Bayesian Networks

II. Probabilistic Independence and Separation in Graphs (part 2)

Prof. Dr. Lars Schmidt-Thieme, L. B. Marinho, K. Buza

Information Systems and Machine Learning Lab (ISMLL)
Institute of Economics and Information Systems
& Institute of Computer Science
University of Hildesheim
http://www.ismll.uni-hildesheim.de

Directed graphs

Definition 1. Let V be any set and

$$E \subseteq V \times V$$

be a subset of sets of ordered pairs of V. Then $G := (V, E)$ is called a **directed graph**. The elements of V are called **vertices** or **nodes**, the elements of E **edges**.

Let $e = (x, y) \in E$ be an edge, then we call the vertices x, y **incident** to the edge e. We call two vertices $x, y \in V$ **adjacent**, if there is an edge $(x, y) \in E$ or $(y, x) \in E$.

The set of all vertices with an edge from a given vertex $x \in V$ is called its **fanout**:

$$\text{fanout}(x) := \{y \in V \mid (x, y) \in E\}$$

The set of all vertices with an edge to a given vertex $x \in V$ is called its **fanin**:

$$\text{fanin}(x) := \{y \in V \mid (y, x) \in E\}$$

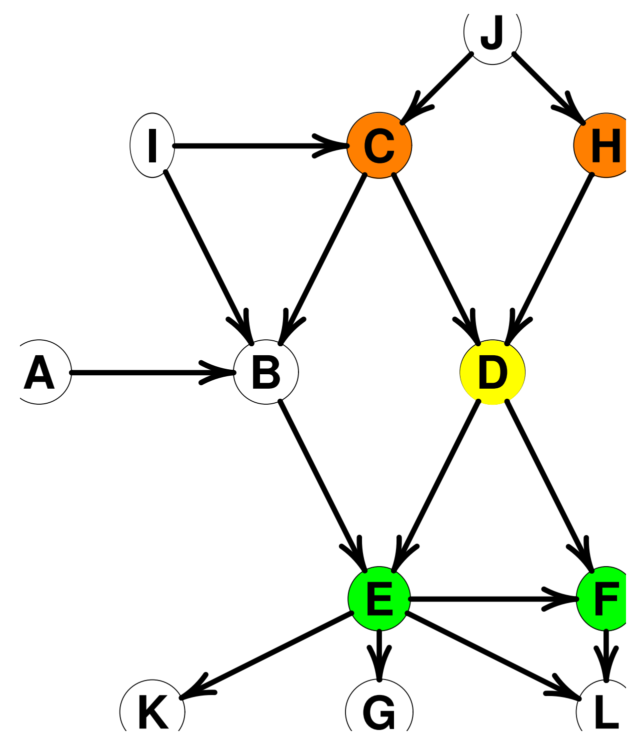


Figure 1: Fanin (orange) and fanout (green) of a node (blue).

Paths on directed graphs

Definition 2. Let $G = (V, E)$ be a directed graph. We call

$$G^* := V^*_{|G} := \{p \in V^* \,|\, (p_i, p_{i+1}) \in E, \\ i = 1, \ldots, |p| - 1\}$$

the **set of paths on** G. For two vertices $x, y \in V$ we denote by

$$G^*_{[x,y]} := \{p \in V^*_{|G} \,|\, p_1 = x, p_{|p|} = y\}$$

the **set of paths from** x **to** y.

The notions of **subpath**, **interior**, and **proper path** carry over to directed graphs.

A proper path $p = (p_1, \ldots, p_n) \in G^*$ with $p_1 = p_n$ is called **cyclic**. A path without cyclic subpath is called a **simple path**. A graph without a cyclic path is called **directed acyclig graph (DAG)**.

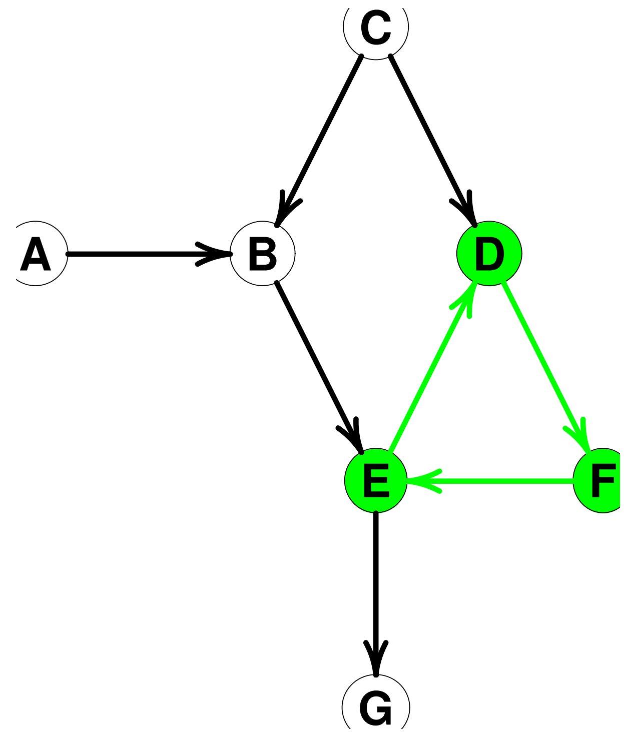


Figure 2: Example for a cycle.

