WPMSIIP 2013

credal networks under epistemic irrelevance

Jasper De Bock

7 September 2013, Lugano

UNIVERSITEIT GENT

Research group SYSTeMS

Jasper De Bock

Gert de Cooman

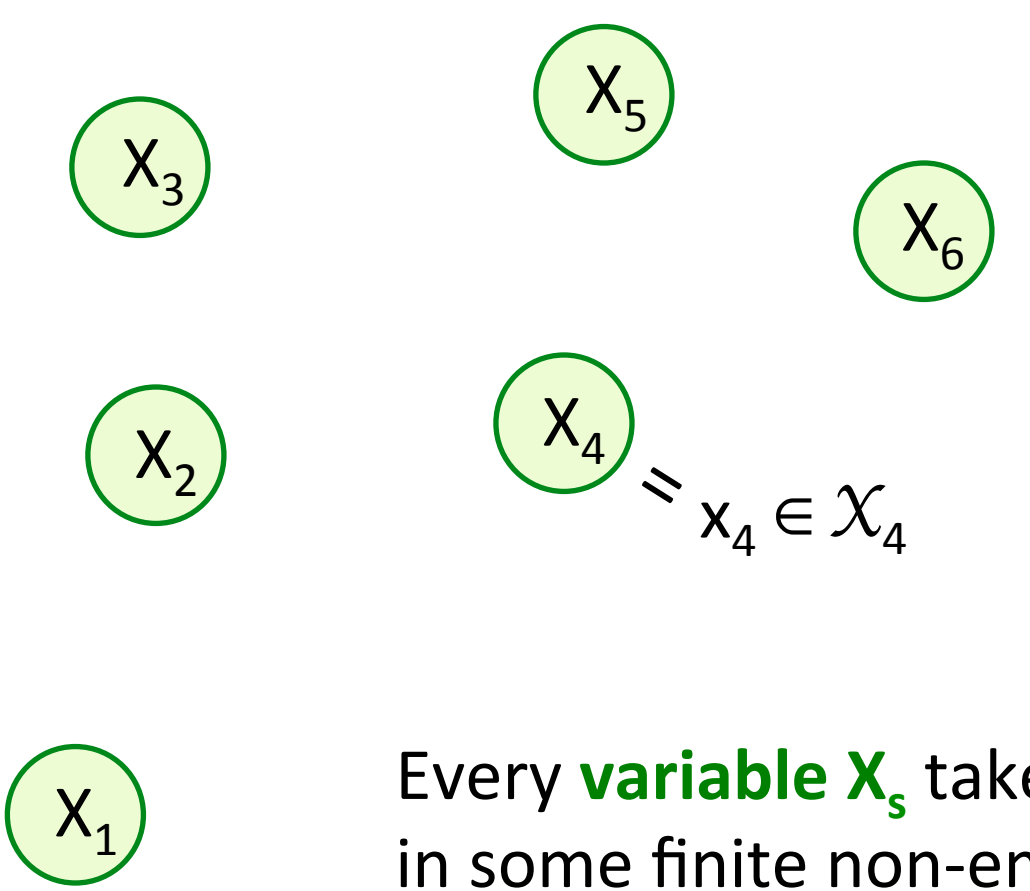
Stavros Lopatatzidis

Arthur Van Camp

Erik Quaeghebeur

Márcio A. Diniz

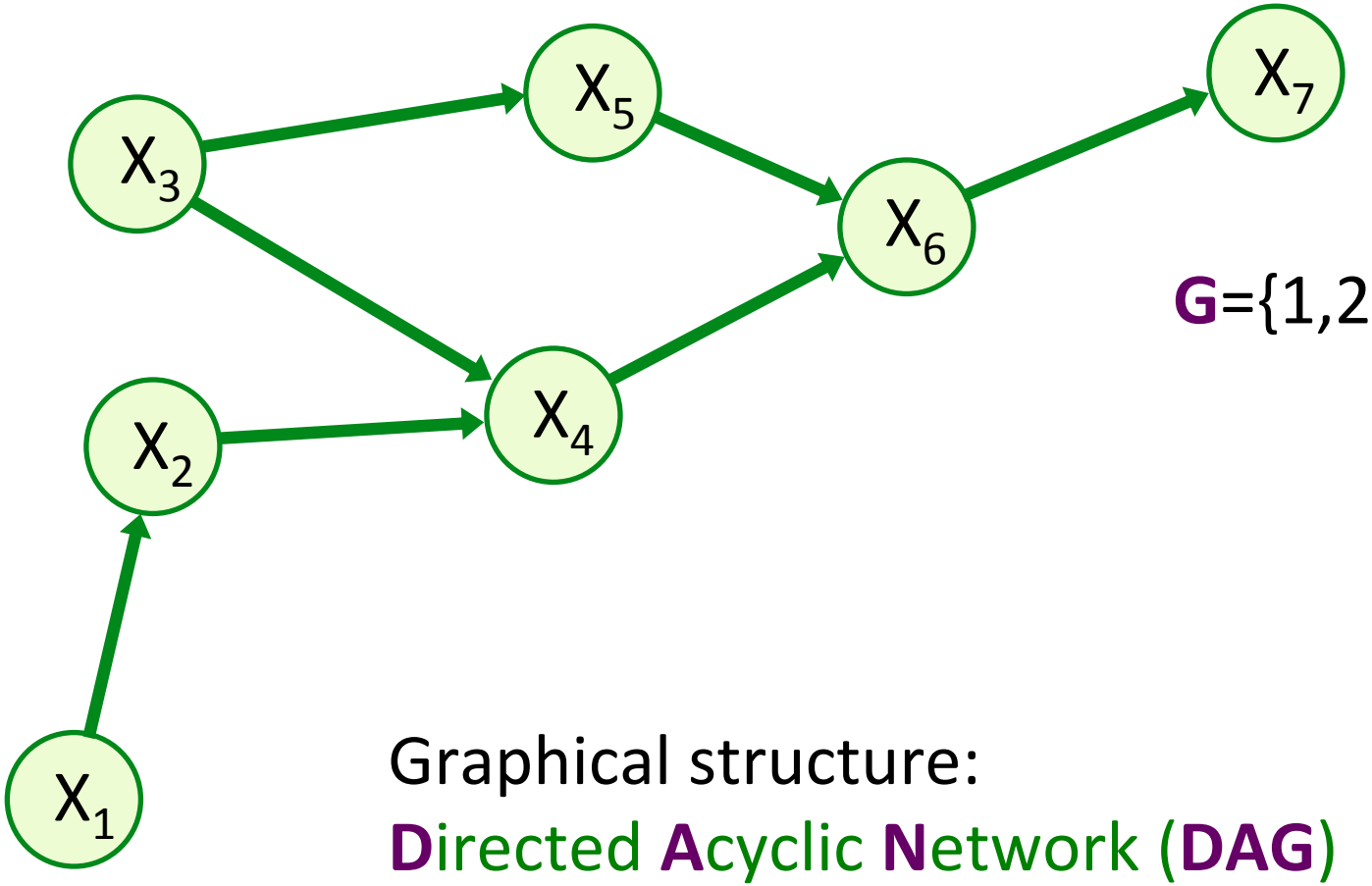
Credal networks: basic setup



A finite number of **variables**

Every **variable X_s** takes values x_s in some finite non-empty set $\mathcal{X}_s$

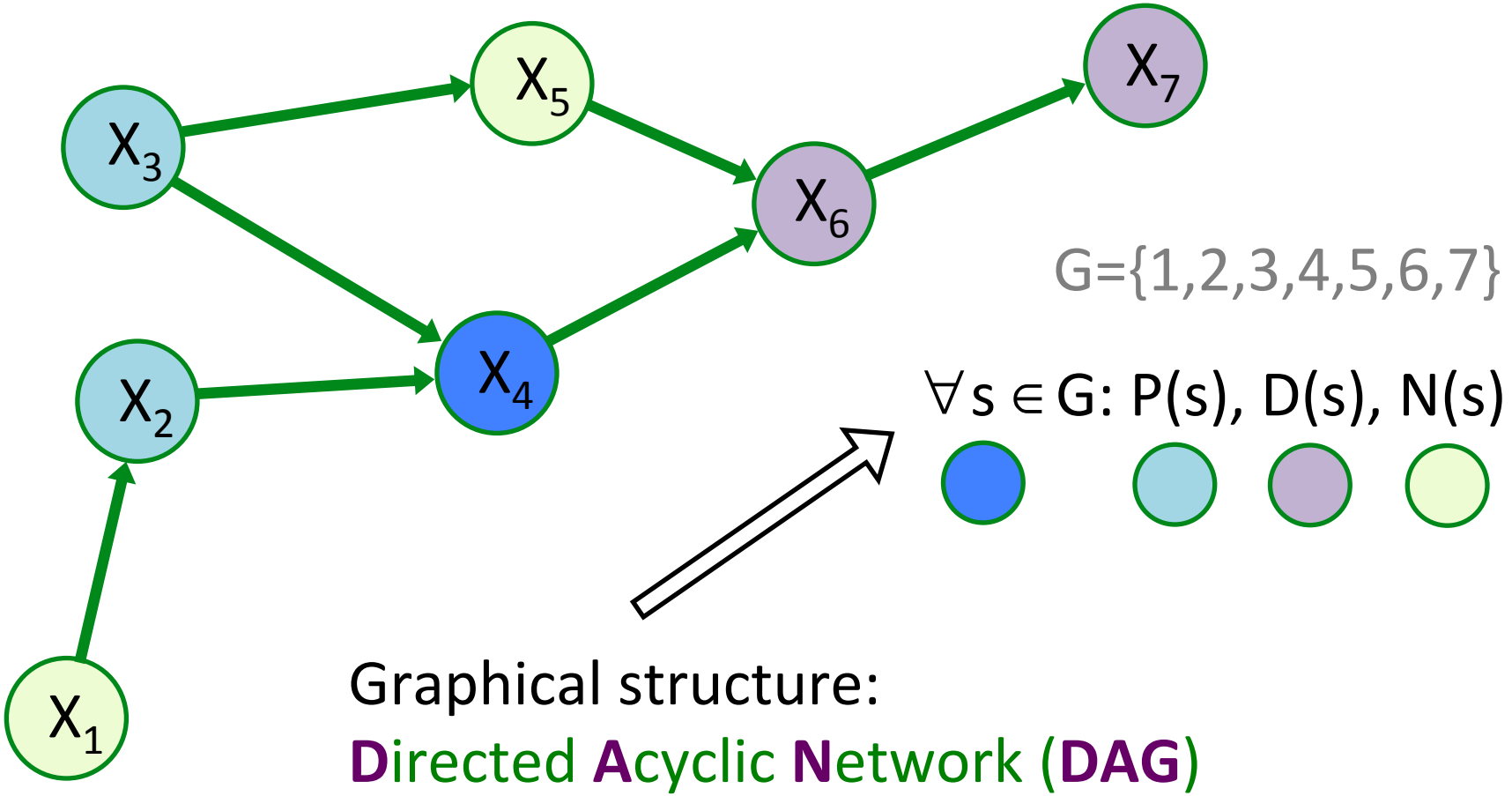
Credal networks: basic setup



G={1,2,3,4,5,6,7}

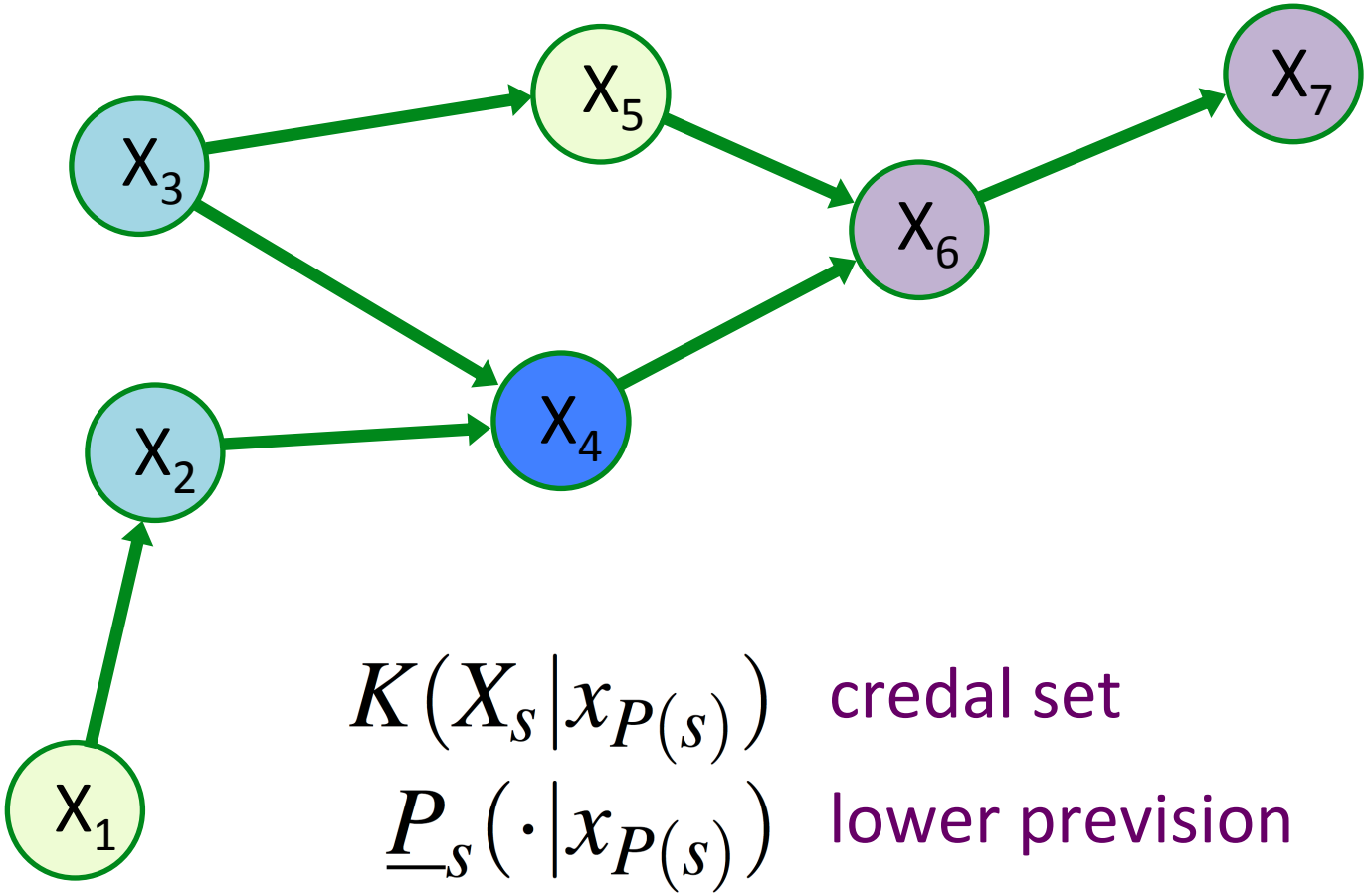
Graphical structure:
Directed **A**cyclic **N**etwork (**DAG**)

Credal networks: basic setup



Graphical structure:
Directed **A**cyclic **N**etwork (**DAG**)

