AMP Chain Graphs: Minimal Separators and Structure Learning Algorithms

Mohammad Ali Javidian, Pooyan Jamshidi, and Marco Valtorta
Department of Computer Science and Engineering

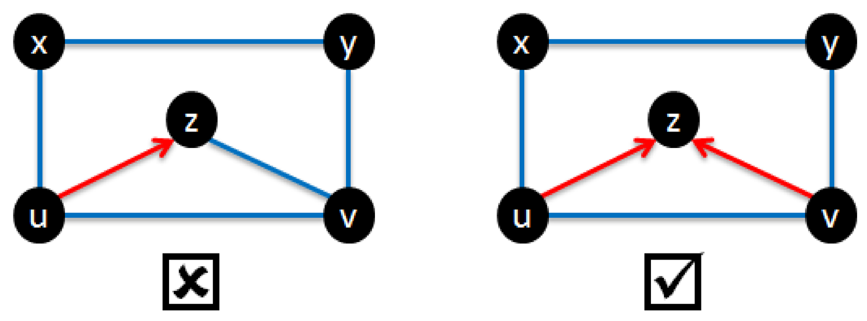
PURDUE UNIVERSITY
UNIVERSITY OF SOUTH CAROLINA

Introduction: AMP Chain Graph

Chain graphs (CGs):

- admit both directed and undirected edges,
- there are no partially directed cycles.

Example:

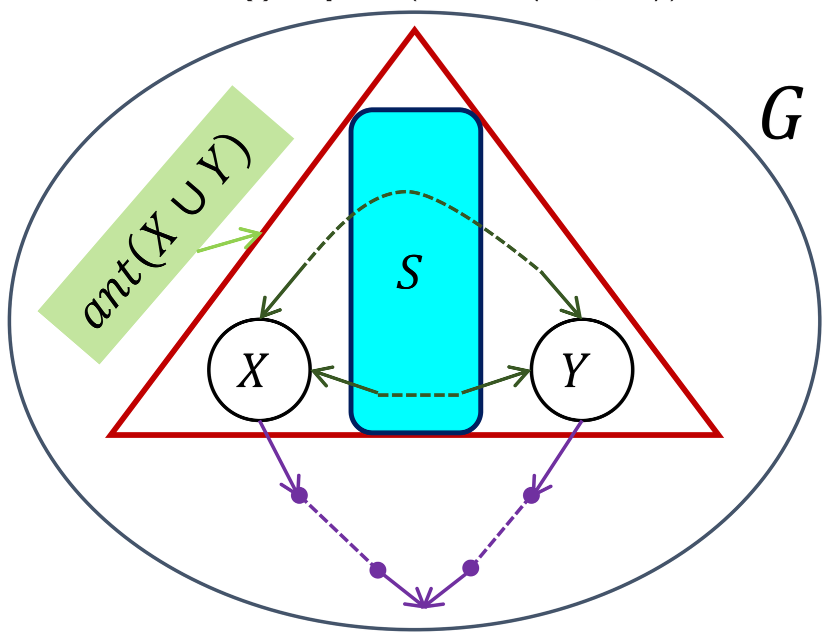


Chain graph models are *more expressive* than *Markov networks* and *Bayesian networks*.
Chain graphs enjoy different interpretations: LWF, AMP, MVR, ...
Here, we focus on chain graphs under the **A**ndersson–**M**adigan–**P**erlman (AMP) interpretation.

We address two problems: (1) finding minimal separators in AMP CGs and (2) learning the structure of AMP CGs from data.