Paths on directed graphs (2/2)

Definition 3. For a DAG G vertices of the fanout are also called **children**

$$\text{child}(x) := \text{fanout}(x) := \{y \in V \,|\, (x, y) \in E\}$$

and the vertices of the fanin **parents**:

$$\text{pa}(x) := \text{fanin}(x) := \{y \in V \,|\, (y, x) \in E\}$$

Vertices y with a proper path from y to x are called **ancestors of** x:

$$\text{anc}(x) := \{y \in V \,|\, \exists p \in G^* : |p| \geq 2, \; p_1 = y, p_{|p|} = x\}$$

Vertices y with a proper path from x to y are called **descendents of** x:

$$\text{desc}(x) := \{y \in V \,|\, \exists p \in G^* : |p| \geq 2, \; p_1 = x, p_{|p|} = y\}$$

Vertices that are not a descendent of x are called **nondescendents of** x.

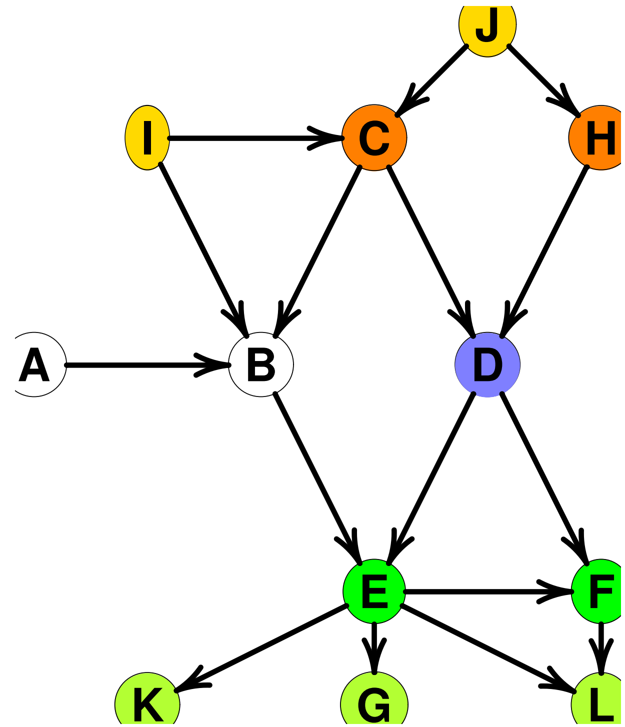


Figure 3: Parents/Fanin (orange) and additional ancestors (light orange), children/fanout (green) and additional descendants (light green) of a node (blue).

Chains

Definition 4. Let $G := (V, E)$ be a directed graph. We can construct an **undirected pendant** $u(G) := (V, u(E))$ **of** G by dropping the directions of the edges:

$$u(E) := \{\{x, y\} \mid (x, y) \in E \text{ or } (y, x) \in E\}$$

The paths on $u(G)$ are called **chains of** G:

$$G^{\blacktriangle} := u(G)^*$$

i.e., a chain is a sequence of vertices that are linked by a forward or a backward edge. If we want to stress the directions of the linking edges, we denote a chain $p = (p_1, \ldots, p_n) \in G^{\blacktriangle}$ by

$$p_1 \leftarrow p_2 \rightarrow p_3 \leftarrow \cdots \leftarrow p_{n-1} \rightarrow p_n$$

The notions of **length**, **subchain**, **interior** and **proper** carry over from undirected paths to chains.

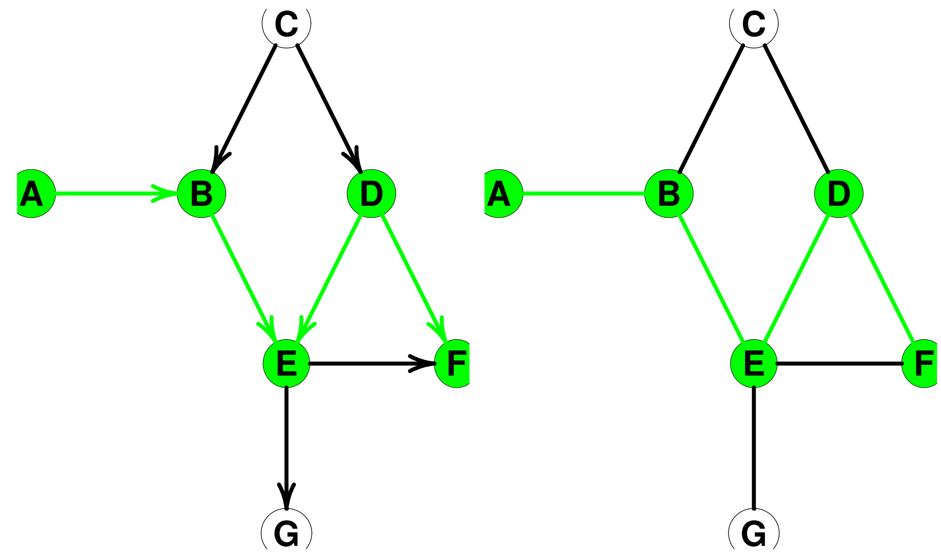


Figure 4: Chain (A, B, E, D, F) on directed graph and path on undirected pendant.