Credal networks: local uncertainty models

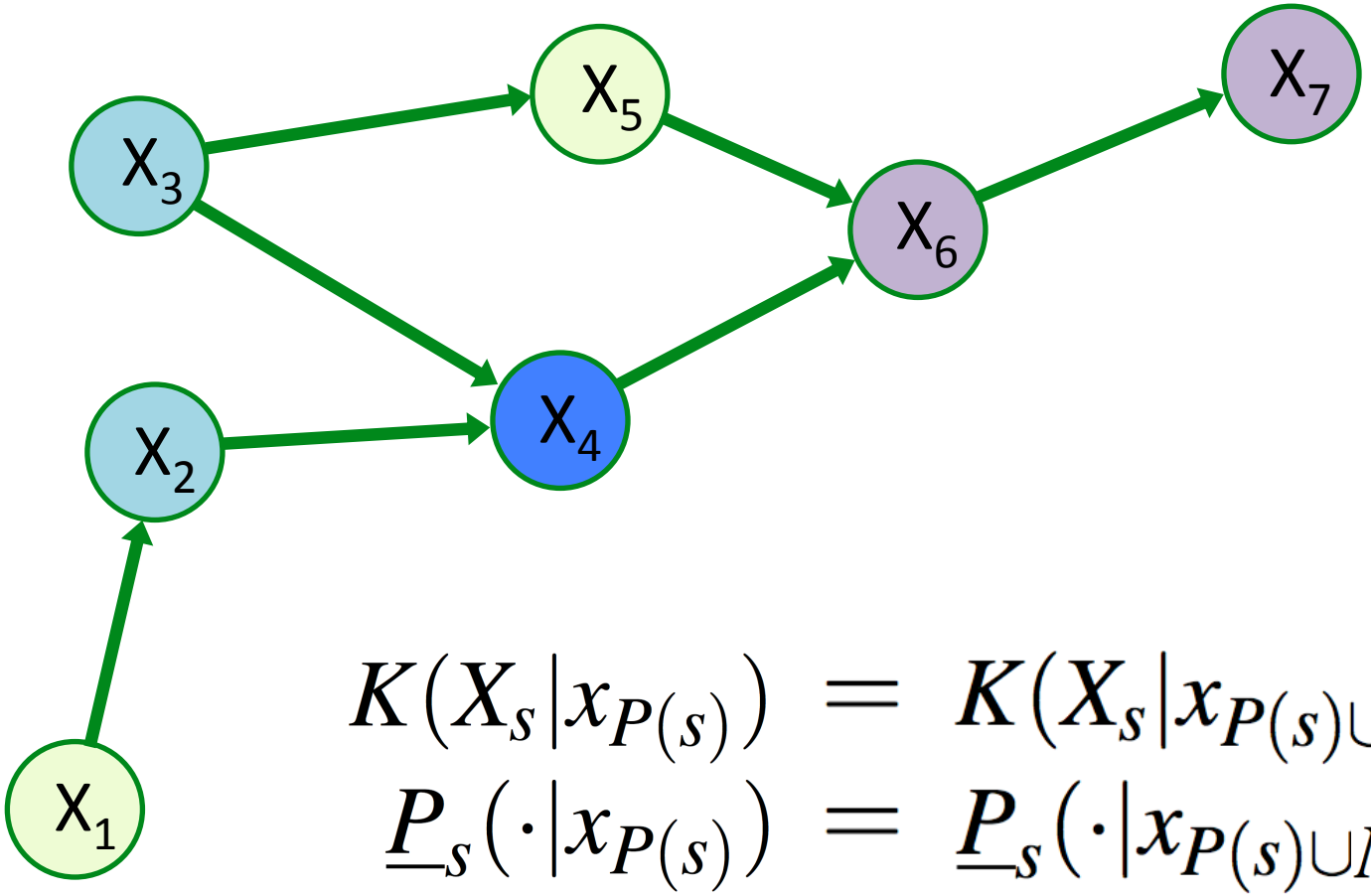


$K(X_s|x_{P(s)})$ credal set

$\underline{P}_s(\cdot|x_{P(s)})$ lower prevision

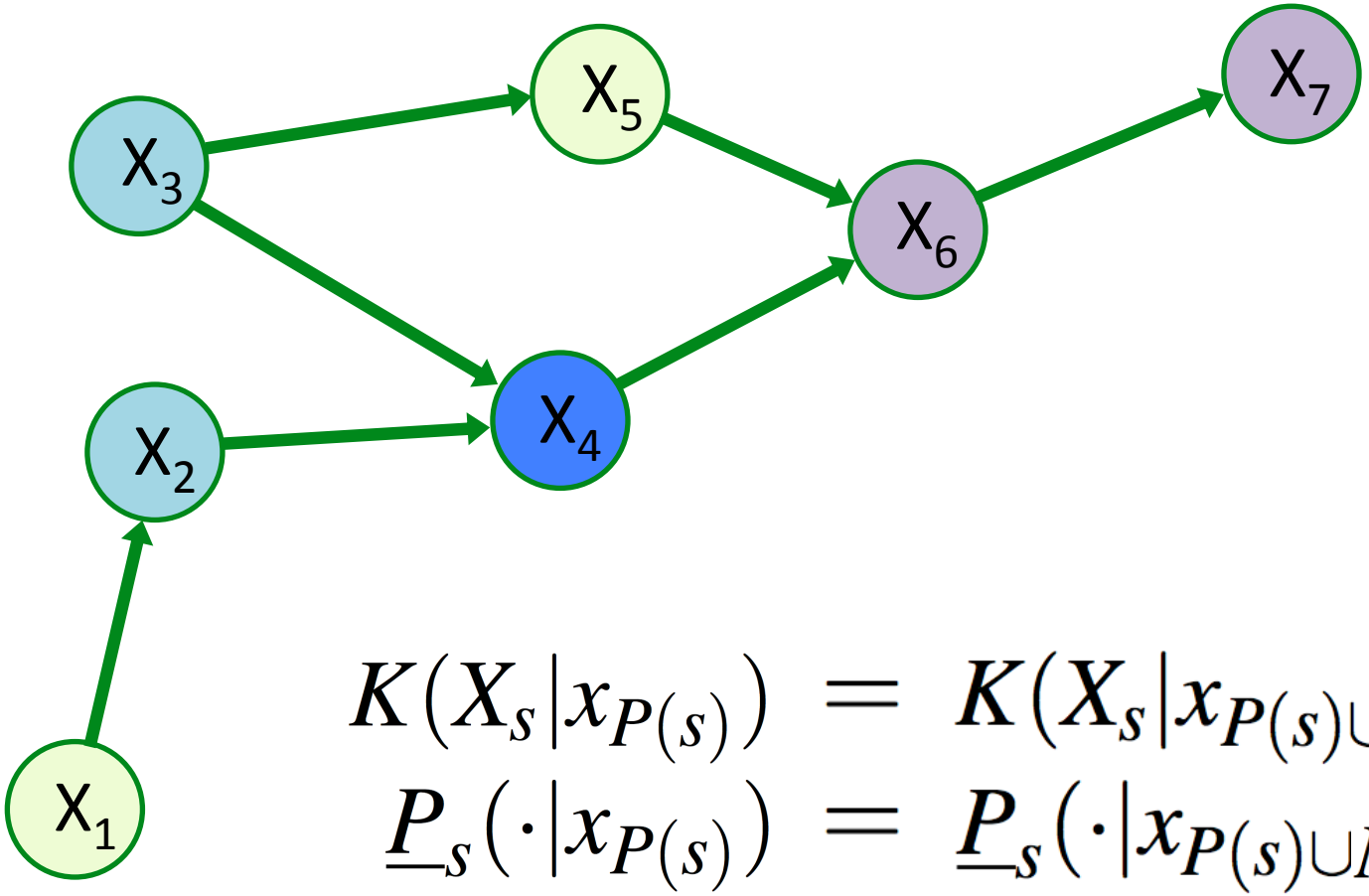
$\mathscr{D}_s|x_{P(s)}$ set of desirable gambles

Credal networks: epistemic irrelevance



$$K(X_s|x_{P(s)}) = K(X_s|x_{P(s)\cup N(s)})$$
$$\underline{P}_s(\cdot|x_{P(s)}) = \underline{P}_s(\cdot|x_{P(s)\cup N(s)})$$
$$\mathscr{D}_s]x_{P(s)} = \mathscr{D}_s]x_{P(s)\cup N(s)}$$

Credal networks: a joint model



$K(X_G)$

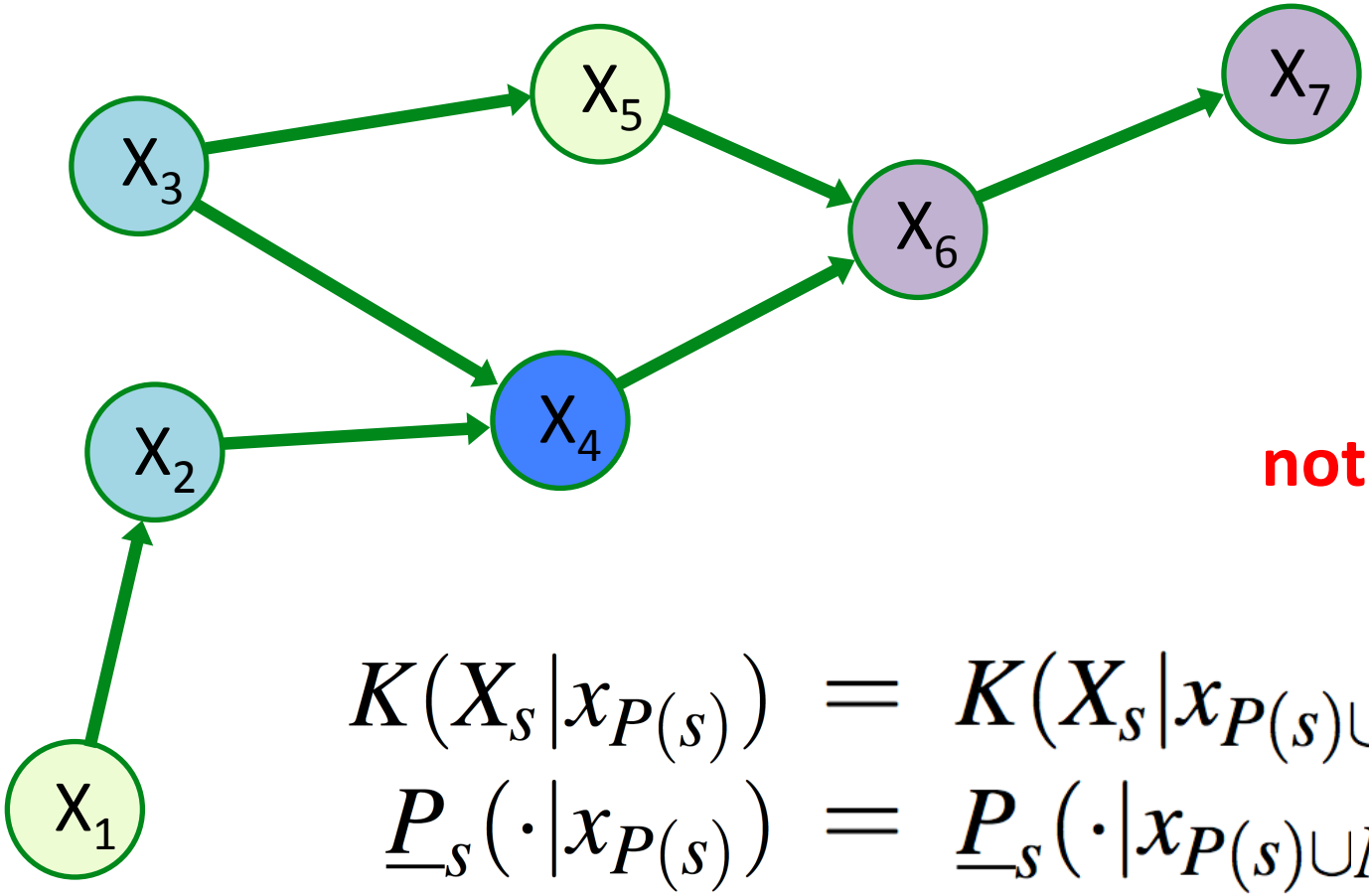
$\underline{P}_G$

$\mathscr{D}_G$

?

$$
\begin{aligned}
K(X_s|x_{P(s)}) &= K(X_s|x_{P(s)\cup N(s)}) \\
\underline{P}_s(\cdot|x_{P(s)}) &= \underline{P}_s(\cdot|x_{P(s)\cup N(s)}) \\
\mathscr{D}_s]x_{P(s)} &= \mathscr{D}_s]x_{P(s)\cup N(s)}
\end{aligned}
$$

Credal networks: a joint model

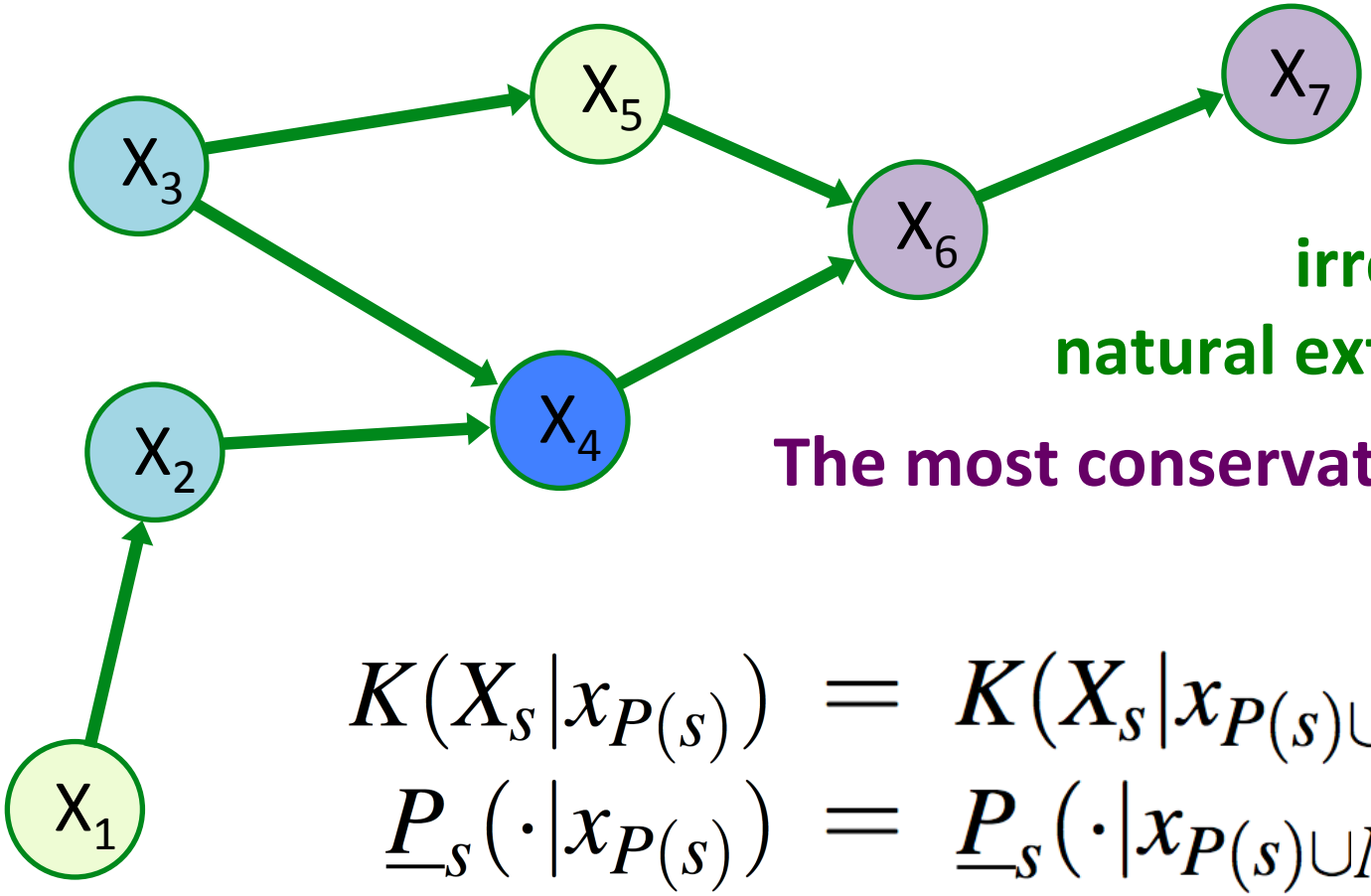


$$K(X_G)$$
$$\underline{P}_G$$
$$\mathscr{D}_G$$

not unique !

$$K(X_s|x_{P(s)}) = K(X_s|x_{P(s)\cup N(s)})$$
$$\underline{P}_s(\cdot|x_{P(s)}) = \underline{P}_s(\cdot|x_{P(s)\cup N(s)})$$
$$\mathscr{D}_s\rfloor x_{P(s)} = \mathscr{D}_s\rfloor x_{P(s)\cup N(s)}$$