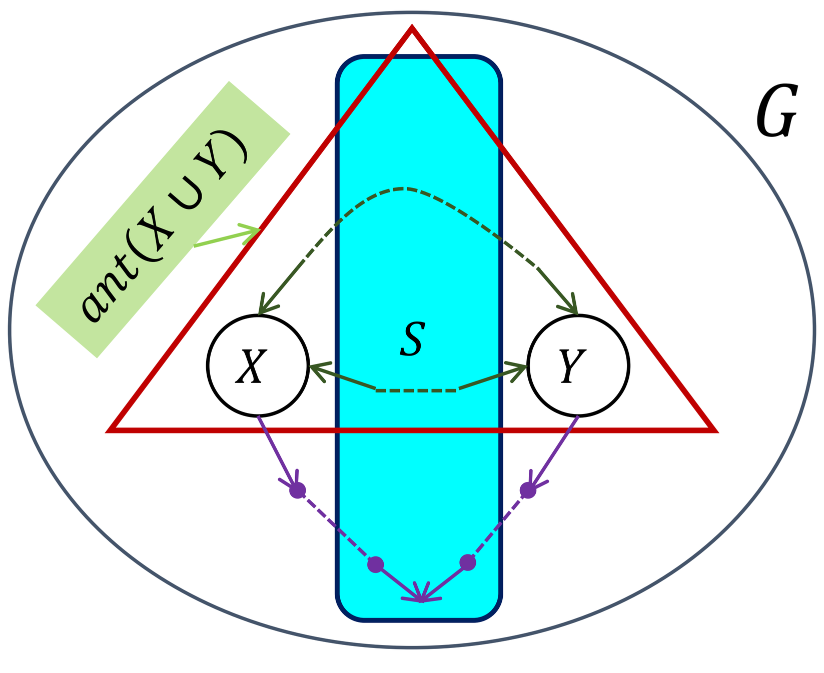
Main Theoretical Result 1

- **Problem 1** (test for minimal separation) Given two non-adjacent nodes X and Y in an AMP chain graph G and a set Z that separates X from Y, test if Z is minimal i.e., no proper subset of Z separates X from Y.
- **Theorem 1.**The problem of finding a minimal separating set for X and Y in an AMP chain graph G is equivalent to the problem of finding a minimal separating set for X and Y in the undirected graph $(G_{ant(X\cup Y)})^a$.



Main Theoretical Result 2

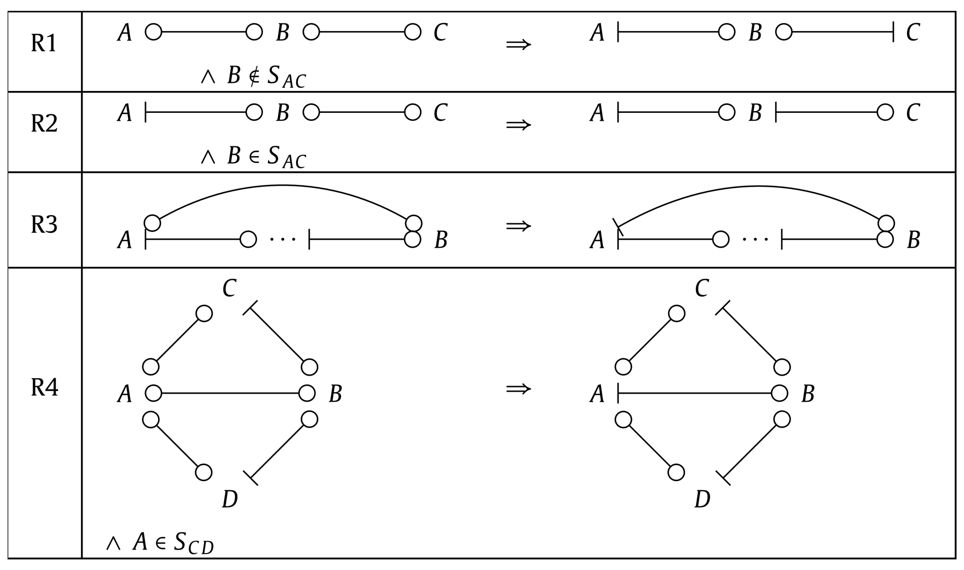
- **Problem 2** (restricted separation) Given two non-adjacent nodes X and Y in an AMP chain graph G and a set S of nodes not containing X and Y, find a subset Z of S that separates X from Y.
- **Theorem 1.**Given two nodes X and Y in an AMP CG G and a set S of nodes not containing X and Y, there exists some subset of S which separates X and Y if and only if the set $S' = S \cap ant(X \cup Y)$ separates X and Y.



PC-like Algorithm (Peña, 2012)