Blocked chains

Definition 5. Let $G := (V, E)$ be a directed graph. We call a chain

$$p_1 \rightarrow p_2 \leftarrow p_3$$

a **head-to-head meeting**.

Let $Z \subseteq V$ be a subset of vertices. Then a chain $p \in G^{\blacktriangle}$ is called **blocked at position** i **by** Z, if for its subchain (p_{i-1}, p_i, p_{i+1}) there is

$$\begin{cases} p_i \in Z, & \text{if not } p_{i-1} \rightarrow p_i \leftarrow p_{i+1} \\ p_i \notin Z \cup \operatorname{anc}(Z), & \text{else} \end{cases}$$

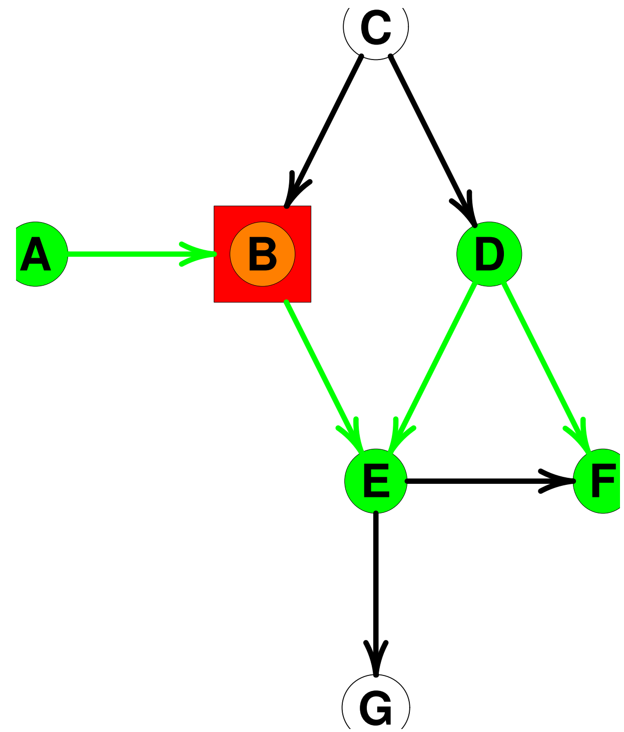


Figure 5: Chain (A, B, E, D, F) is blocked by $Z = \{B\}$ at 2.

Blocked chains / more examples

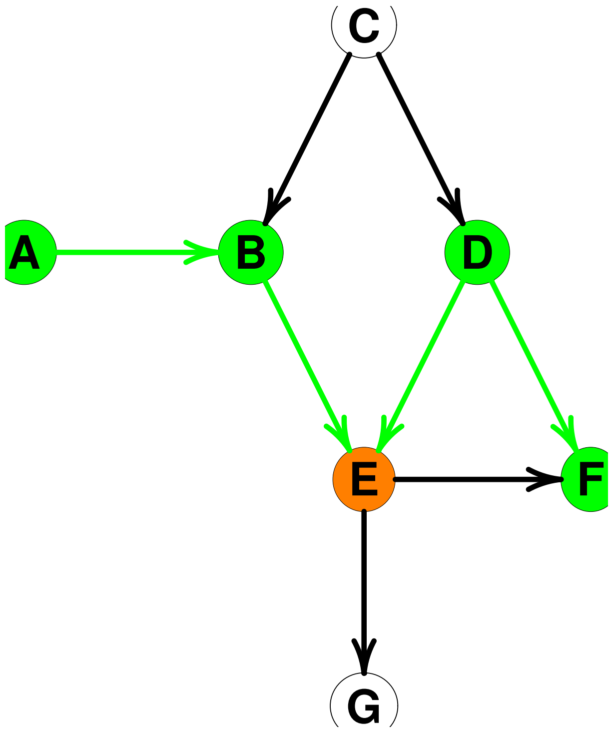


Figure 6: Chain (A, B, E, D, F) is blocked by $Z = \emptyset$ at 3.