Credal networks: a joint model



$K^{\mathrm{irr}}(X_G)$

$\underline{P}_G^{\mathrm{irr}}$

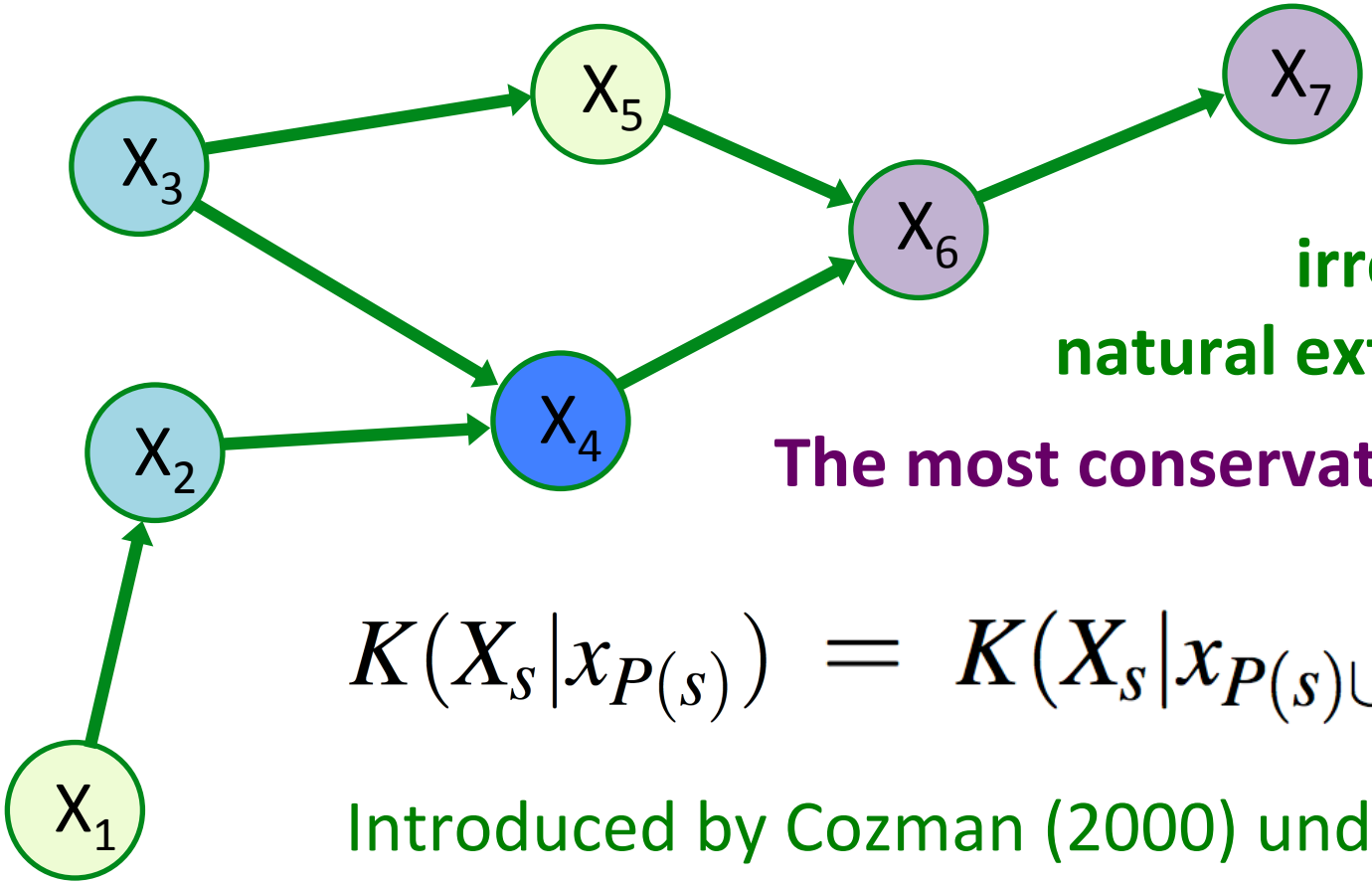
$\mathscr{D}_G^{\mathrm{irr}}$

irrelevant natural extension

The most conservative one !

$$
\begin{aligned}
K(X_s|x_{P(s)}) &= K(X_s|x_{P(s)\cup N(s)}) \\
\underline{P}_s(\cdot|x_{P(s)}) &= \underline{P}_s(\cdot|x_{P(s)\cup N(s)}) \\
\mathscr{D}_s\rfloor x_{P(s)} &= \mathscr{D}_s\rfloor x_{P(s)\cup N(s)}
\end{aligned}
$$

Credal networks using credal sets



$K^{\mathrm{irr}}(X_G)$

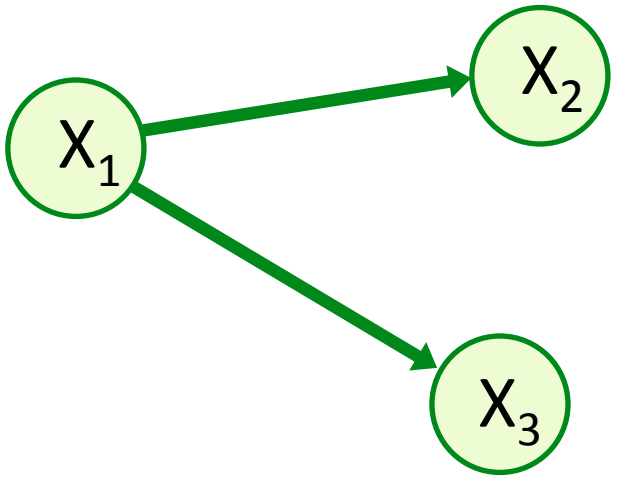
irrelevant
natural extension

The most conservative one !

$$K(X_s|x_{P(s)}) = K(X_s|x_{P(s)\cup N(s)})$$

Introduced by Cozman (2000) under the assumption of positive lower probability.

Credal networks using credal sets



$$K^{\mathrm{irr}}(X_G)$$

irrelevant
natural extension

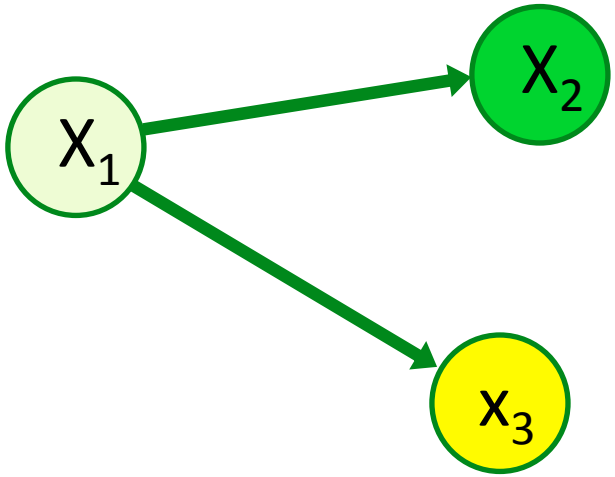
The most conservative one !

$$K(X_s|x_{P(s)}) = K(X_s|x_{P(s)\cup N(s)})$$

Introduced by Cozman (2000) under the assumption of positive lower probability.

Description in terms of linear constraints!

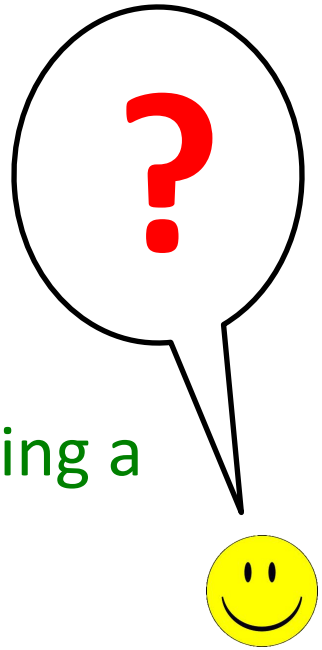
Credal networks using credal sets



$K^{\text{irr}}(X_G)$

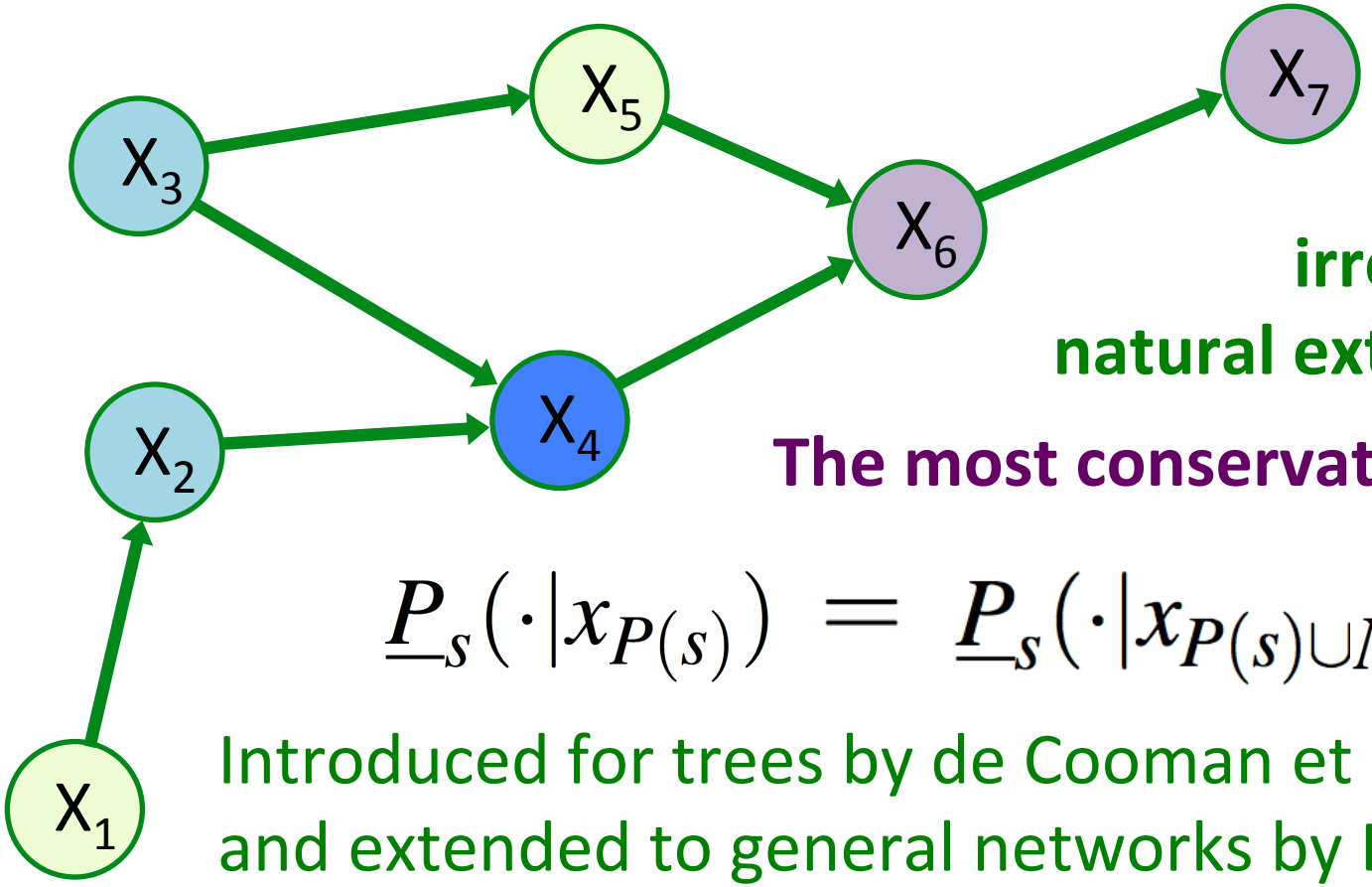
Query nodes

Evidence nodes



- Inference problems can be reduced to solving a (potentialy large) linear program!
- Lots of potential to derive both outer and inner approximations

Credal networks using lower previsions



$\underline{P}_G^{\text{irr}}$

irrelevant natural extension

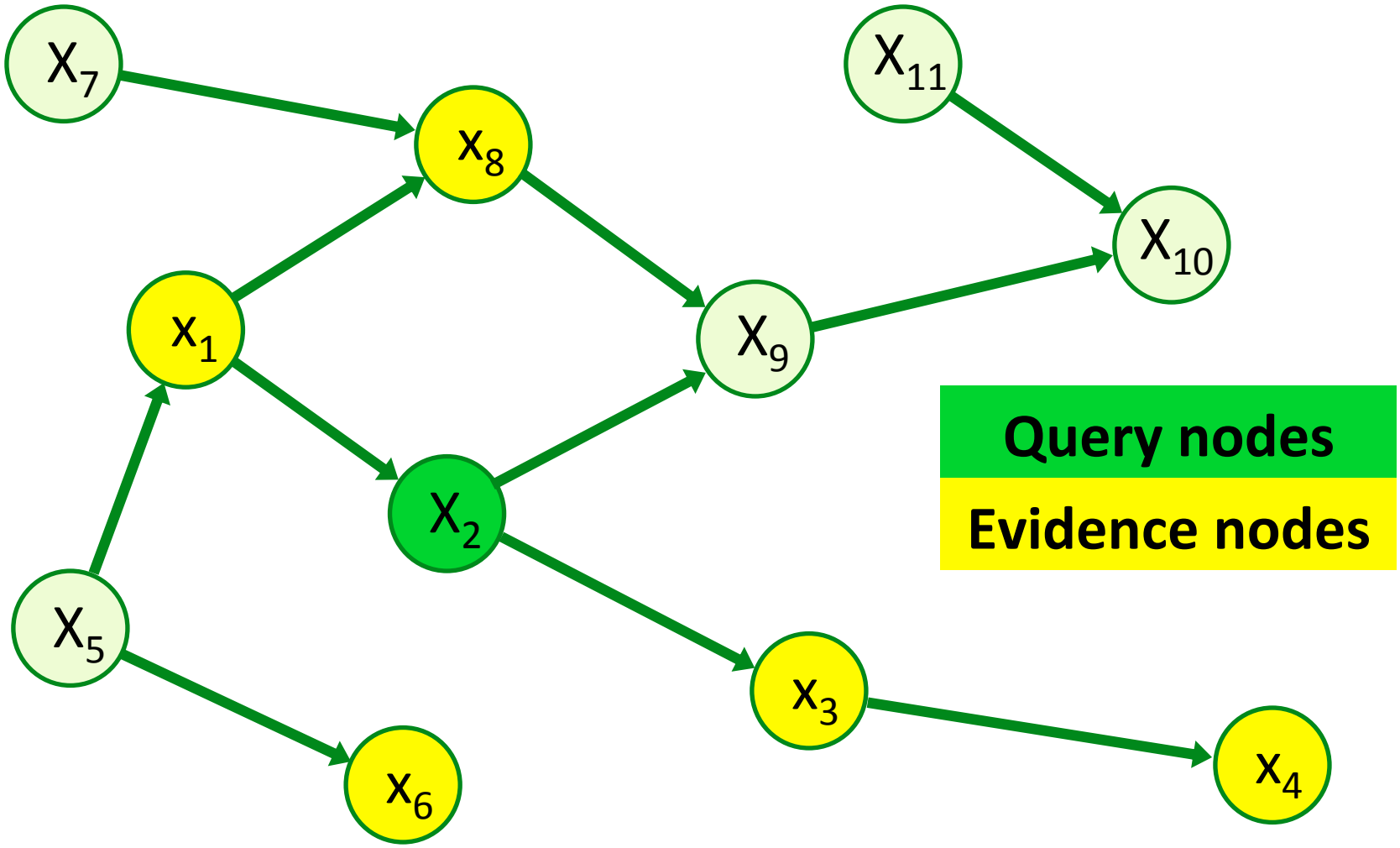
The most conservative one !

$$\underline{P}_s(\cdot|x_{P(s)}) = \underline{P}_s(\cdot|x_{P(s)\cup N(s)})$$

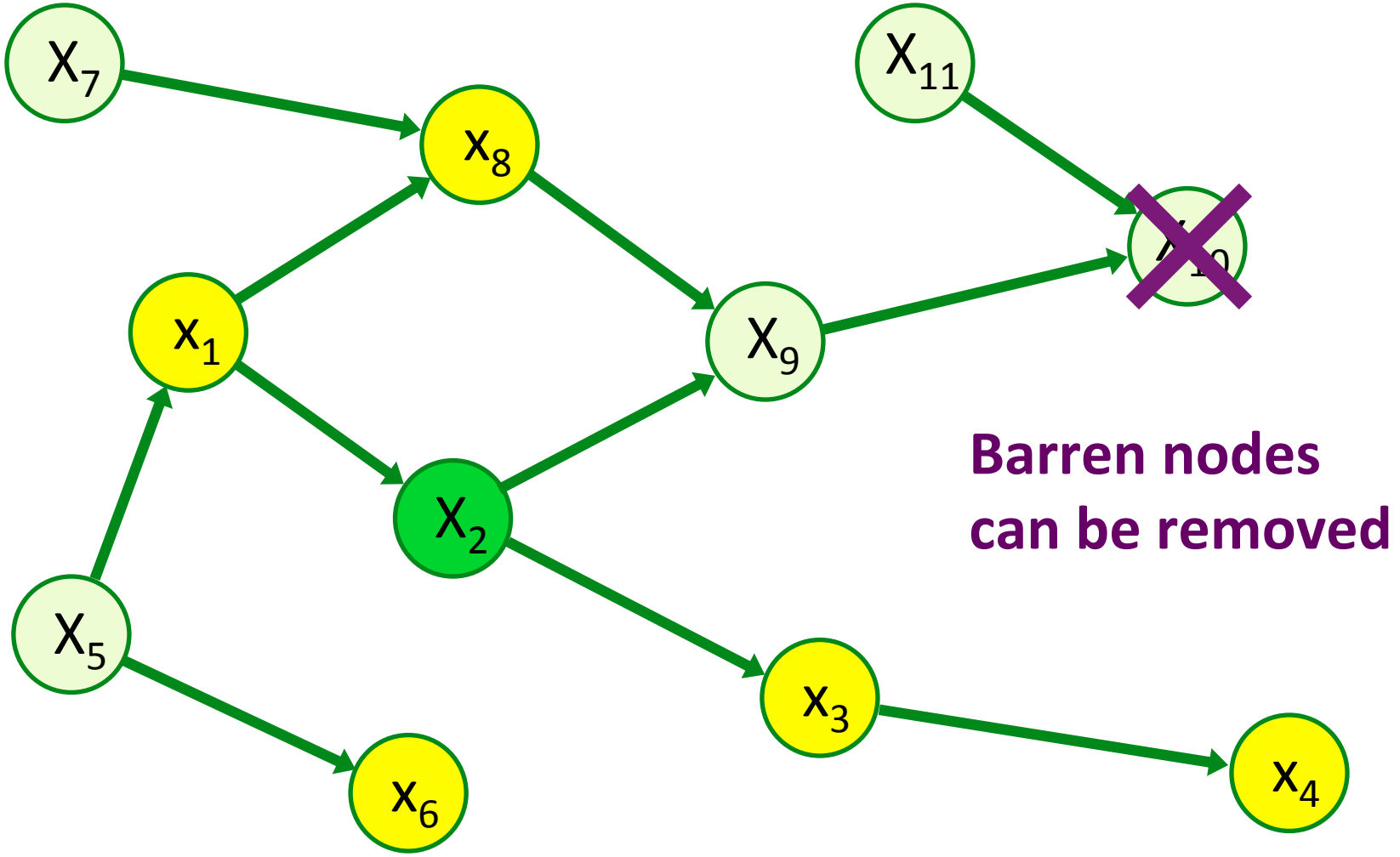
Introduced for trees by de Cooman et al. (2010) and extended to general networks by De Bock & de Cooman (2013), without positivity assumptions.