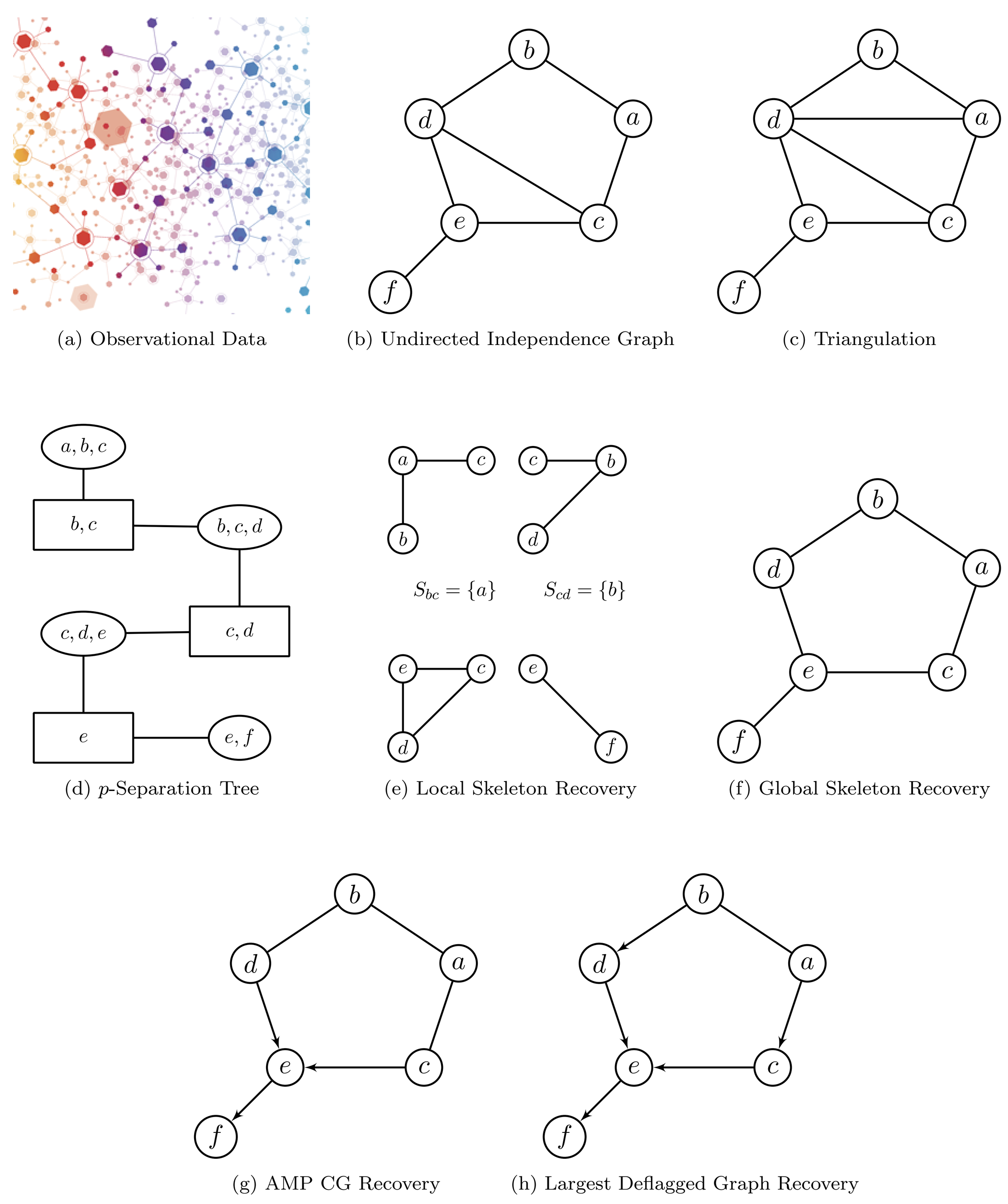
PC-LIKE Algorithm has two main phases:

/* **Skeleton Recovery:** */
for $i \leftarrow 0$ **to** $|V_H| - 2$ **do**
  while *possible* **do**
    Select any ordered pair of nodes u and v in H such that $u \in ad_H(v)$ and $|[ad_H(u) \cup ad_H(ad_H(u))] \setminus \{u,v\}| \geq i$, using order($V$);
    /* $ad_H(x) := \{y \in V | x \longrightarrow y, y \longrightarrow x,$ or $x — y\}$ */
    if *there exists* $S \subseteq ([ad_H(u) \cup ad_H(ad_H(u))] \setminus \{u,v\})$ *s.t.* $|S| = i$ *and* $u \perp\!\!\!\perp_p v | S$ *(i.e., u is independent of v given S in the probability distribution p)* **then**
      Set $S_{uv} = S_{vu} = S$;
      Remove the edge $u — v$ from H;
    end
  end
end
/* **Orientation phase:** */



LCD-AMP Algorithm

Main idea: Learning the structure of AMP chain graphs based on the idea of **decomposing** the learning problem by means of **separation trees** into a set of *smaller scale problems* on its *decomposed subgraphs*.



Implication: *Higher power of computational independence test*, better learned structure quality, along with the ability of exploiting **parallel computing**, make our proposed algorithms more desirable and suitable for **big data analysis** when AMP chain graphs are being used.