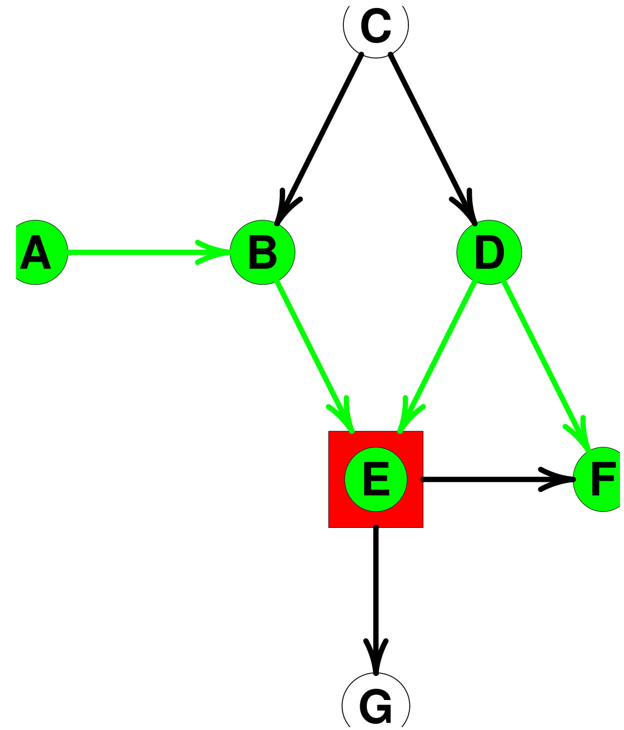
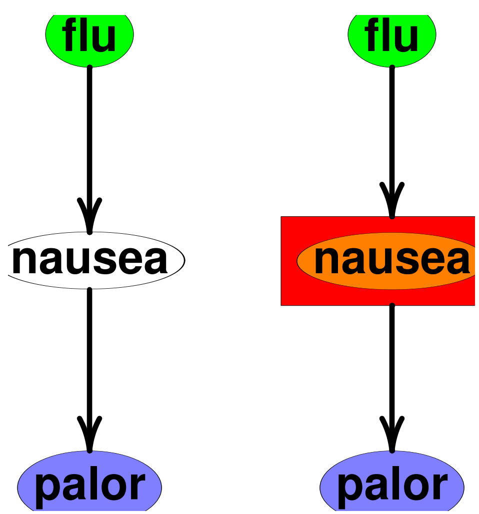


Figure 7: Chain (A, B, E, D, F) is **not** blocked by $Z = \{E\}$ at 3.

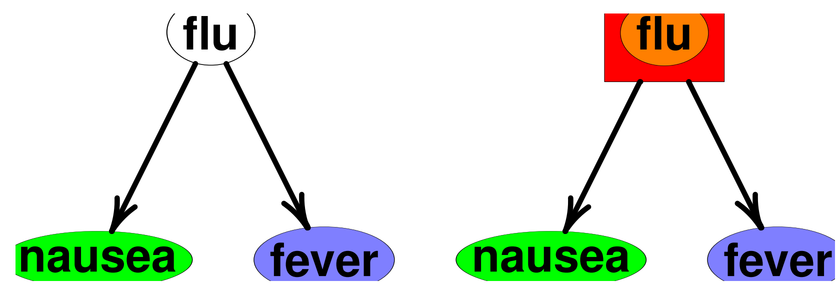
Blocked chains / rationale

The notion of blocking is choosen in a way so that chains model "flow of causal influence" through a causal network where the states of the vertices Z are already know.

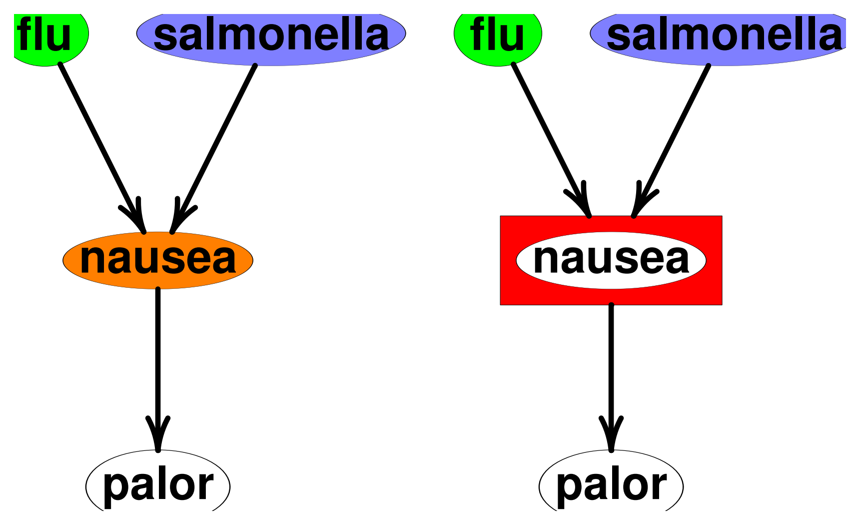
1) Serial connection / intermediate cause:



2) Diverging connection / common cause:



3) Converging connection / common effect:



Models "discounting" [Nea03, p. 51].

The moral graph

Definition 6. Let $G := (V, E)$ be a DAG.

As the **moral graph of** G we denote the undirected skeleton graph of G plus additional edges between each two parents of a vertex, i.e. $\mathrm{moral}(G) := (V, E')$ with