The joint is still described by the same linear constraints!

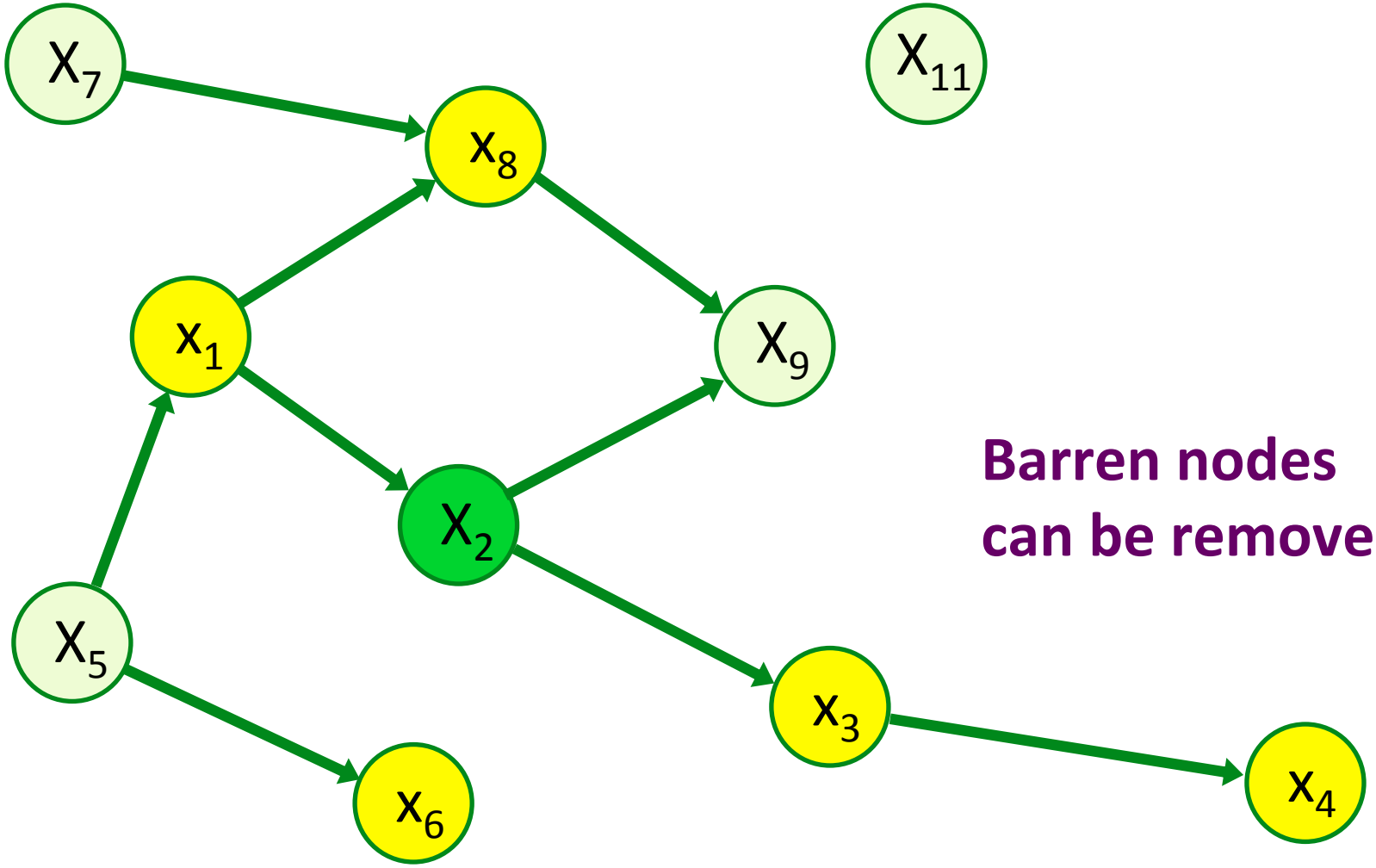
Bayesian networks: useful properties



Bayesian networks: useful properties

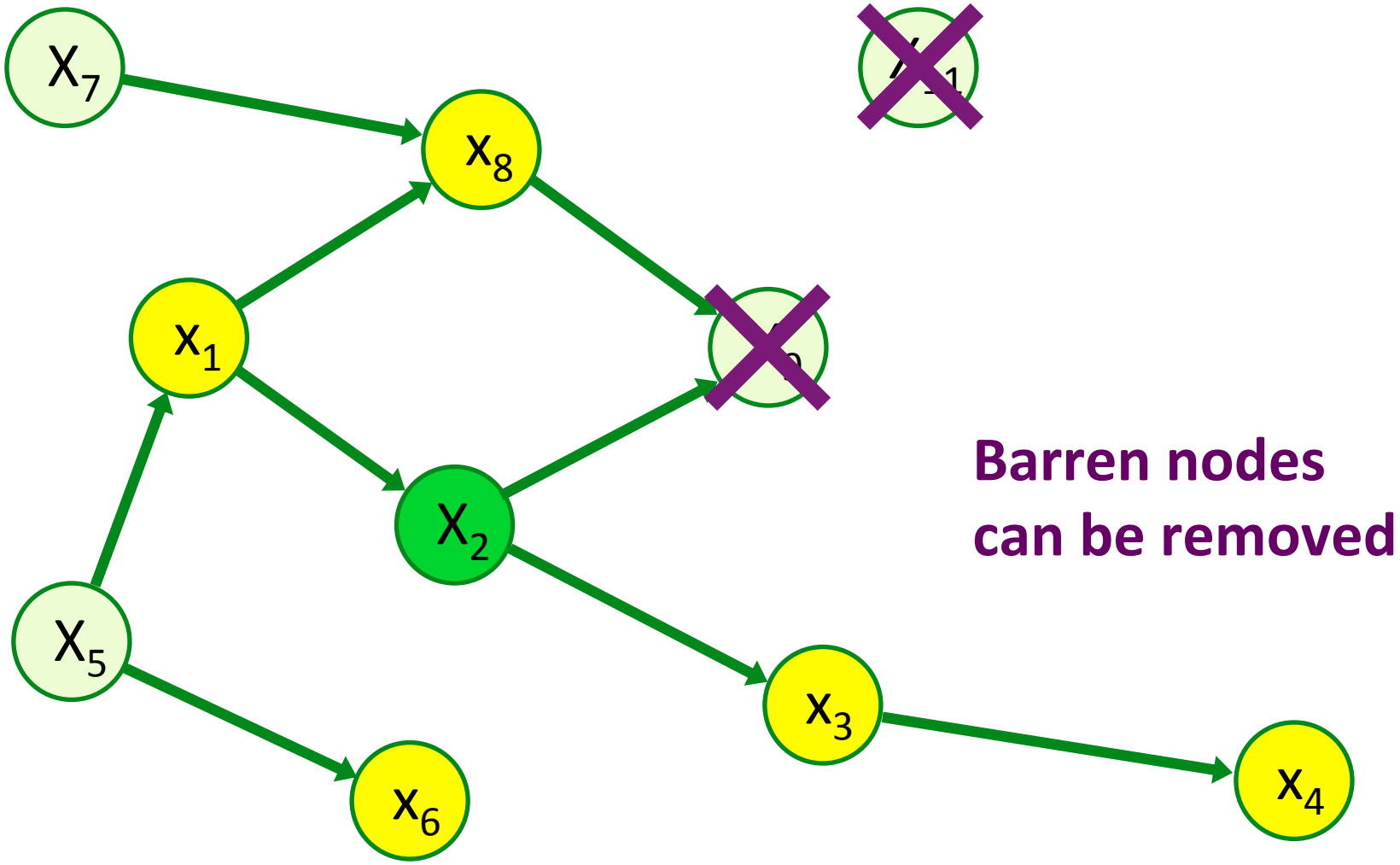


Bayesian networks: useful properties

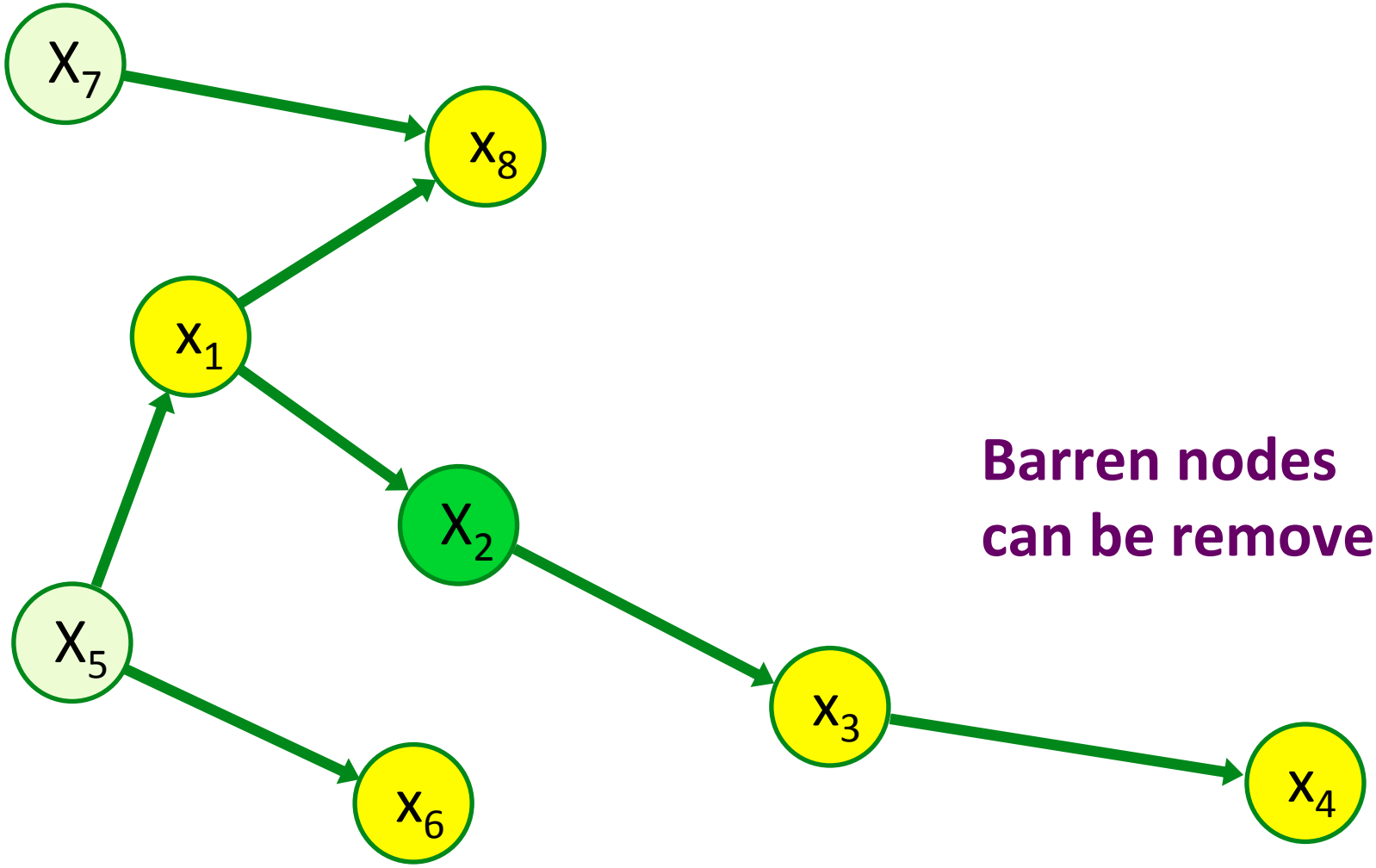


Barren nodes can be removed

Bayesian networks: useful properties

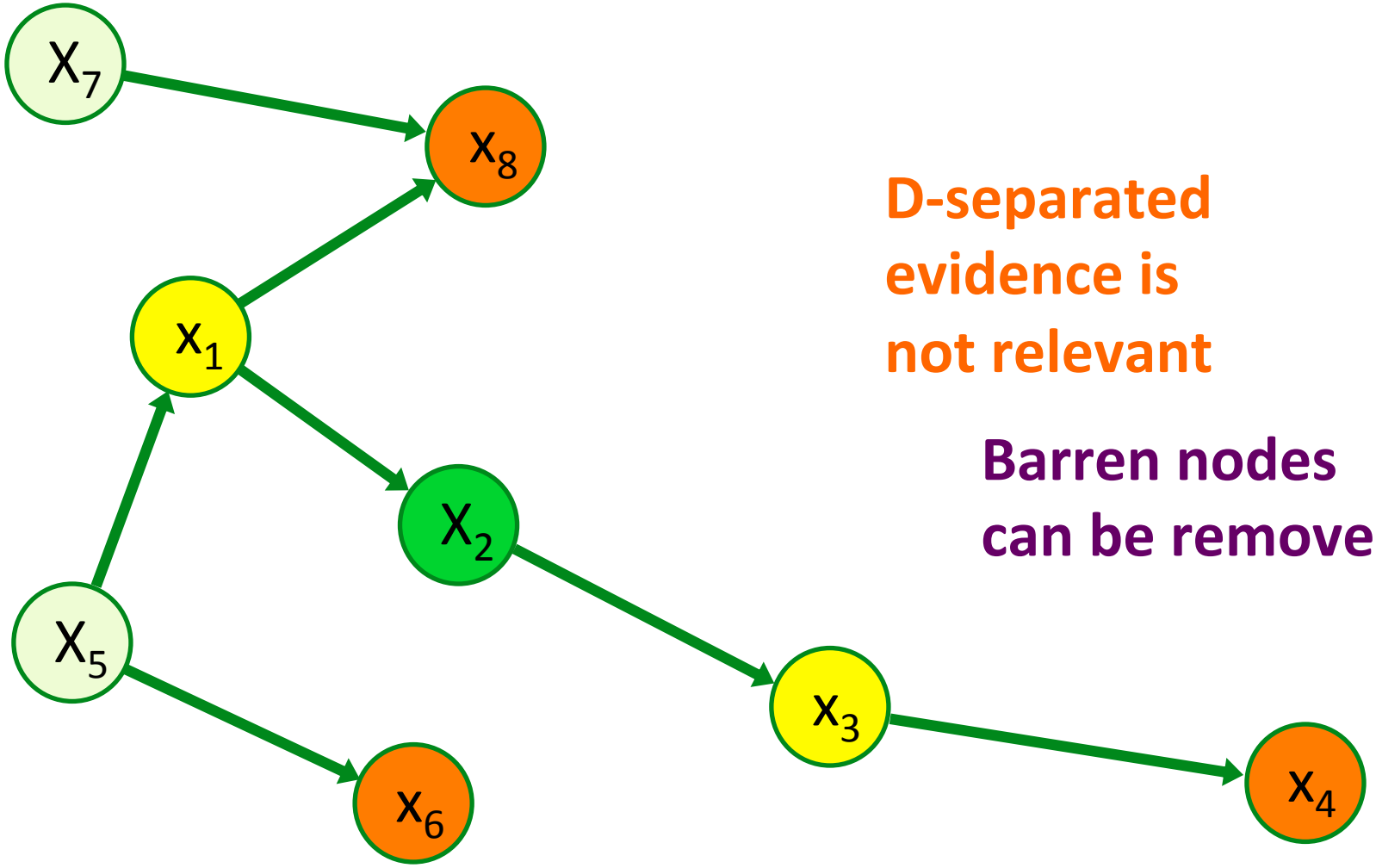


Bayesian networks: useful properties



Barren nodes can be removed

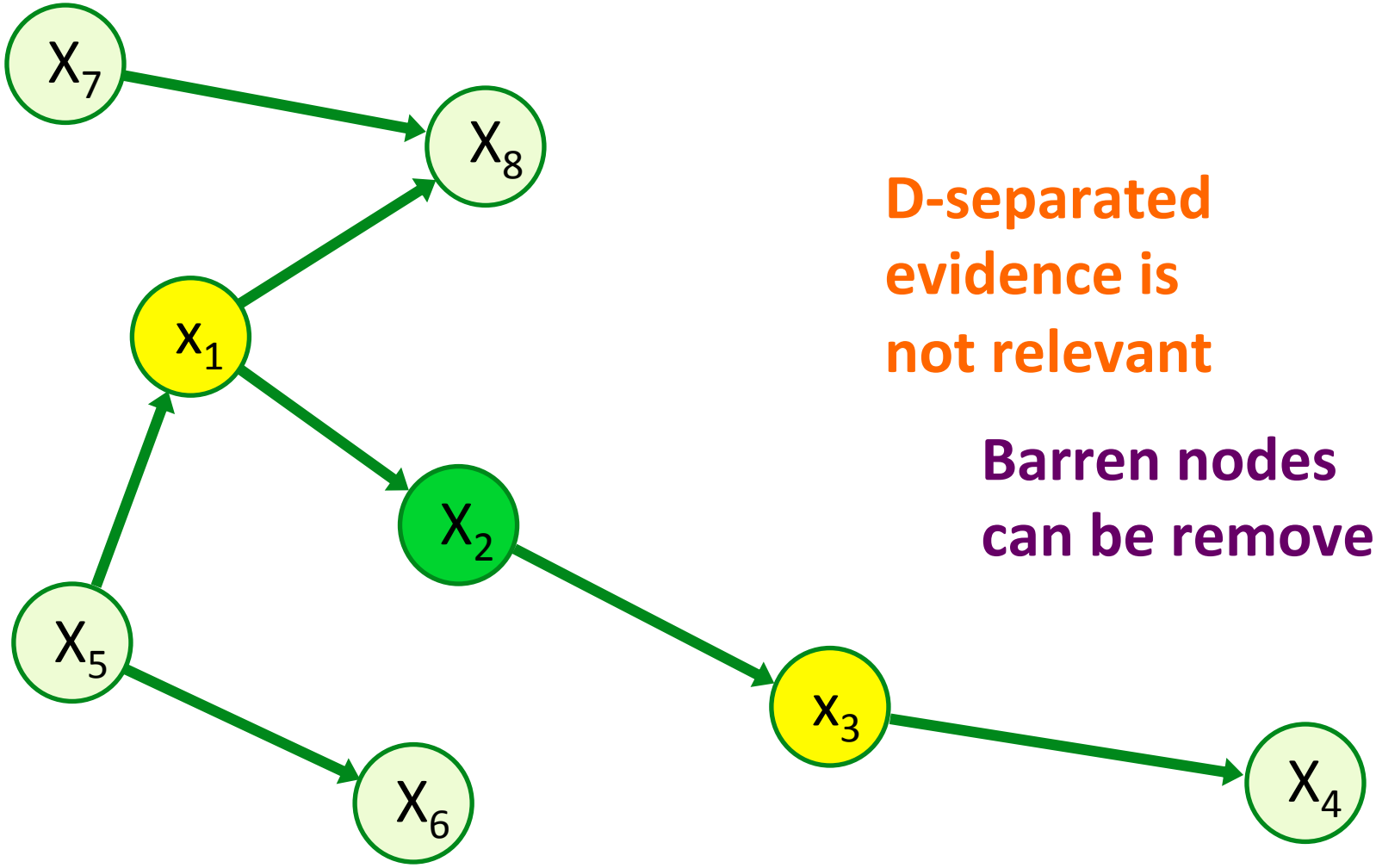
Bayesian networks: useful properties



D-separated evidence is not relevant

Barren nodes can be removed

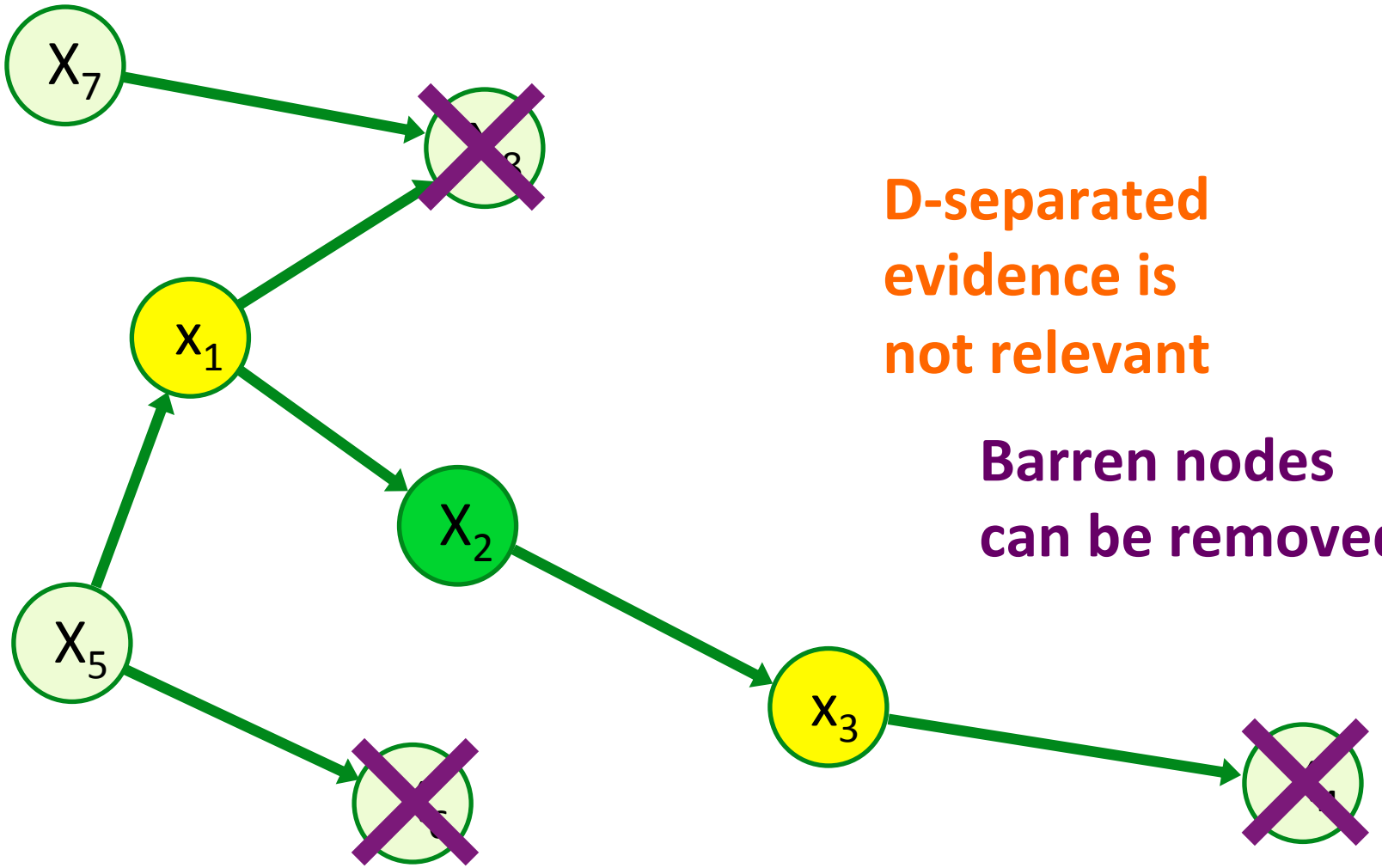
