraph of G which has relation to G.

Cocore is unique up to isomorphism.

Example

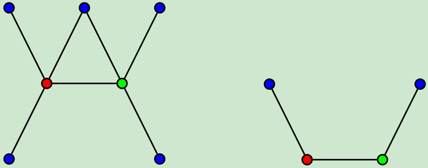


Figure: G and its cocore

Proposition

G is a cocore if and only if any vertex neighborhood is not the union of other vertex neighborhoods.

Proposition

Cocore of G is the smallest member of the equivalence class of relational equivalence $\backsim$.

Example

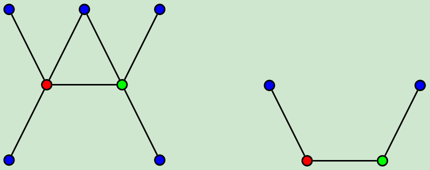


Figure: G and its cocore

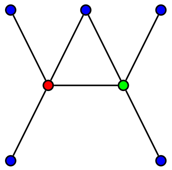


Figure: G

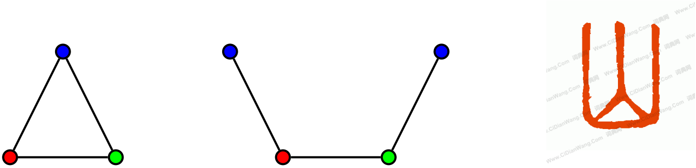


Figure: core cocore R-core

Thank you!