$$E' := u(E) \cup \{\{x, y\} \mid \exists z \in V : x, y \in \mathrm{pa}(z)\}$$

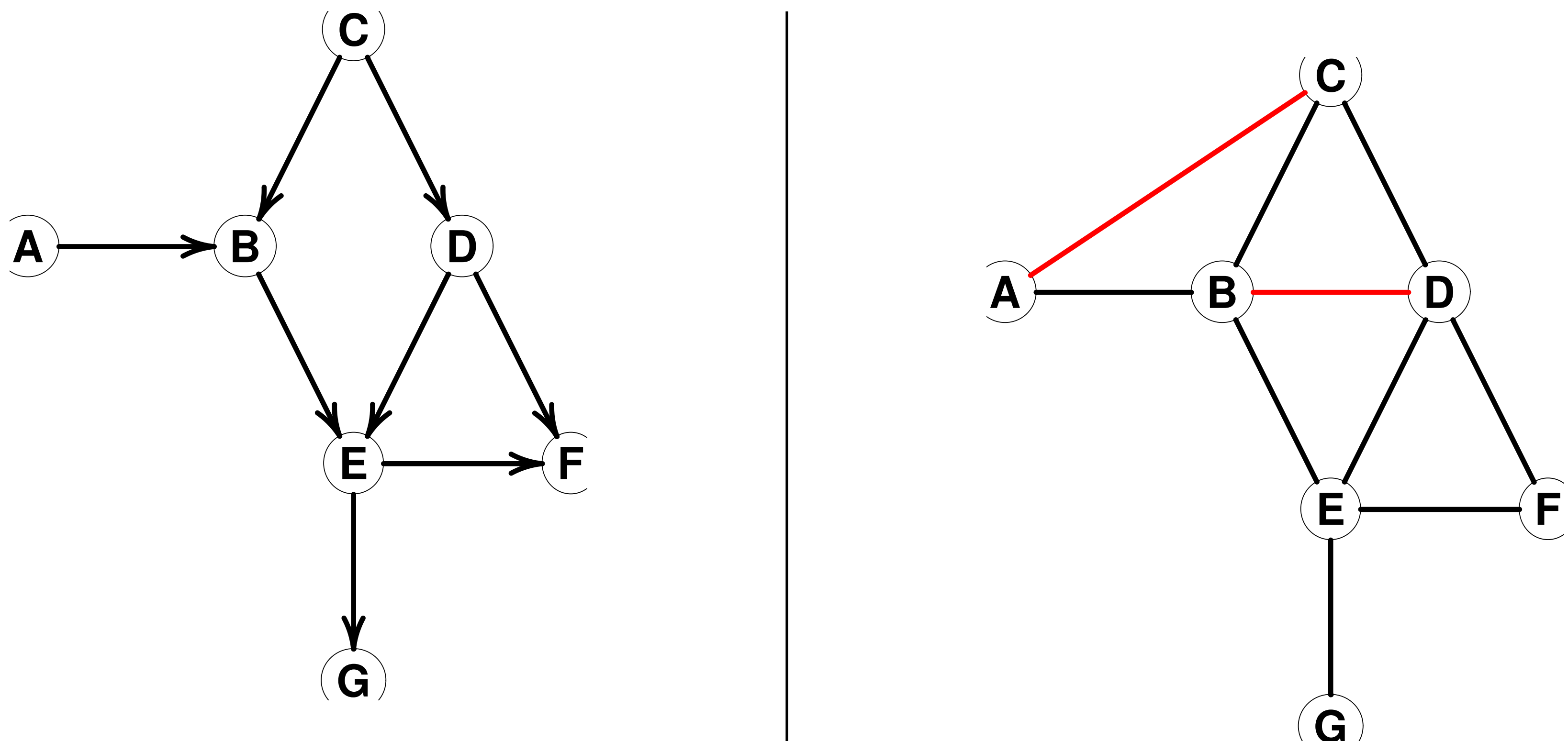


Figure 9: DAG and its moral graph.

Separation in DAGs (d-separation)

Let $G := (V, E)$ be a DAG.

Let $X, Y, Z \subseteq V$ be three disjoint subsets of vertices. We say, the vertices X and Y are **separated by** Z **in** G, if

(i) every chain from any vertex from X to any vertex from Y is blocked by Z

or equivalently

(ii) X and Y are u-separated by Z in the moral graph of the ancestral hull of $X \cup Y \cup Z$.

We write $I_G(X, Y|Z)$ for the statement, that X and Y are separated by Z in G.

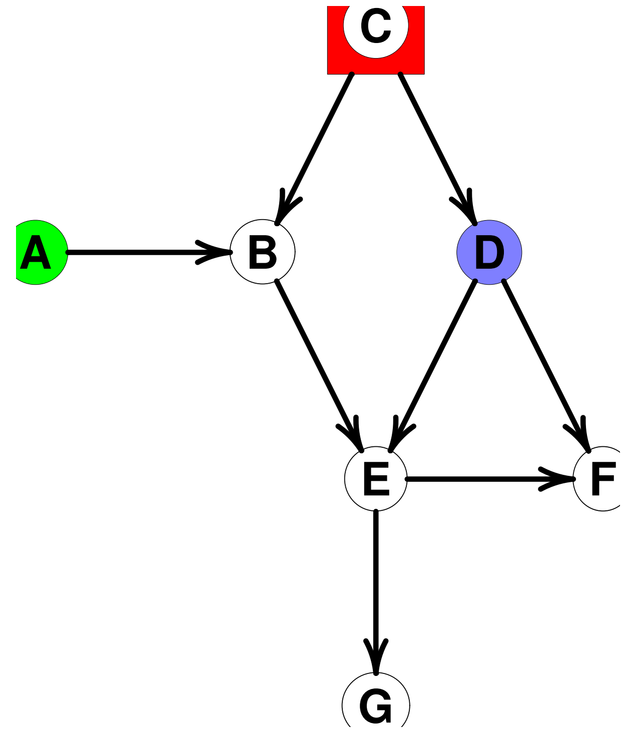


Figure 10: Are the vertices A and D separated by C in G?

Separation in DAGs (d-separation) / examples

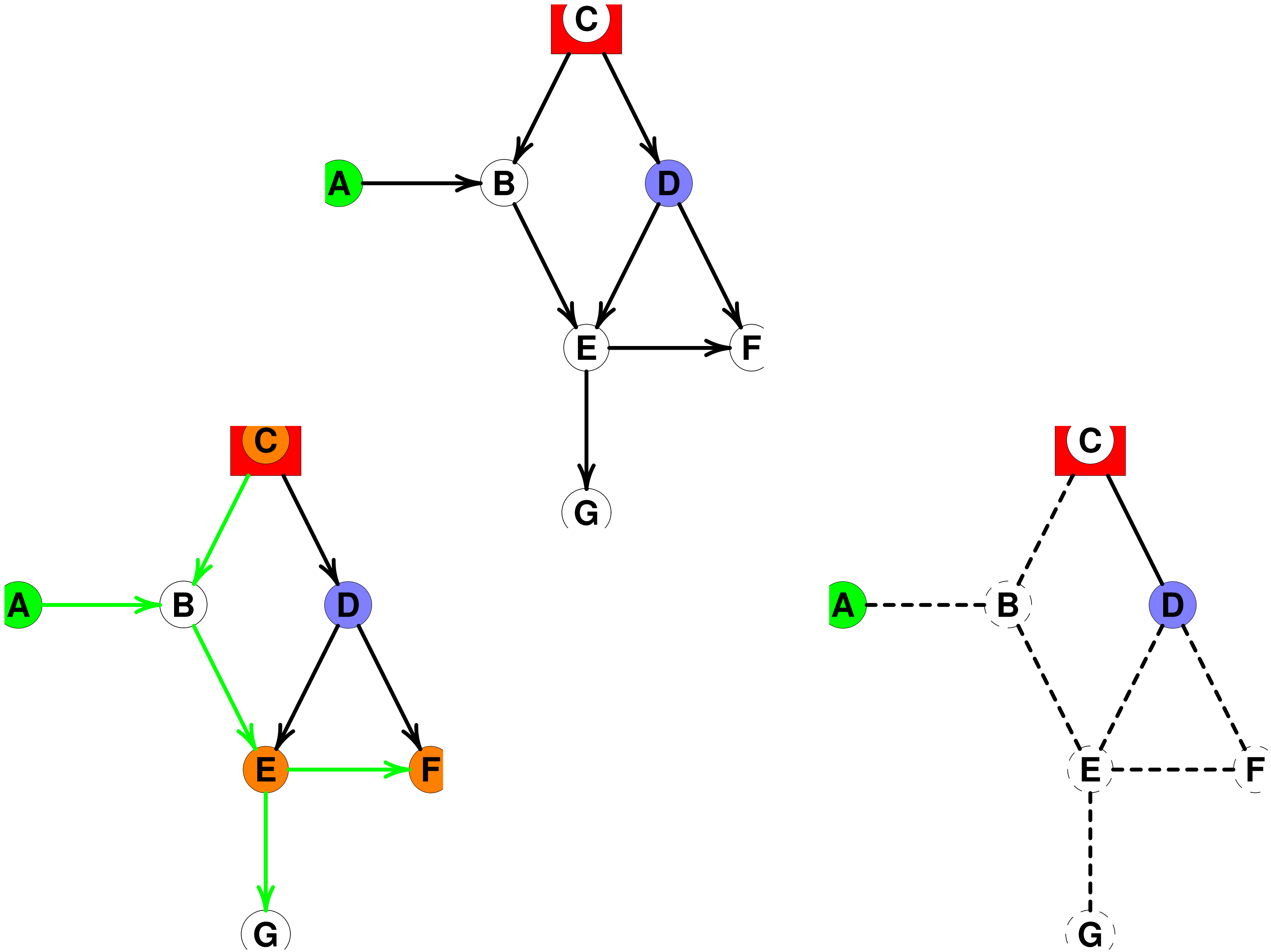


Figure 11: A and D are separated by C in G.

Separation in DAGs (d-separation) / more examples

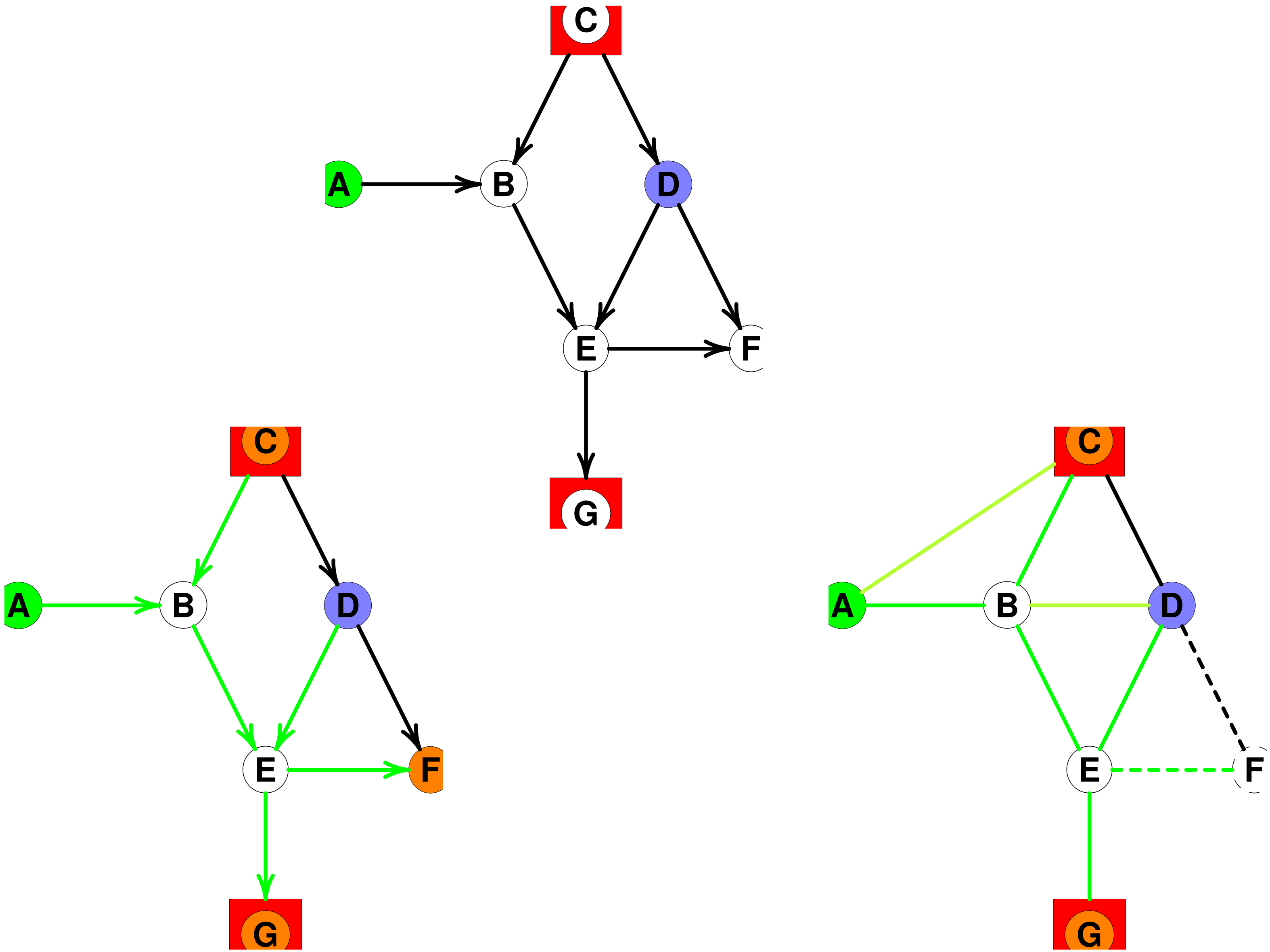


Figure 12: A and D are not separated by $\{C, G\}$ in G.

Checking d-separation

To test, if for a given graph $G = (V, E)$ two given sets $X, Y \subseteq V$ of vertices are d-separated by a third given set $Z \subseteq V$ of vertices, we may build the moral graph of the ancestral hull and apply the u-separation criterion.

$\text{check-d-separation}(G, X, Y, Z)$:
$G' := \text{moral}(\text{anc}_G(X \cup Y \cup Z))$
return *check-u-separation*(G', X, Y, Z)

Figure 13: Algorithm for checking d-separation via u-separation in the moral graph.

A drawback of this algorithm is that we have to rebuild the moral graph of the ancestral hull whenever X or Y changes.

Checking d-separation

Instead of constructing a moral graph, we can modify **a breadth-first search for chains** to find all vertices not d-separated from X by Z in G.

The breadth-first search must not hop over head-to-head meetings with the middle vertex not in Z nor having an descendent in Z.

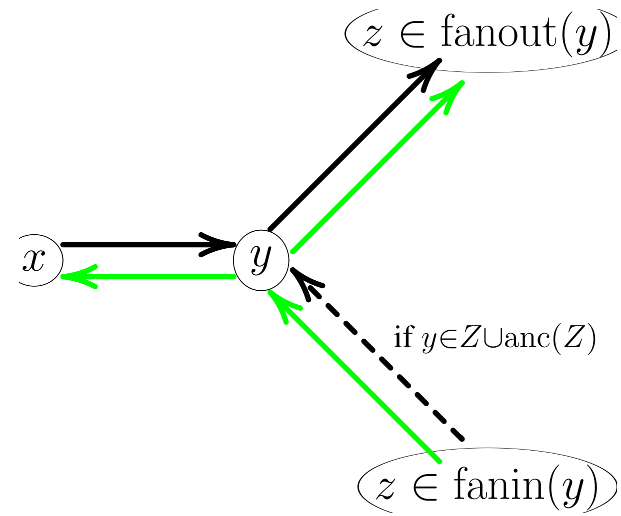


Figure 14: Restricted breadth-first search of non-blocked chains.

enumerate-d-separation$(G = (V, E), X, Z)$:
borderForward $:= \emptyset$
borderBackward $:= X \setminus Z$
reached $:= \emptyset$
while *borderForward* $\neq \emptyset$ or *borderBackward* $\neq \emptyset$ **do**
    reached $:=$ *reached* $\cup$ (*borderForward* $\setminus Z$) $\cup$ *borderBackward*
    borderForward $:= \mathrm{fanout}_G$(*borderBackward* $\cup$ (*borderForward* $\setminus Z$)) $\setminus$ *reached*
    borderBackward $:= \mathrm{fanin}_G$(*borderBackward* $\cup$ (*borderForward* $\cap (Z \cup \mathrm{anc}(Z))$)) $\setminus Z \setminus$ *reached*
od
return $V \setminus$ *reached*

Figure 15: Algorithm for enumerating all vertices d-separated from X by Z in G via restricted breadth-first search (see [Nea03, p. 80–86] for another formulation).

Properties of d-separation / no strong union

For d-separation the strong union property does not hold.

I is called **strongly unionable**, if

$$I(X, Y|Z) \Rightarrow I(X, Y|Z \cup Z') \quad \text{for all } Z' \text{ disjunct with } X, Y$$

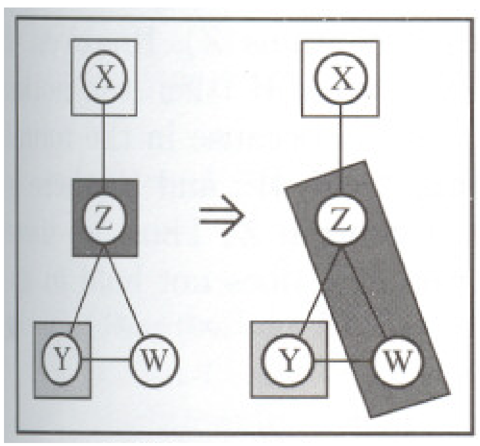


Figure 16: Example for strong union in undirected graphs (u-separation) [CGH97, p. 189].

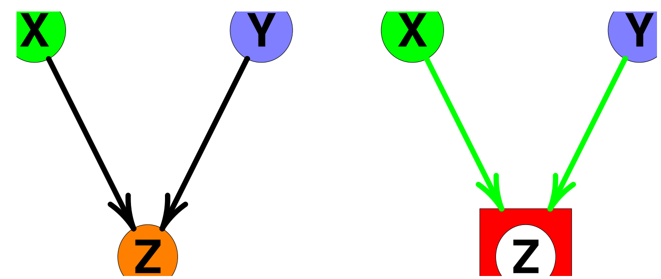


Figure 17: Counterexample for strong unions in DAGs (d-separation).

Properties of d-separation / no strong transitivity

For d-separation the strong transitivity property does not hold.

I is called **strongly transitive**, if

$$I(X, Y|Z) \Rightarrow I(X, \{v\}|Z) \text{ or } I(\{v\}, Y|Z) \quad \forall v \in V \setminus Z$$

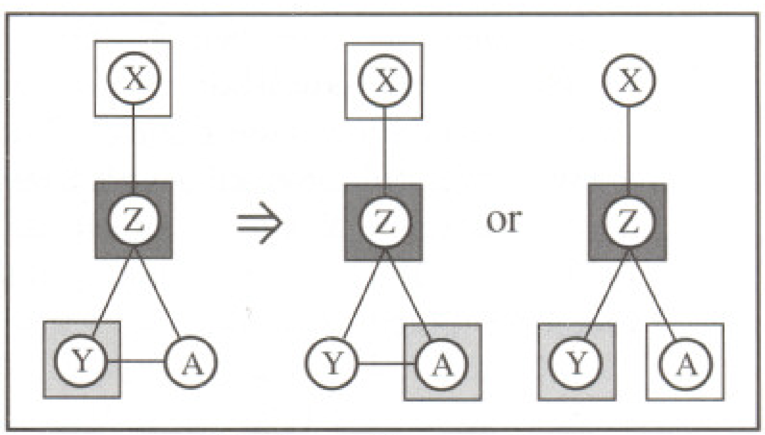


Figure 18: Example for strong transitivity in undirected graphs (u-separation) [CGH97, p. 189].

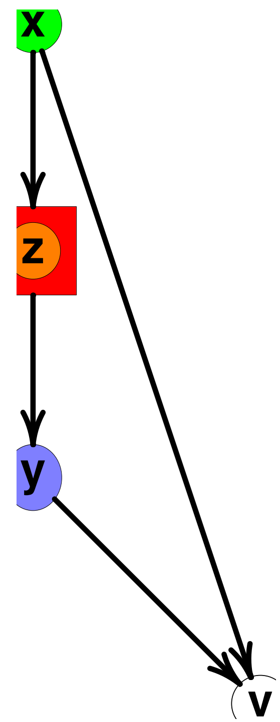


Figure 19: Counterexample for strong transitivity in DAGs (d-separation).

Properties of d-separation

relation	symmetry	decomposition	composition	strong union	weak union	contraction	intersection	strong transitivity	weak transitivity	chordality
u-separation	+	+	+	+	+	+	+	+	+	−
d-separation	+	+	+	−	+		+	−	+	+

References

[CGH97] Enrique Castillo, José Manuel Gutiérrez, and Ali S. Hadi. *Expert Systems and Probabilistic Network Models*. Springer, New York, 1997.

[Nea03] Richard E. Neapolitan. *Learning Bayesian Networks*. Prentice Hall, 2003.