Bayesian networks: useful properties



D-separated evidence is not relevant

Barren nodes can be removed

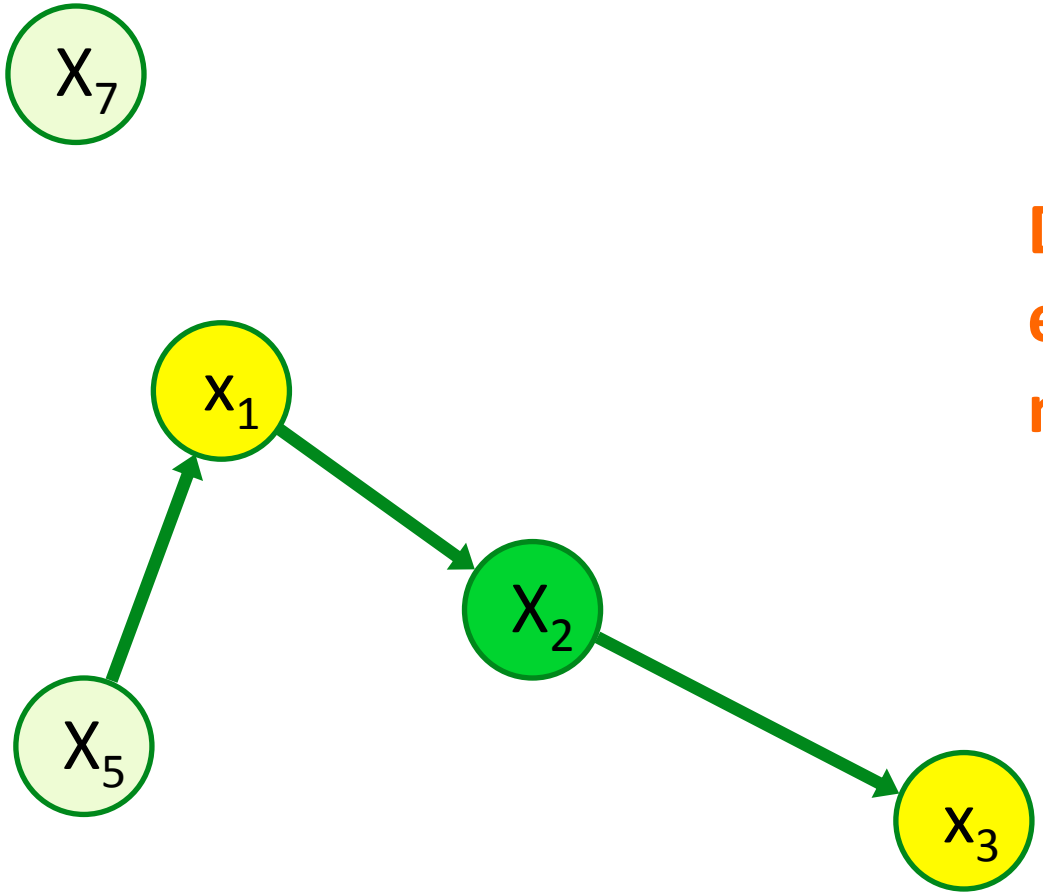
Bayesian networks: useful properties



D-separated evidence is not relevant

Barren nodes can be removed

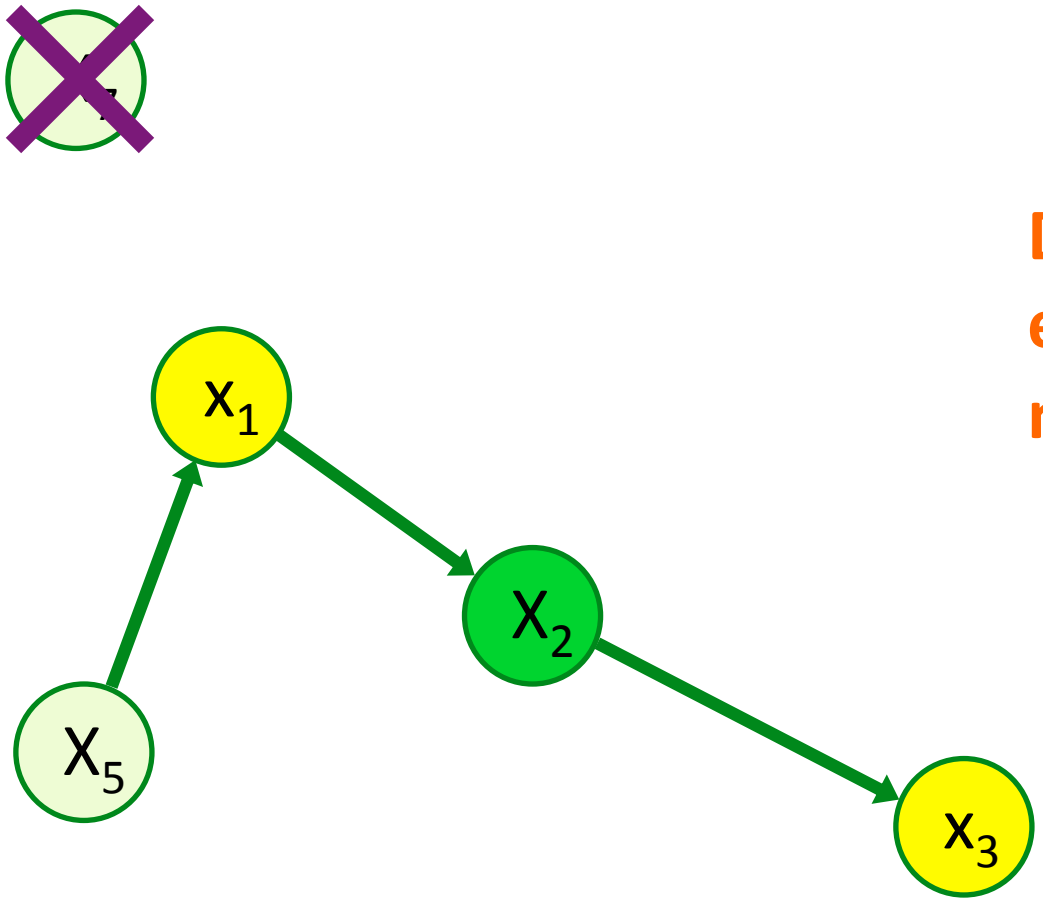
Bayesian networks: useful properties



D-separated evidence is not relevant

Barren nodes can be removed

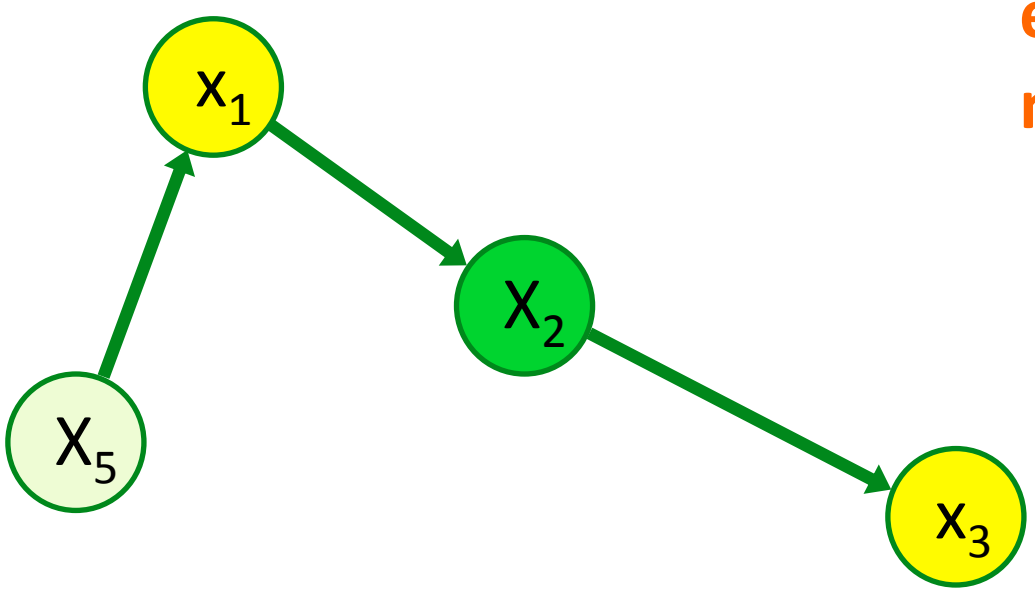
Bayesian networks: useful properties



D-separated evidence is not relevant

Barren nodes can be removed

Bayesian networks: useful properties



D-separated evidence is not relevant

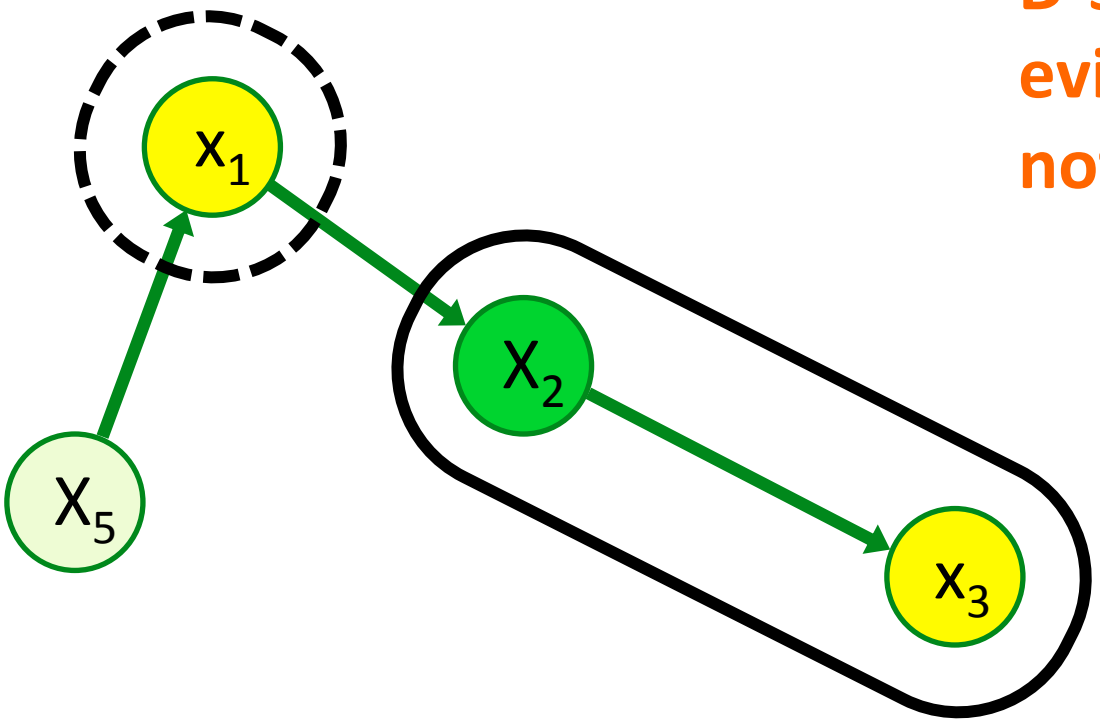
Barren nodes can be removed

Bayesian networks: useful properties

Conditional marginalisation properties

D-separated evidence is not relevant

Barren nodes can be removed

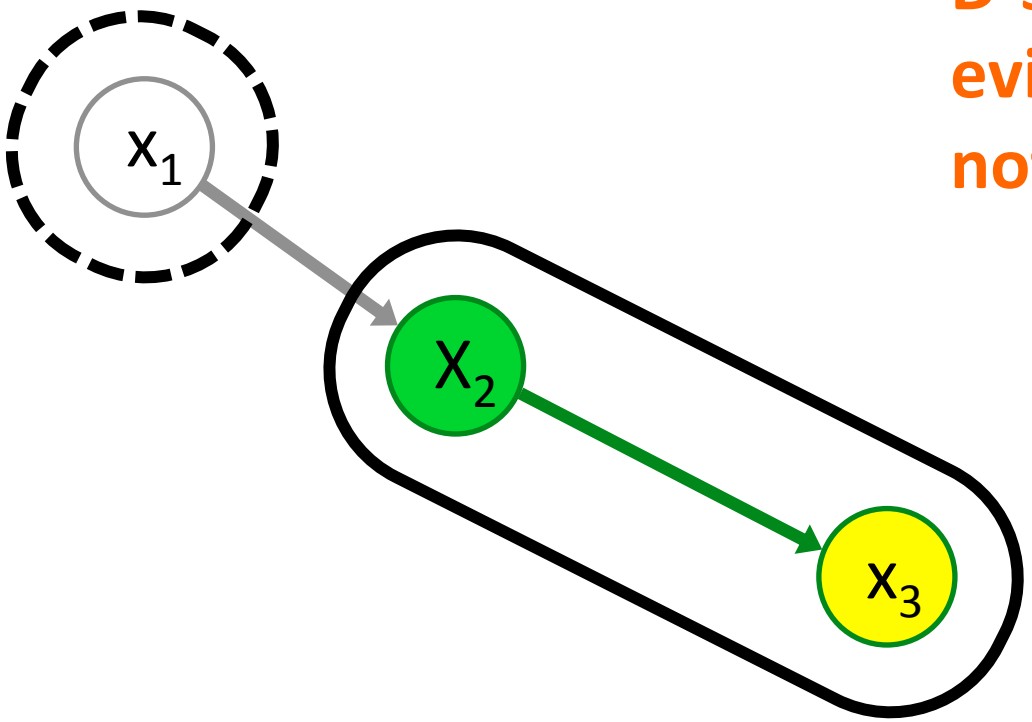


Bayesian networks: useful properties

Conditional marginalisation properties

D-separated evidence is not relevant

Barren nodes can be removed

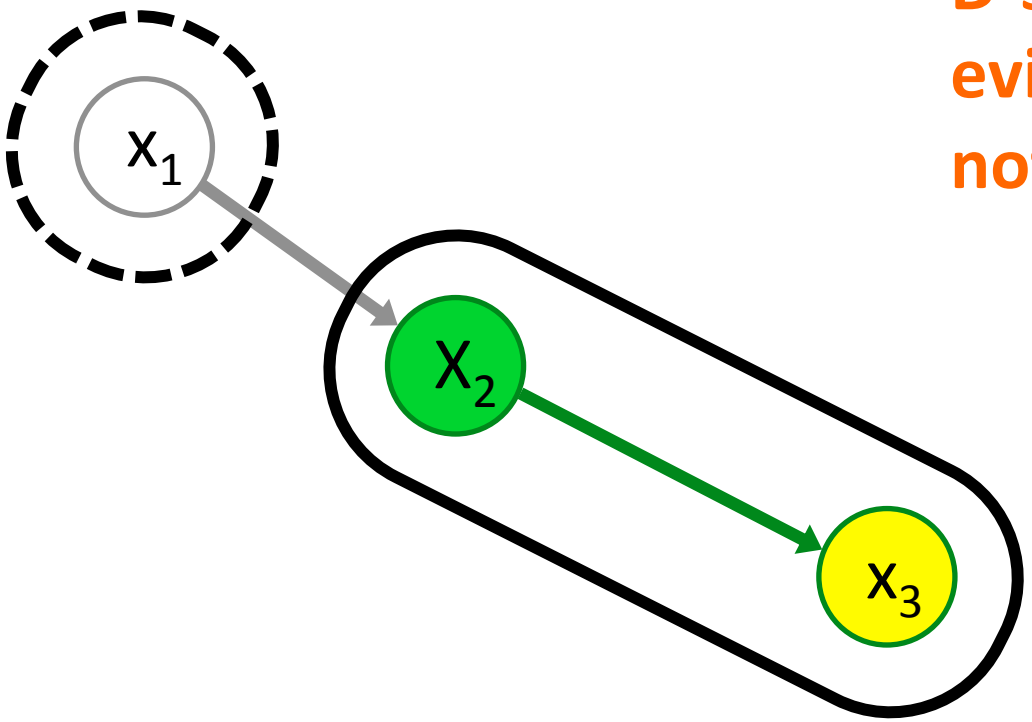


Credal networks: useful properties ?

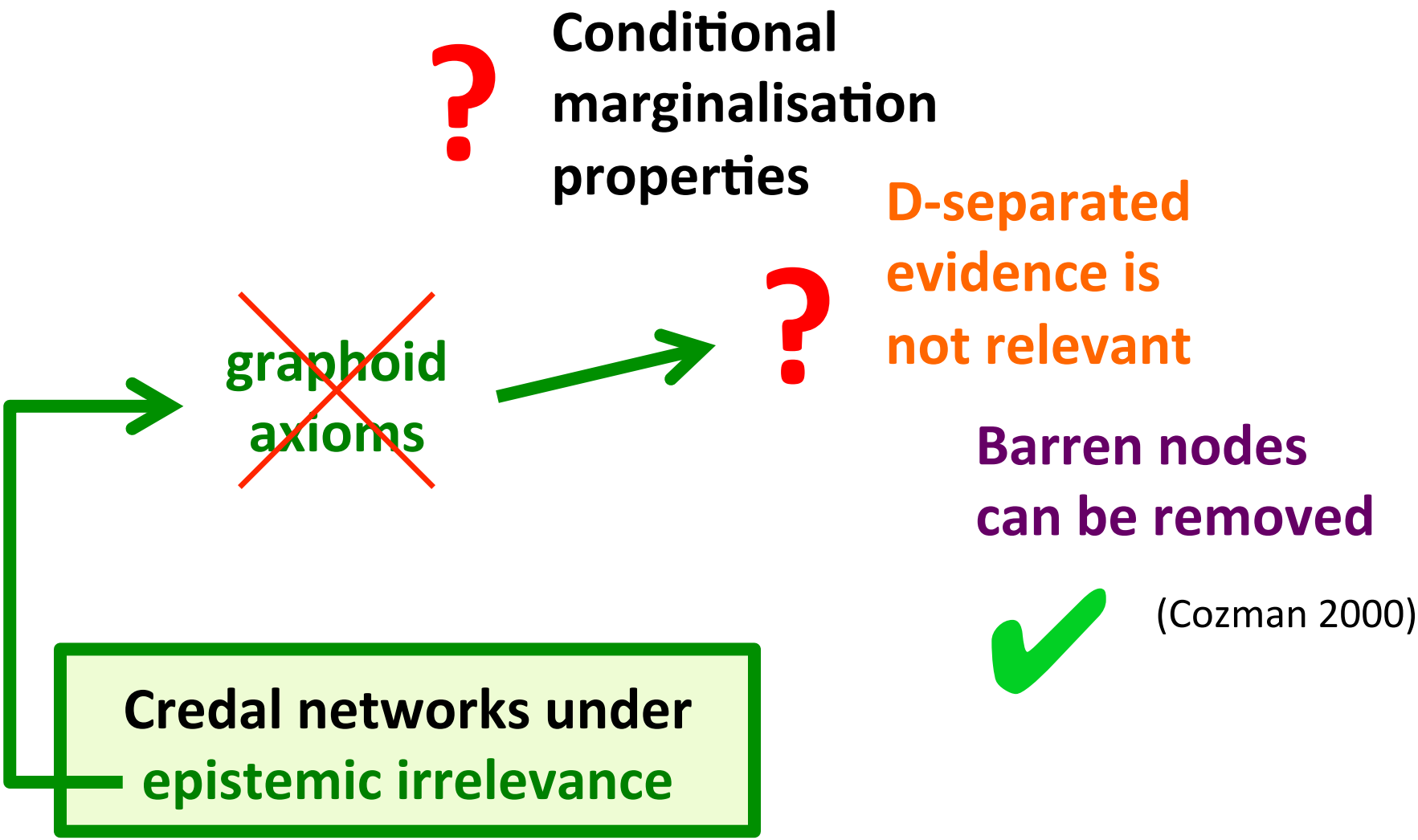
Conditional marginalisation properties

D-separated evidence is not relevant

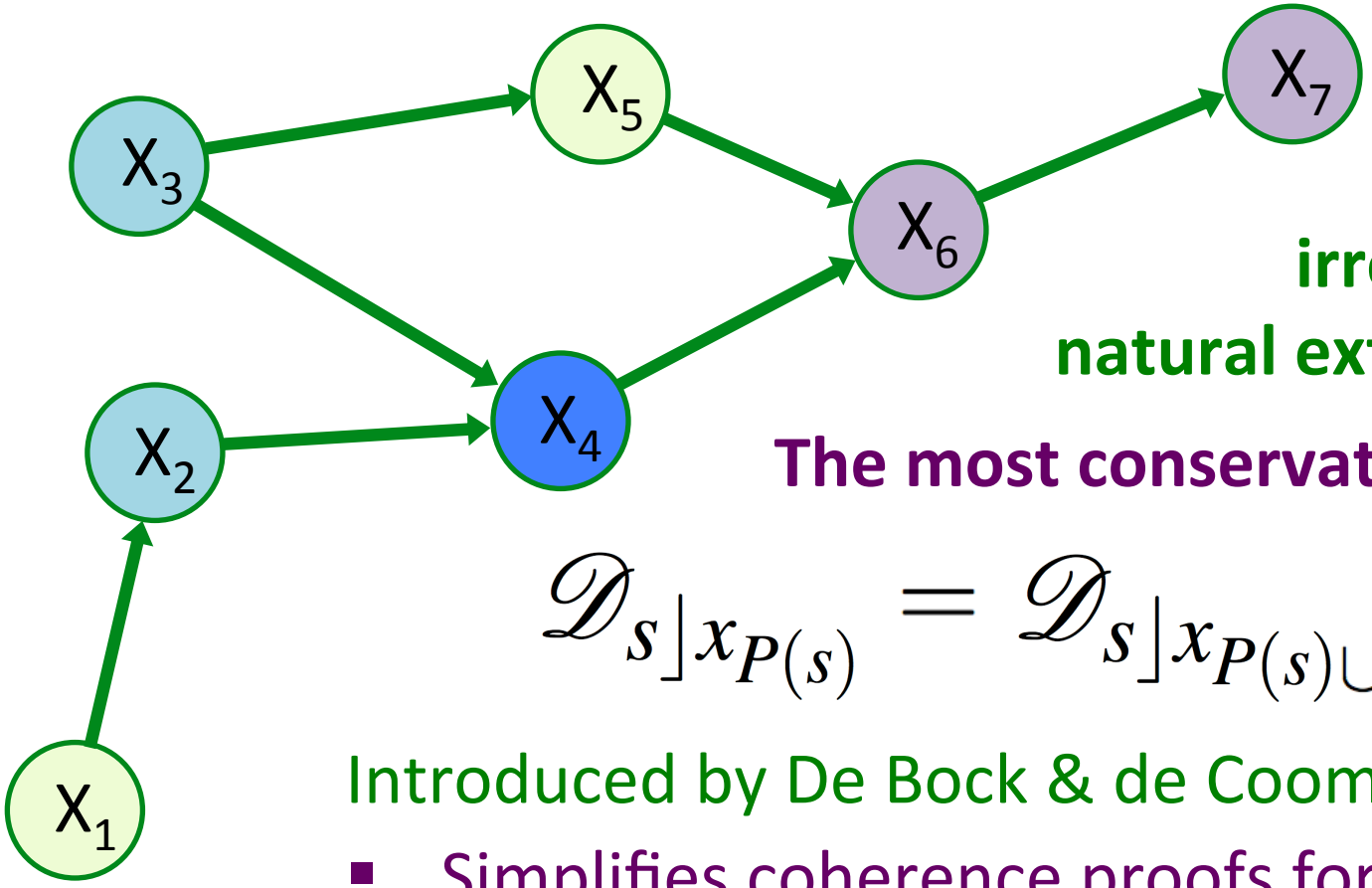
Barren nodes can be removed



Credal networks: useful properties ?



Credal networks using SDGs



$\mathscr{D}_G^{\mathrm{irr}}$

irrelevant natural extension

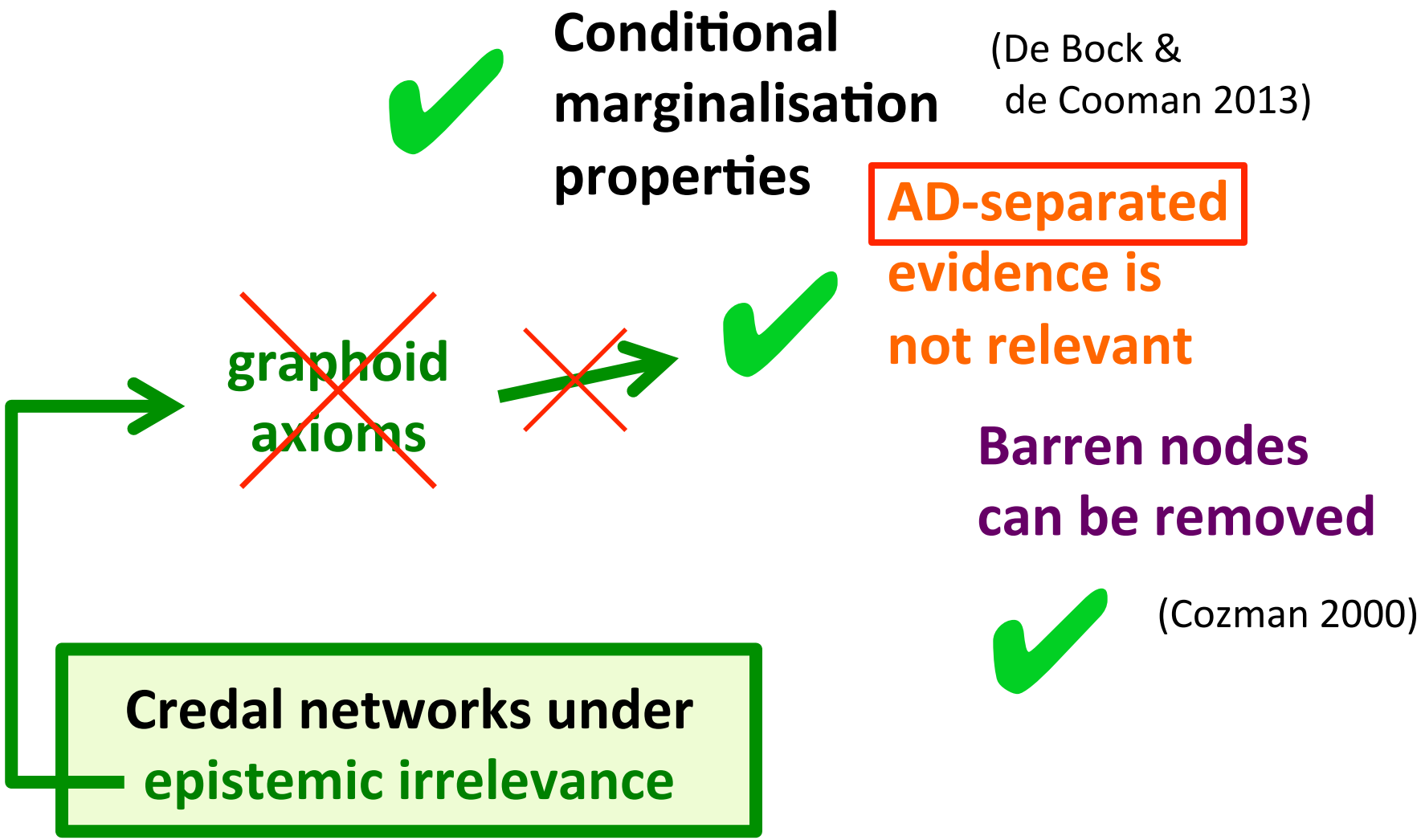
The most conservative one !

$$\mathscr{D}_{s\rfloor x_{P(s)}} = \mathscr{D}_{s\rfloor x_{P(s)\cup N(s)}}$$

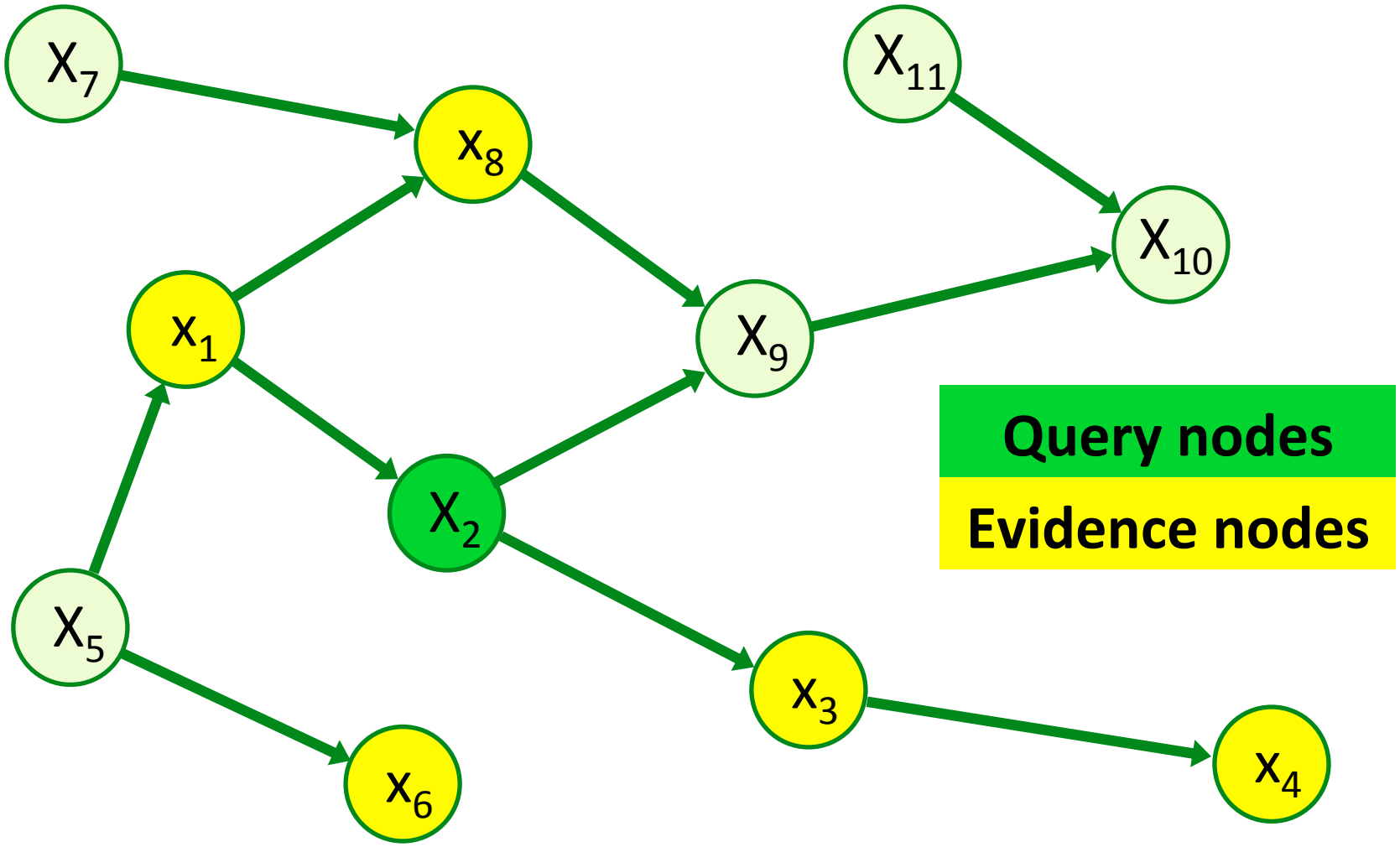
Introduced by De Bock & de Cooman (2013)

- Simplifies coherence proofs for LPs
- (conditional) marginalisation properties
- AD-separation implies irrelevance

Credal networks: useful properties ✔



Credal networks: useful properties ✔

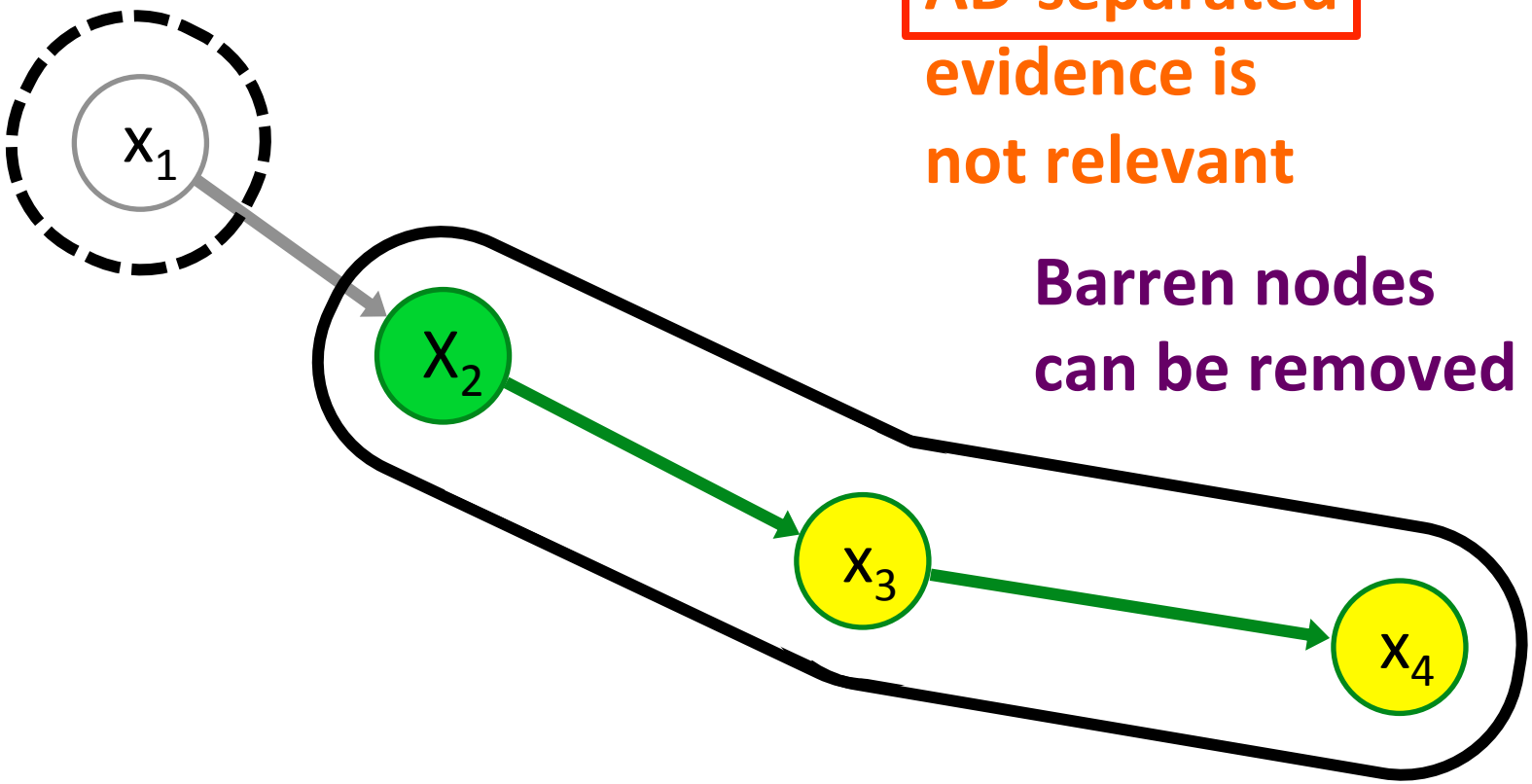


Credal networks: useful properties ✔

Conditional marginalisation properties

AD-separated evidence is not relevant

Barren nodes can be removed

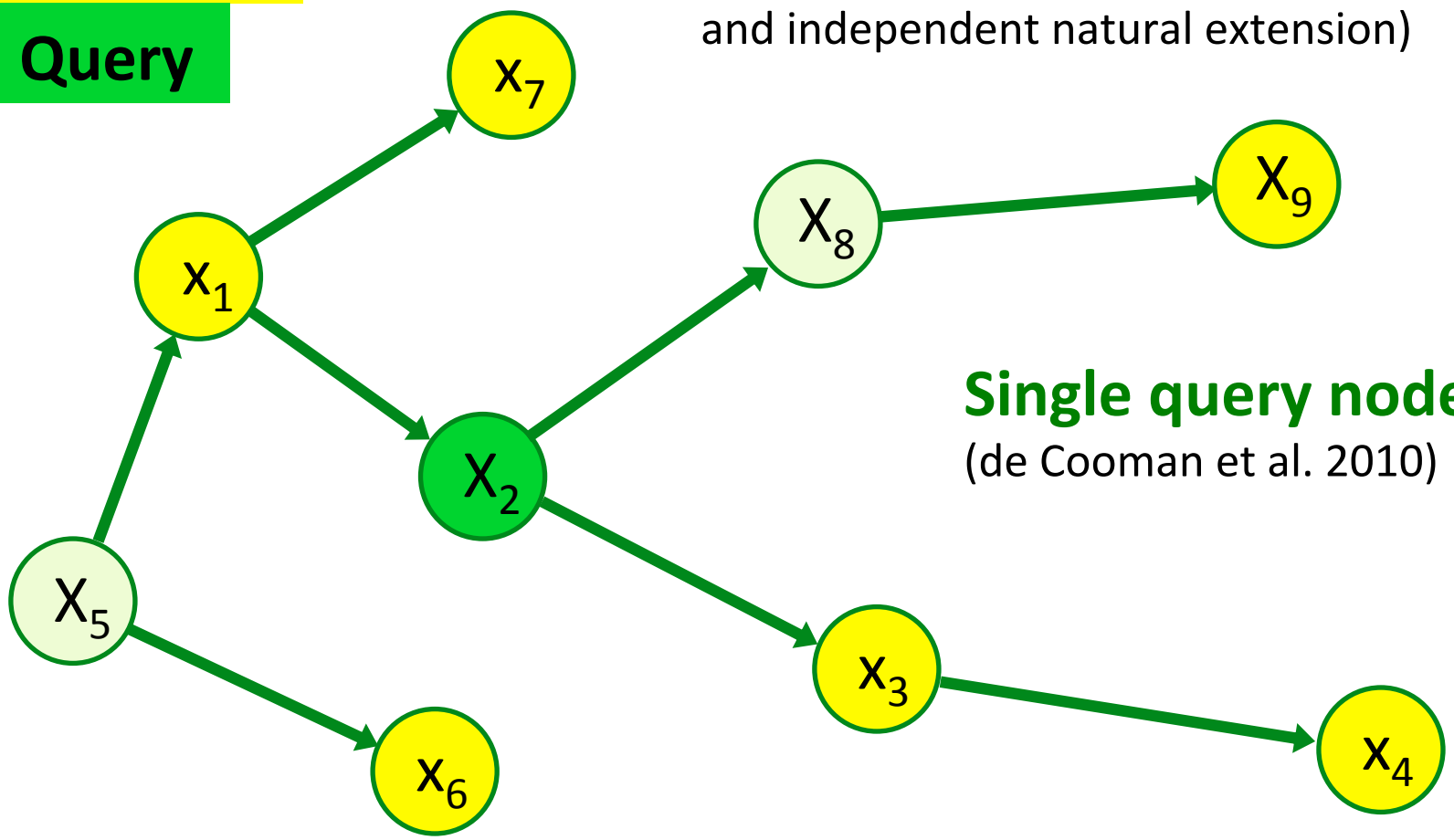


Credal trees: efficient (linear!) algorithms

Evidence

Query

(In trees, the joint can be constructed recursively by applying marginal extension and independent natural extension)



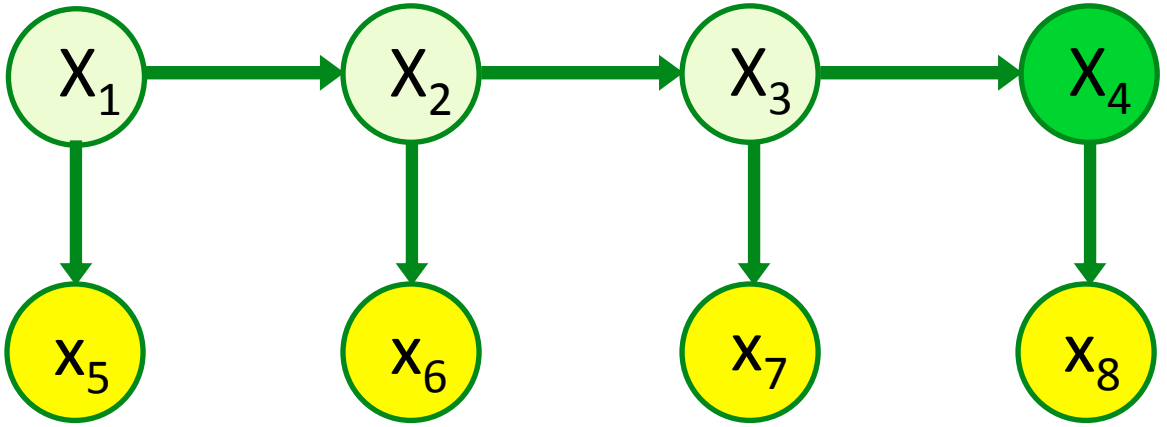
Single query node

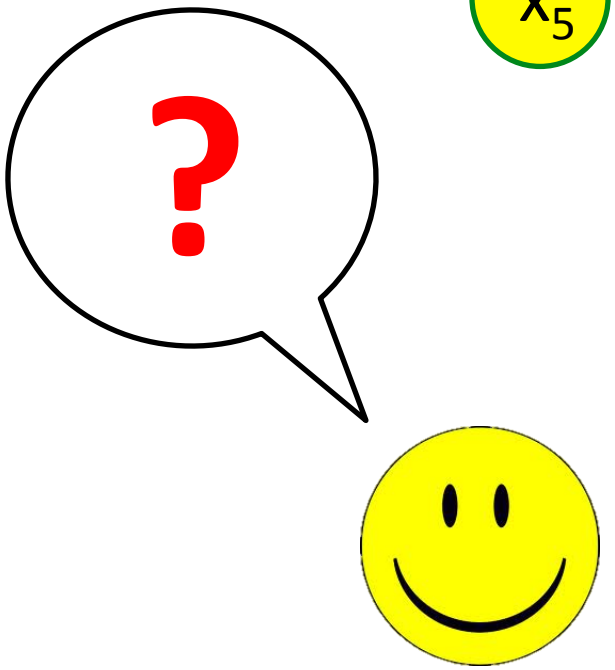
(de Cooman et al. 2010)

Credal trees: efficient (linear!) algorithms

Evidence

Query





Single query node
(de Cooman et al. 2010)

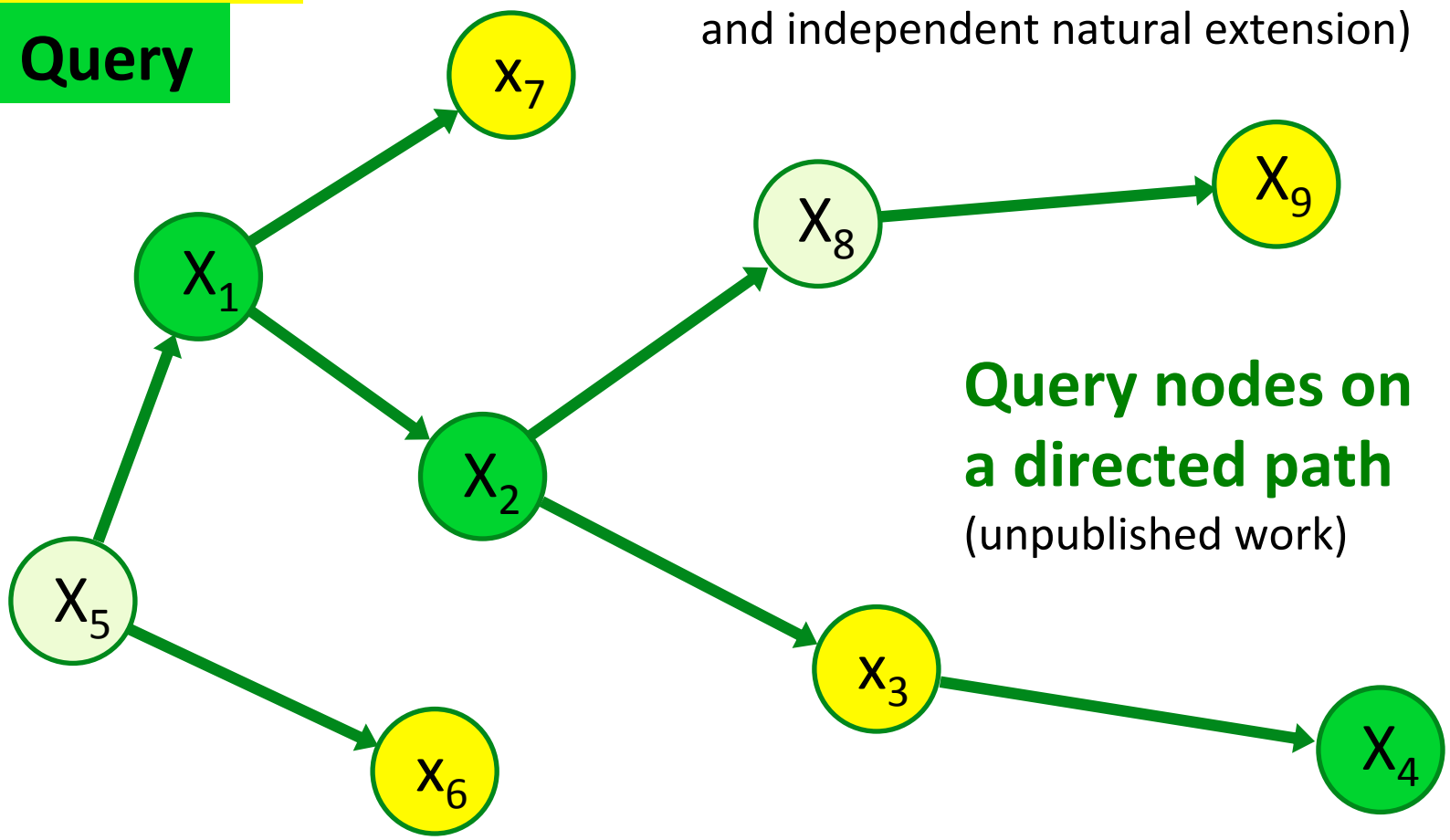
Example:
kalman filtering
(Benavoli et al. 2011)

Credal trees: efficient (linear!) algorithms

Evidence

Query

(In trees, the joint can be constructed recursively by applying marginal extension and independent natural extension)

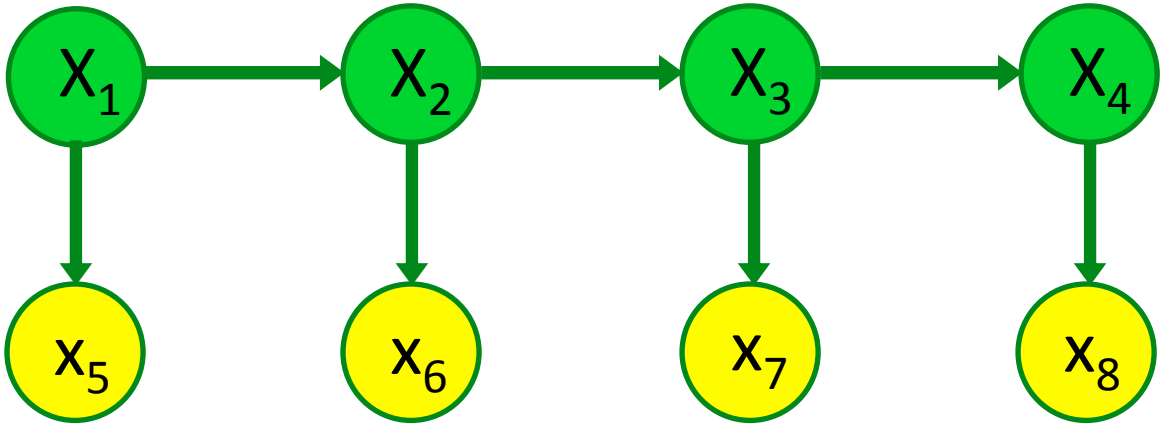


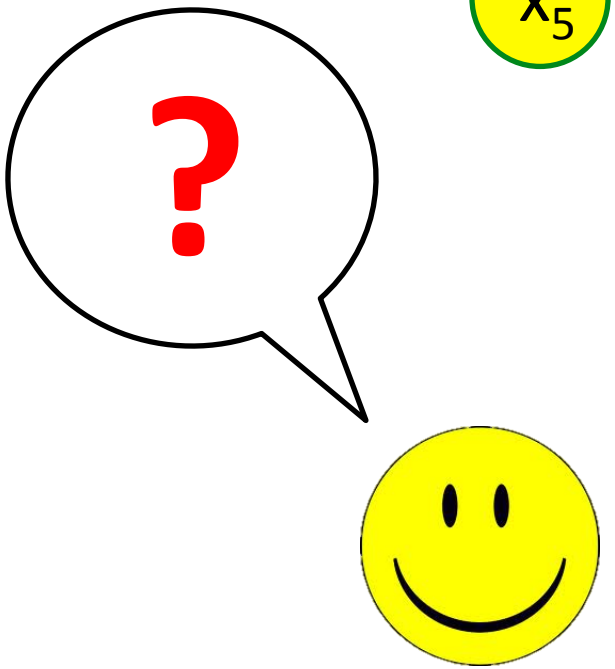
Query nodes on a directed path
(unpublished work)

Credal trees: efficient (linear!) algorithms

Evidence

Query





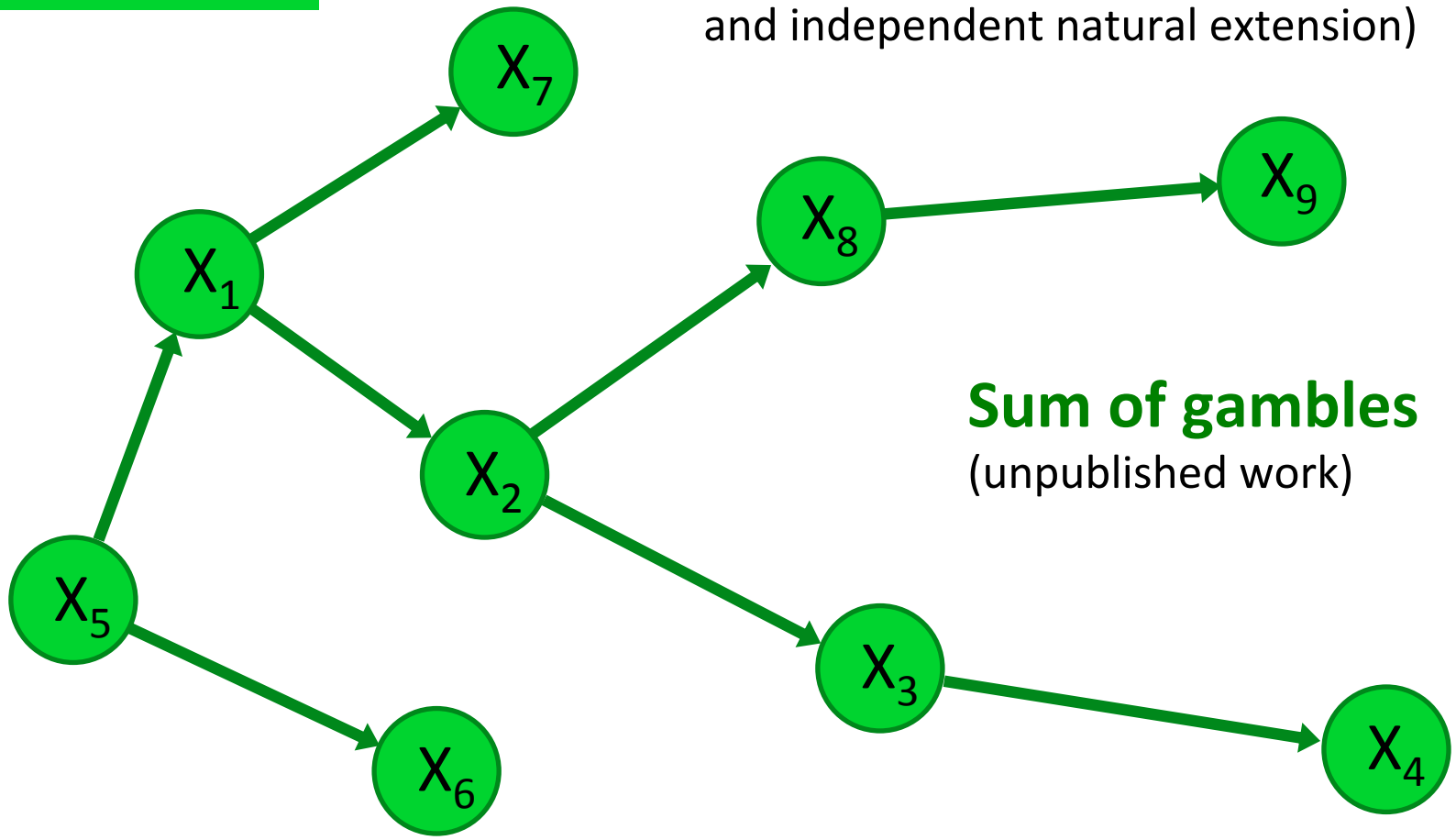
Query nodes on a directed path
(unpublished work)

Example: finding maximin solution in imprecise Viterbi

Credal trees: efficient (linear!) algorithms

Gambles

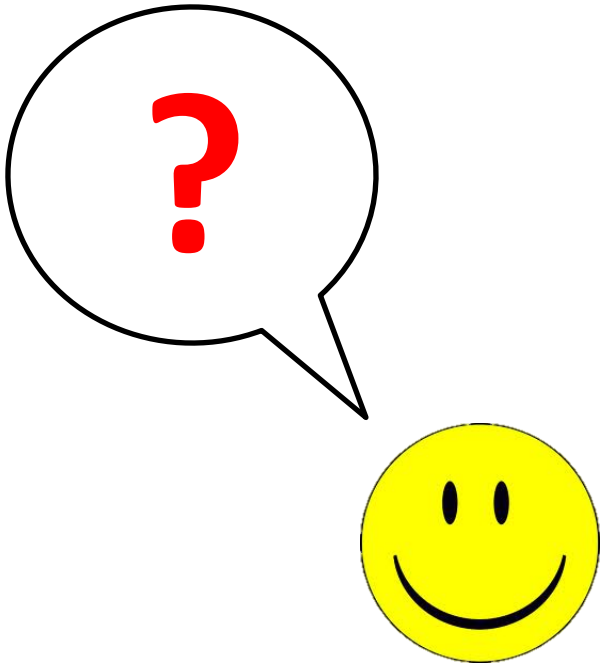
(In trees, the joint can be constructed recursively by applying marginal extension and independent natural extension)



Sum of gambles
(unpublished work)

Credal trees: efficient (linear!) algorithms

Gambles



Sum of gambles
(unpublished work)

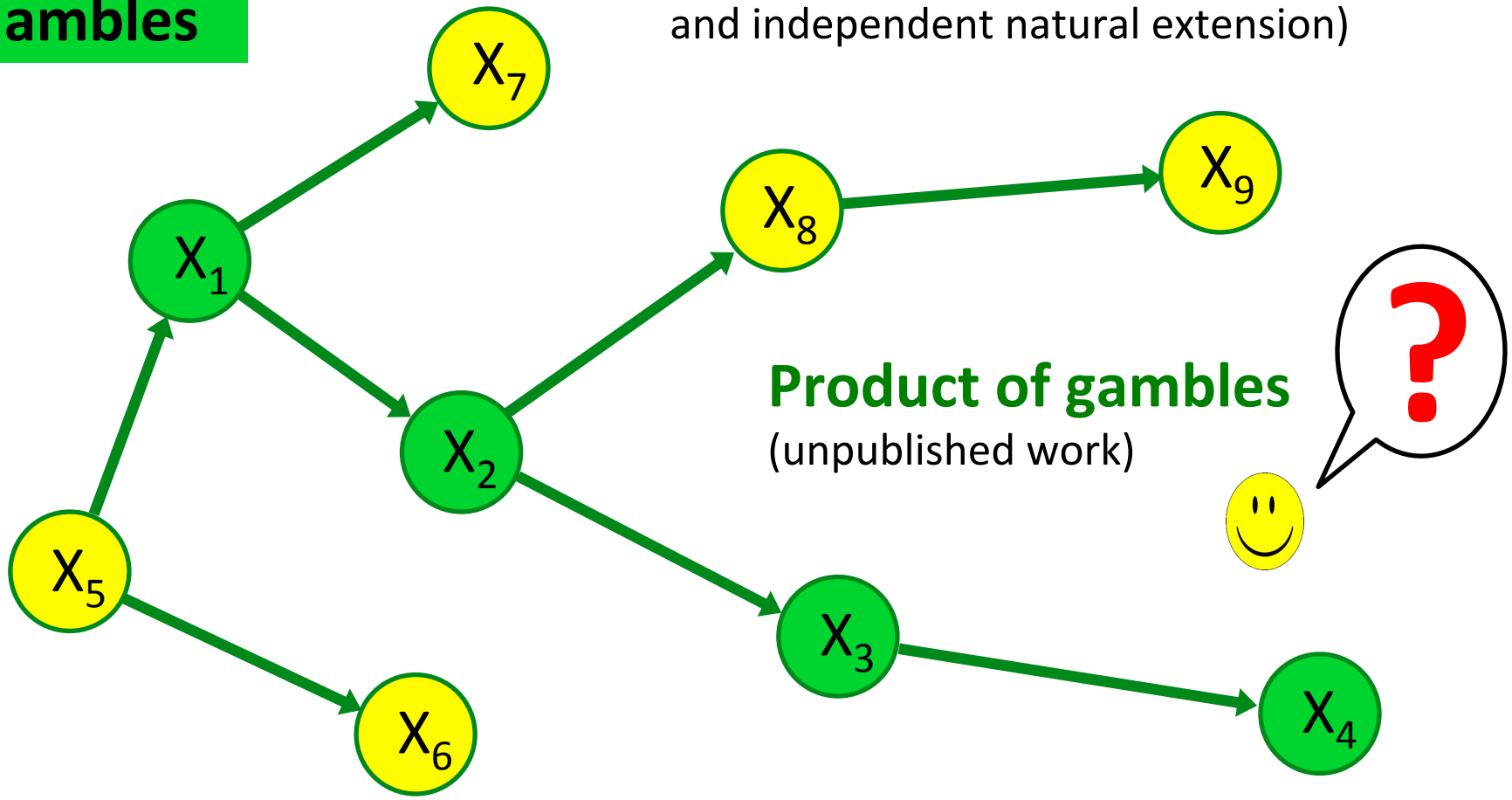
Example: time averages in queueing systems

Credal trees: efficient (linear!) algorithms

Non-negative gambles

Gambles

(In trees, the joint can be constructed recursively by applying marginal extension and independent natural extension)



Product of gambles
(unpublished work)