Stable-PC4AMP Algorithm

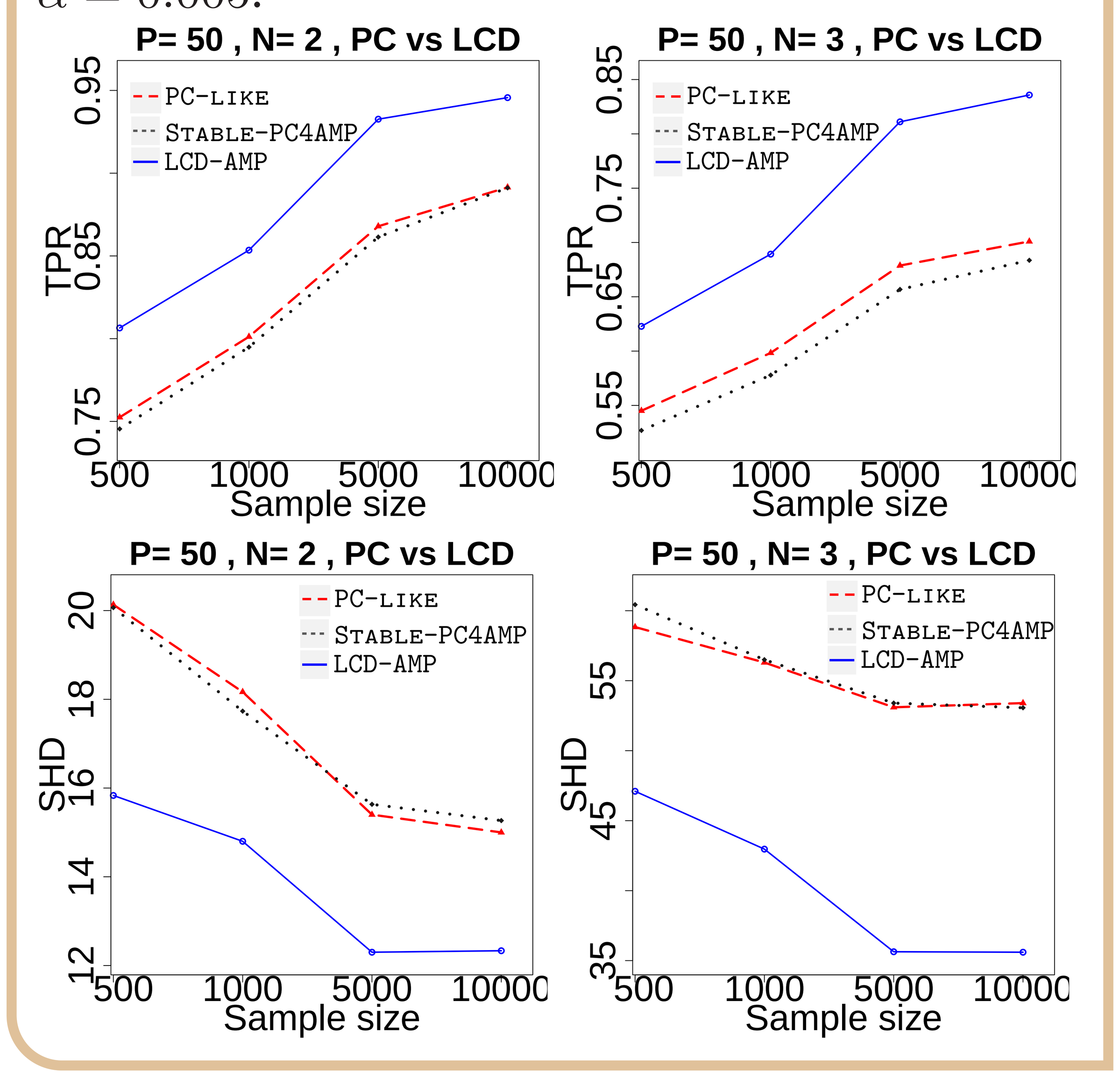
PC-LIKE algorithm is *order-dependent*: the output graph can **depend on the order** in which the variables are given.

/* **Skeleton Recovery:** */
for $i \leftarrow 0$ **to** $|V_H| - 2$ **do**
  for $j \leftarrow 1$ **to** $|V_H|$ **do**
    Set $a_H(v_j) = ad_H(v_j) \cup ad_H(ad_H(v_j))$;
    /* $ad_H(x) := \{y \in V | x \longrightarrow y, y \longrightarrow x,$ or $x — y\}$ */
  end
  while *possible* **do**
    Select any ordered pair of nodes u and v in H such that $u \in ad_H(v)$ and $|a_H(u) \setminus \{u,v\}| \geq i$, using order($V$);
    if *there exists* $S \subseteq (a_H(u) \setminus \{u,v\})$ *s.t.* $|S| = i$ *and* $u \perp\!\!\!\perp_p v | S$ *(i.e., u is independent of v given S in the probability distribution p)* **then**
      Set $S_{uv} = S_{vu} = S$;
      Remove the edge $u — v$ from H;
    end
  end
end

Implication: CI tests at each level can be grouped and distributed over different cores of the computer, and the results can be integrated at the end of each level. This makes our algorithm usable in dealing with **big data** via taking advantage of **parallel computation**.

Experimental Evaluation

Performance metrics: TPR and Structural Hamming Distance (SHD).
Synthetic Data: average over 30 repetitions with 50 variables correspond to expected degree of nodes N= 2, 3, and the significance level $\alpha = 0.005$.



References

Paper:

Code: