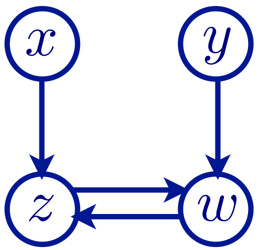
Path Constraints for Causal Discovery

Frederick Eberhardt

This is joint work with Antti Hyttinen, Patrik Hoyer & Matti Järvisalo

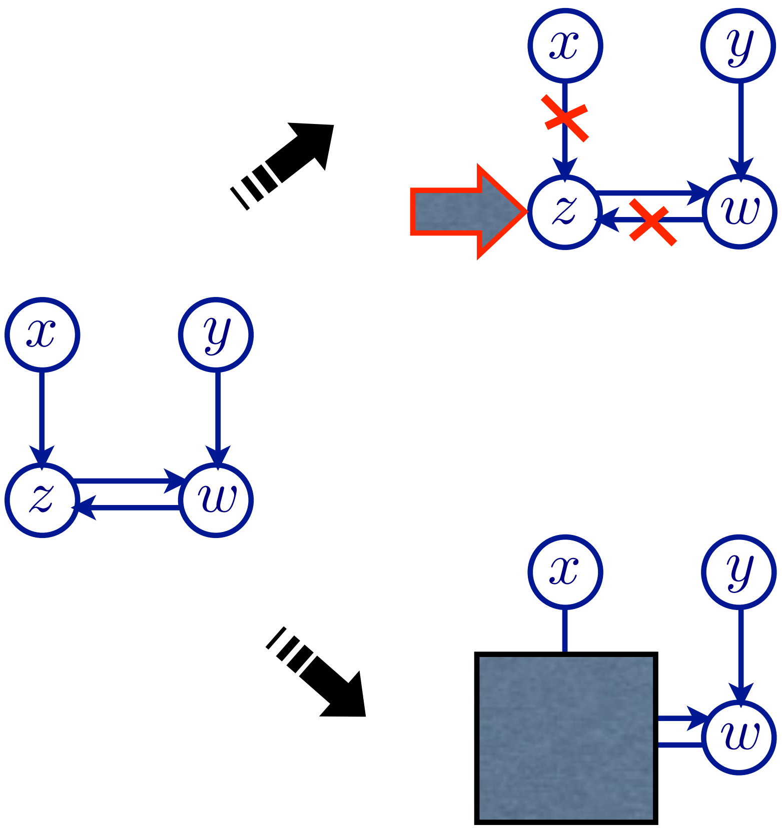
Causal Structure Search

true
model



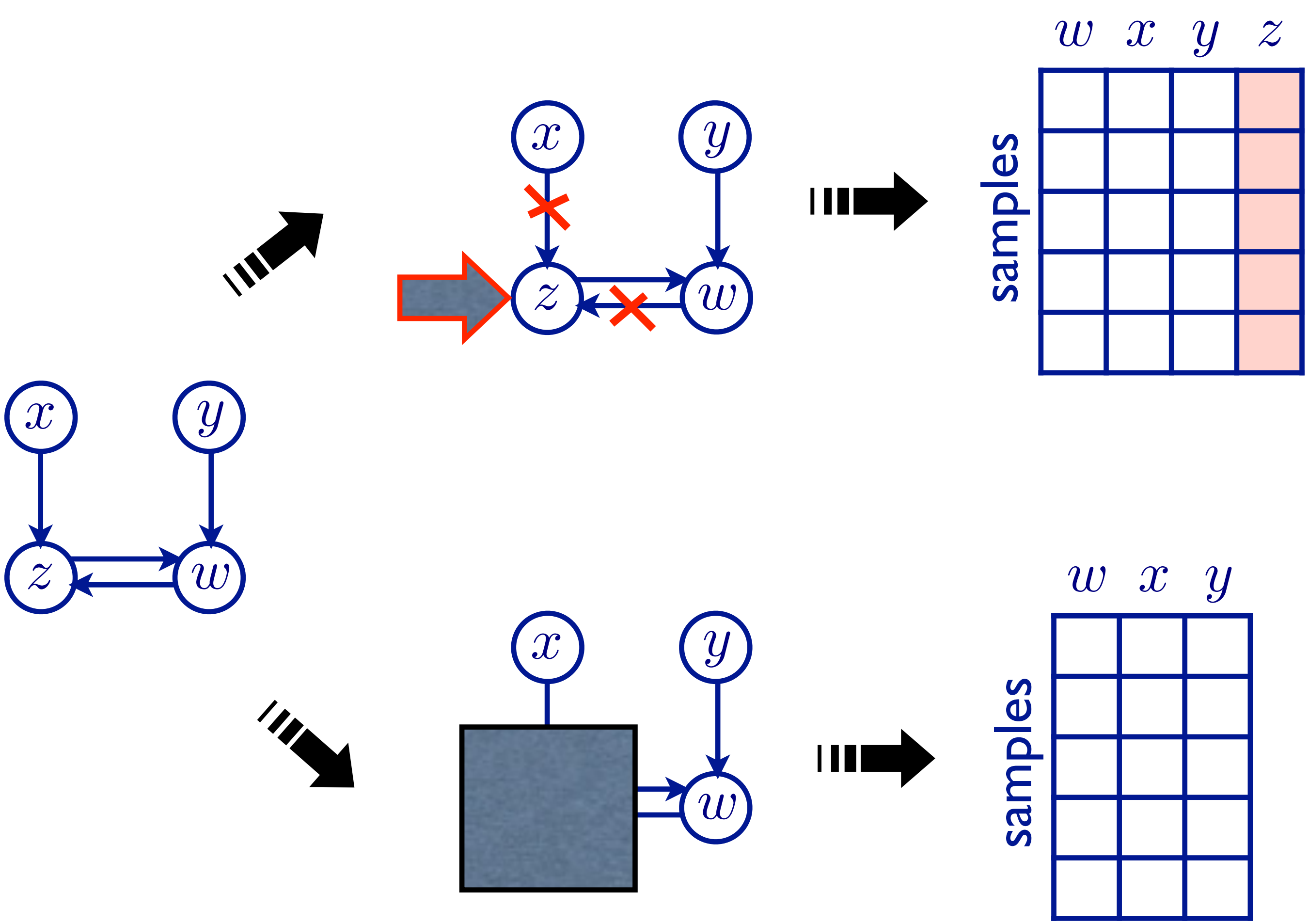
Causal Structure Search

true model → experimental conditions

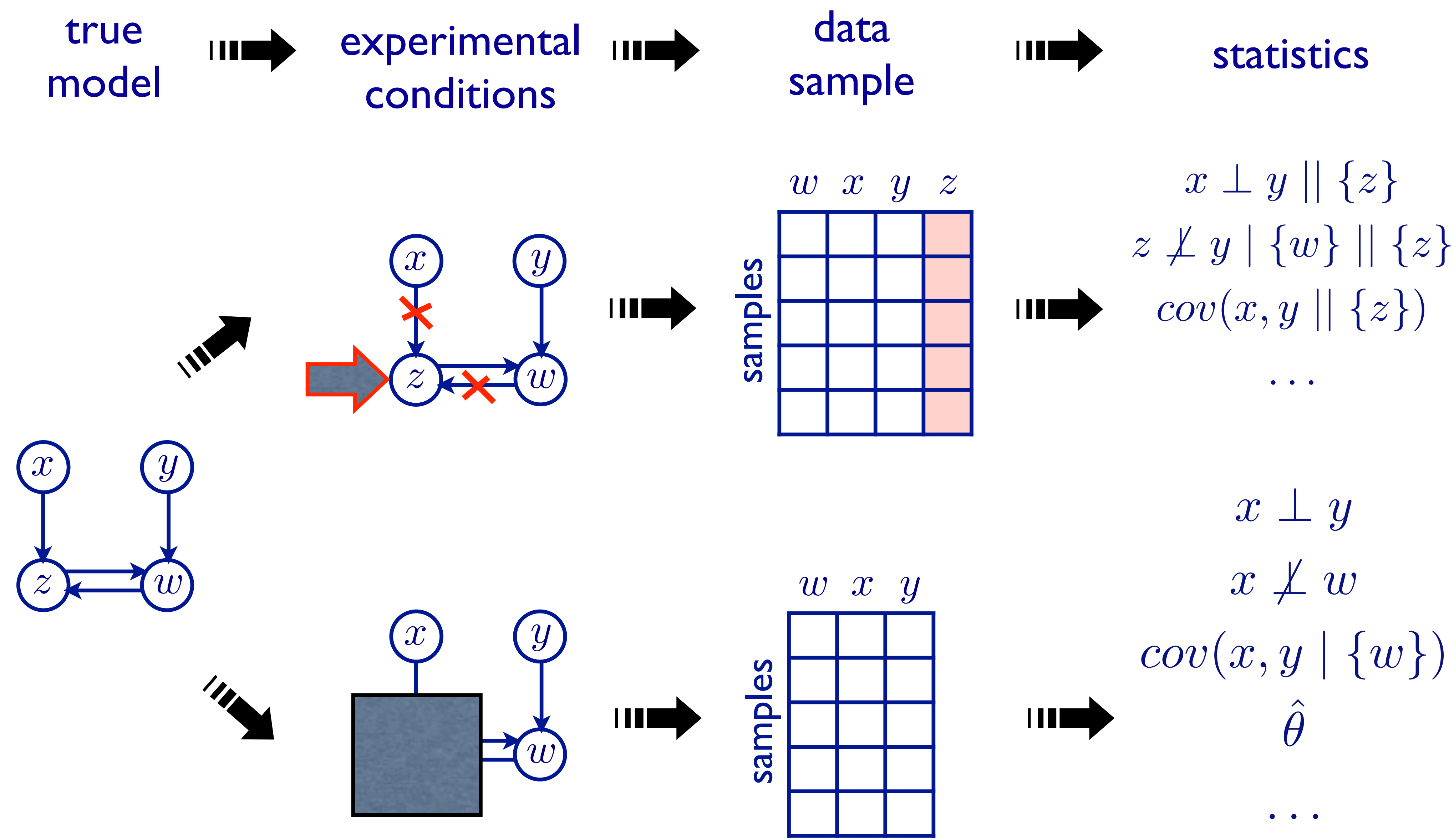


Causal Structure Search

true model → experimental conditions → data sample



Causal Structure Search



Output

statistics

$$x \perp y \mid\mid \{z\}$$

$$z \not\perp y \mid \{w\} \mid\mid \{z\}$$

$$cov(x, y \mid\mid \{z\})$$

$$\cdots$$

$$x \perp y$$

$$x \not\perp w$$

$$cov(x, y \mid \{w\})$$

$$\hat{\theta}$$

$$\cdots$$

Output

statistics

$$x \perp y \;||\; \{z\}$$

$$z \not\perp y \mid \{w\} \;||\; \{z\}$$

$$cov(x, y \;||\; \{z\})$$

$$\ldots$$

$$x \perp y$$

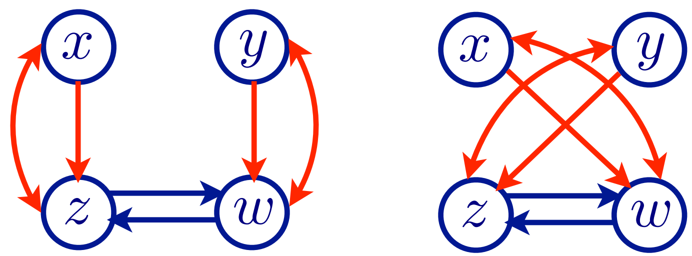
$$x \not\perp w$$

$$cov(x, y \mid \{w\})$$

$$\hat{\theta}$$

$$\ldots$$

equivalence classes



Output

statistics

$$x \perp y \,||\, \{z\}$$

$$z \not\perp y \mid \{w\} \,||\, \{z\}$$

$$cov(x, y \,||\, \{z\})$$

$$\ldots$$

$$x \perp y$$

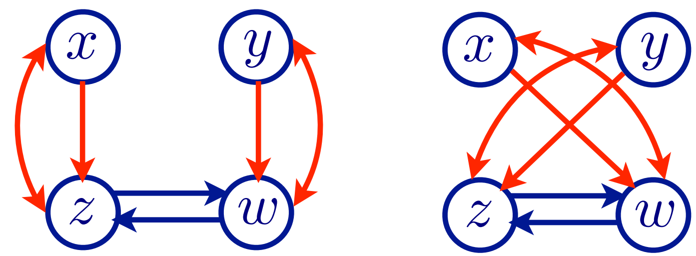
$$x \not\perp w$$

$$cov(x, y \mid \{w\})$$

$$\hat{\theta}$$

$$\ldots$$

equivalence classes



model specifications

	w	x	y	z
w	0	0	?	a
x	0	0	0	0
y	0	0	0	0
z	b	?	?	0

direct edges

	w	x	y	z
w	0	0	?	?
x	0	0	0	?
y	?	0	0	0
z	?	?	0	0

confounders

Search Space Assumptions

- Causal Markov
- Causal Faithfulness

Search Space Assumptions

- Causal Markov
- Causal Faithfulness
- overlapping data sets

Search Space Assumptions

- Causal Markov
- Causal Faithfulness
- overlapping data sets
- set of variables is jointly causally insufficient

Search Space Assumptions

- Causal Markov
- Causal Faithfulness
- overlapping data sets
- set of variables is jointly causally insufficient
- cyclic or acyclic causal structure

Search Space Assumptions

- Causal Markov
- Causal Faithfulness
- overlapping data sets
- set of variables is jointly causally insufficient
- cyclic or acyclic causal structure
- experimental and observational data sets

Search Space Assumptions

- Causal Markov
- Causal Faithfulness
- overlapping data sets
- set of variables is jointly causally insufficient
- cyclic or acyclic causal structure
- experimental and observational data sets
- d-separation oracle
  - in the cyclic case we can consider a linear Gaussian parameterization (see Spirtes, 1995)
  - known problems for the discrete case (see Pearl & Dechter, 1996, and Neal, 2000)

SAT-based causal discovery

graphical constraints

SAT-based causal discovery

graphical constraints

- x is a cause of y

SAT-based causal discovery

graphical constraints

- x is a cause of y
- there is a path from x to y
  (via z)

SAT-based causal discovery

graphical constraints

- x is a cause of y
- there is a path from x to y (via z)
- x and y are independent

SAT-based causal discovery

graphical constraints

- x is a cause of y
- there is a path from x to y (via z)
- x and y are independent
- x and y are correlated conditional on **C** in an experiment where x was subject to intervention

SAT-based causal discovery

graphical constraints

- x is a cause of y
- there is a path from x to y (via z)
- x and y are independent
- x and y are correlated conditional on **C** in an experiment where x was subject to intervention

propositional constraints (in CNF) on true graph

$$(A \vee B \vee C) \wedge (D \vee E)$$

$$A$$

$$B \wedge E$$

$$\ldots$$

SAT-based causal discovery

graphical constraints

- x is a cause of y
- there is a path from x to y (via z)
- x and y are independent
- x and y are correlated conditional on **C** in an experiment where x was subject to intervention

propositional constraints (in CNF) on true graph

$$(A \vee B \vee C) \wedge (D \vee E)$$

$$A$$

$$B \wedge E$$

$$\ldots$$

SAT-solver

SAT-based causal discovery

graphical constraints

- x is a cause of y
- there is a path from x to y (via z)
- x and y are independent
- x and y are correlated conditional on **C** in an experiment where x was subject to intervention

propositional constraints (in CNF) on true graph

$(A \vee B \vee C) \wedge (D \vee E)$

A

$B \wedge E$

$\dots$

SAT-solver

	w	x	y	z
w	0	0	?	1
x	0	0	0	0
y	0	0	0	0
z	?	?	?	0

direct edges

	w	x	y	z
w	0	0	?	?
x	0	0	0	?
y	?	0	0	0
z	?	?	0	0

confounders

SAT-based causal discovery

graphical constraints

- x is a cause of y
- there is a path from x to y (via z)
- x and y are independent
- x and y are correlated conditional on **C** in an experiment where x was subject to intervention

propositional constraints (in CNF) on true graph

$$(A \vee B \vee C) \wedge (D \vee E)$$

$$A$$

$$B \wedge E$$

$$\dots$$

SAT-solver

	w	x	y	z
w	0	0	?	1
x	0	0	0	0
y	0	0	0	0
z	?	?	?	0

direct edges

	w	x	y	z
w	0	0	?	?
x	0	0	0	?
y	?	0	0	0
z	?	?	0	0

confounders

SATisfiability solver

- finds a truth value assignment for a Boolean formula in Conjunctive Normal Form (CNF)

SATisfiability solver

- finds a truth value assignment for a Boolean formula in Conjunctive Normal Form (CNF)
- a Boolean term X is a **backbone variable** if X takes the same value (T or F) in all satisfying truth value assignments of a given formula

Encoding: track the endpoints of paths

$$[x \not\leq y \mid \mathbf{C} \parallel \mathbf{J}]$$

$$\Leftrightarrow$$

Encoding: track the endpoints of paths

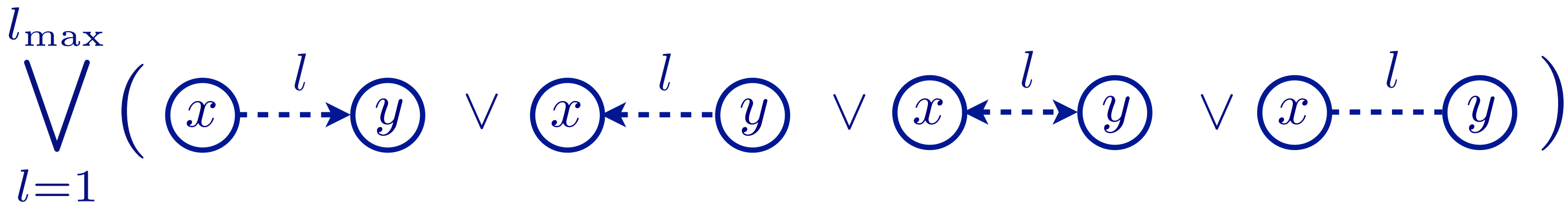
$$[x \not\perp y \mid \mathbf{C} \parallel \mathbf{J}]$$

$$\Leftrightarrow$$

Encoding: track the endpoints of paths

$$[x \not\perp y \mid \mathbf{C} \,\|\, \mathbf{J}]$$

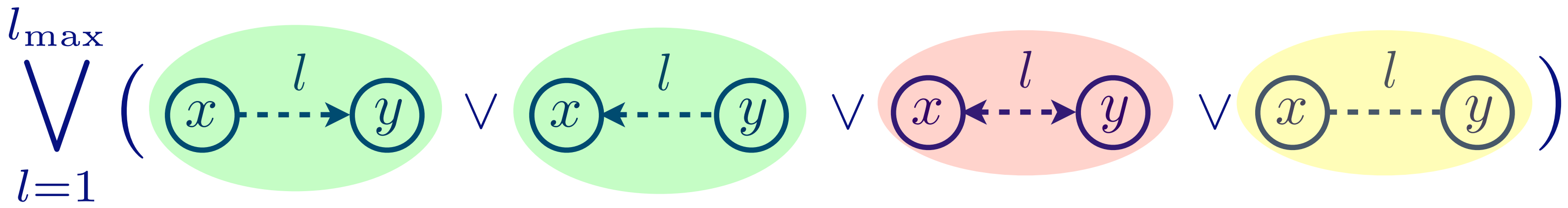
$$\Leftrightarrow$$



Encoding: track the endpoints of paths

$$[x \not\perp y \mid \mathbf{C} \mid\mid \mathbf{J}]$$

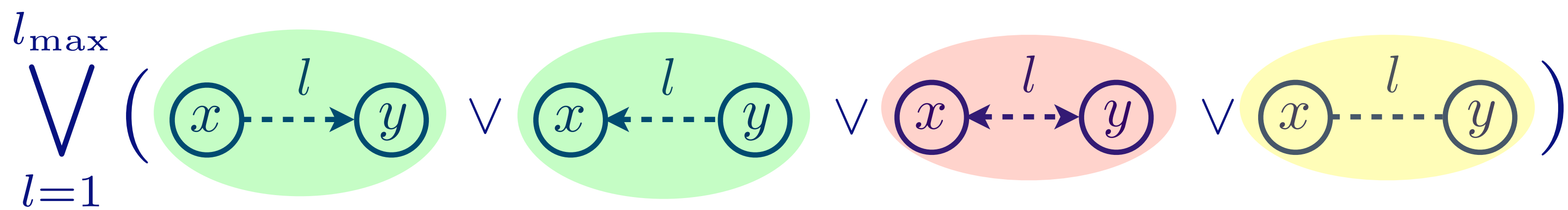
$$\Leftrightarrow$$

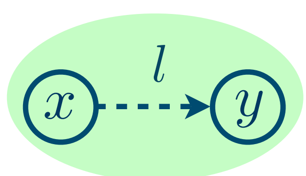


Encoding: track the endpoints of paths

$$[x \not\perp y \mid \mathbf{C} \,||\, \mathbf{J}]$$

$$\Leftrightarrow$$

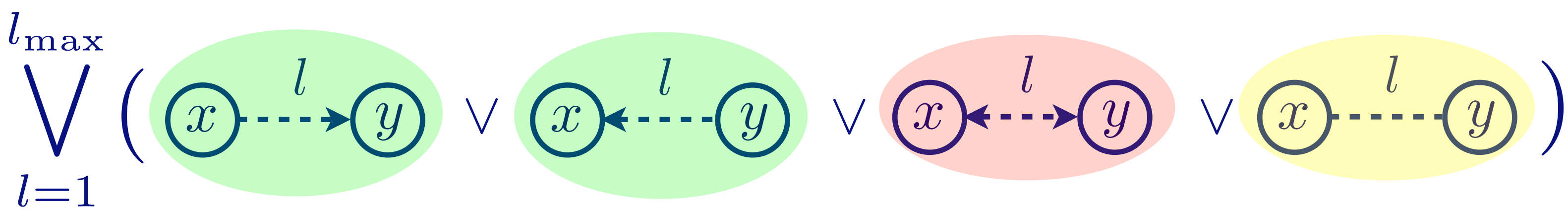


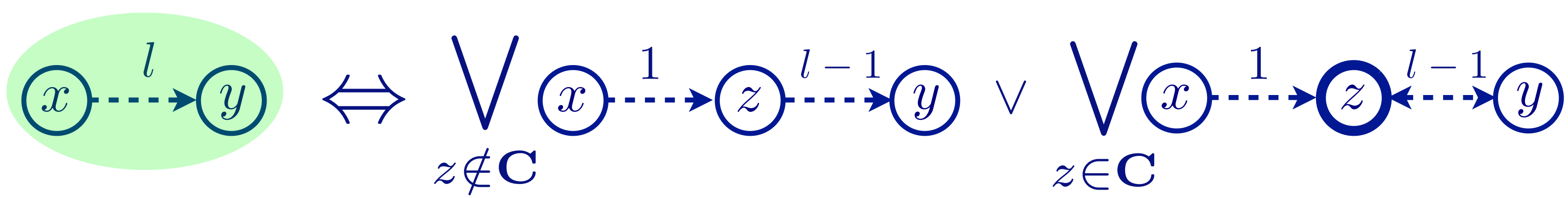


Encoding: track the endpoints of paths

$$[x \not\perp y \mid \mathbf{C} \mid\mid \mathbf{J}]$$

$$\Leftrightarrow$$

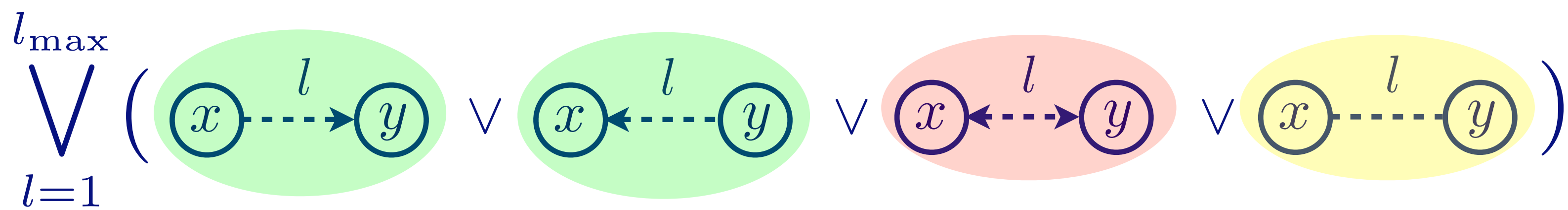


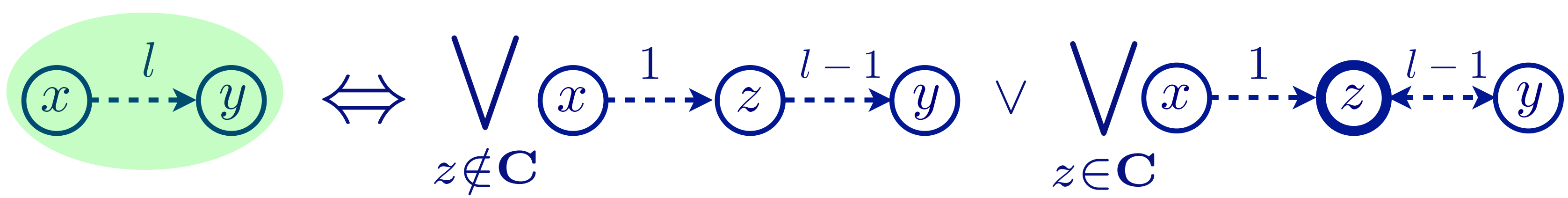


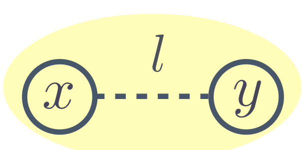
Encoding: track the endpoints of paths

$$[x \not\perp y \mid \mathbf{C} \,\|\, \mathbf{J}]$$

$$\Leftrightarrow$$



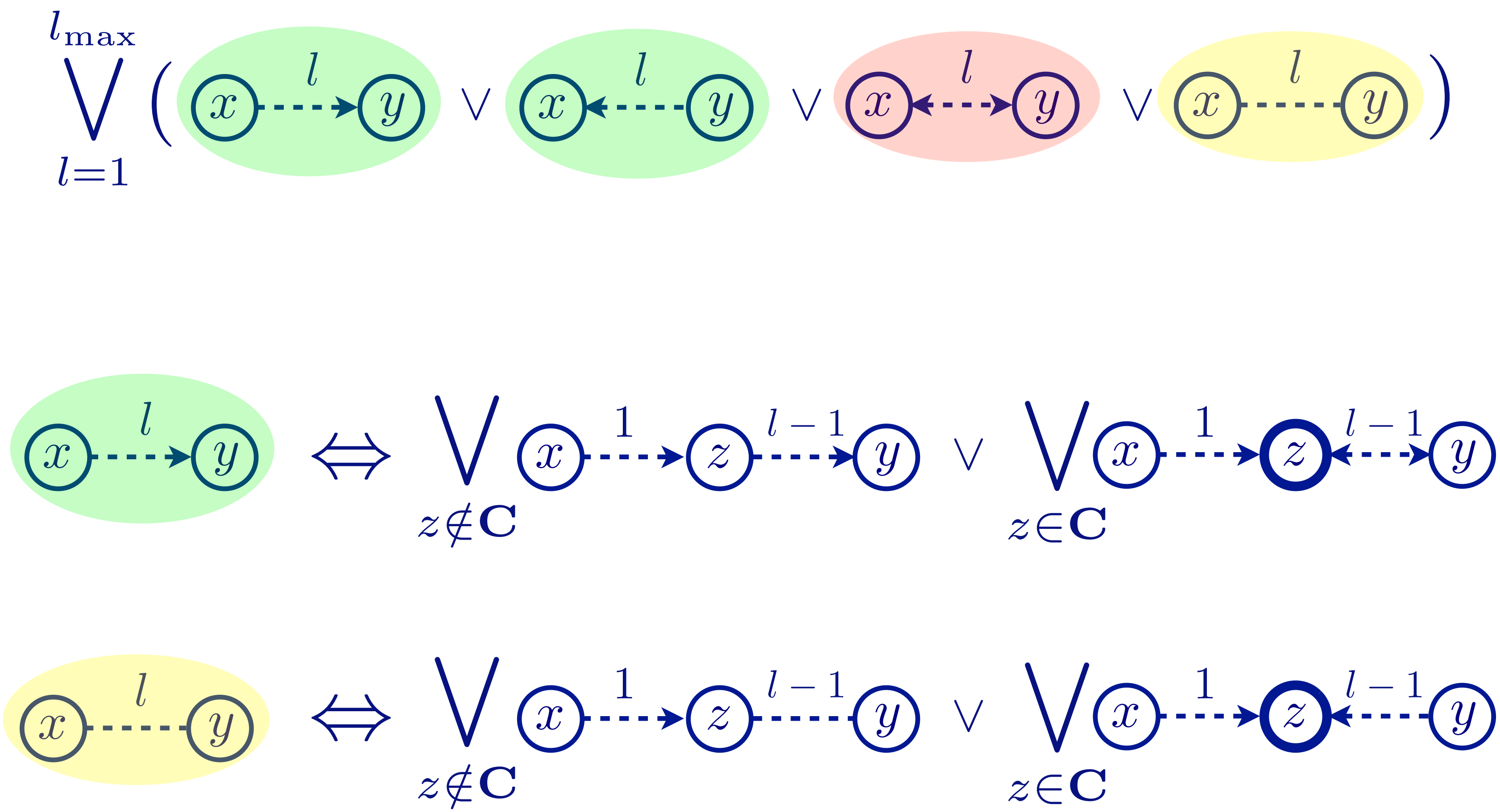




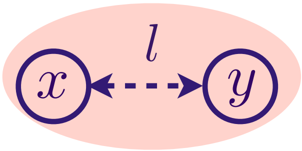
Encoding: track the endpoints of paths

$$[x \not\perp y \mid \mathbf{C} \mid\mid \mathbf{J}]$$

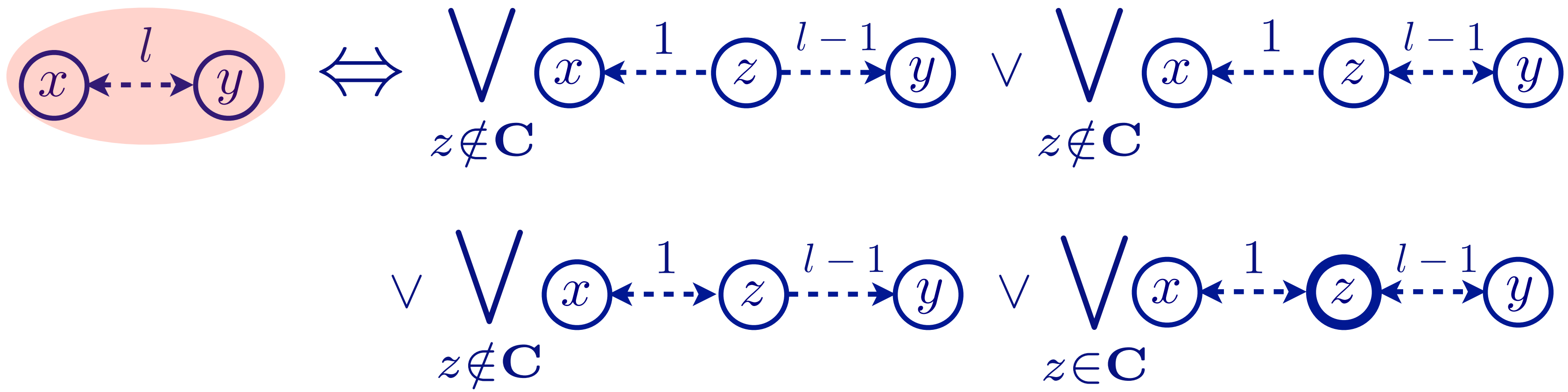
$$\Leftrightarrow$$



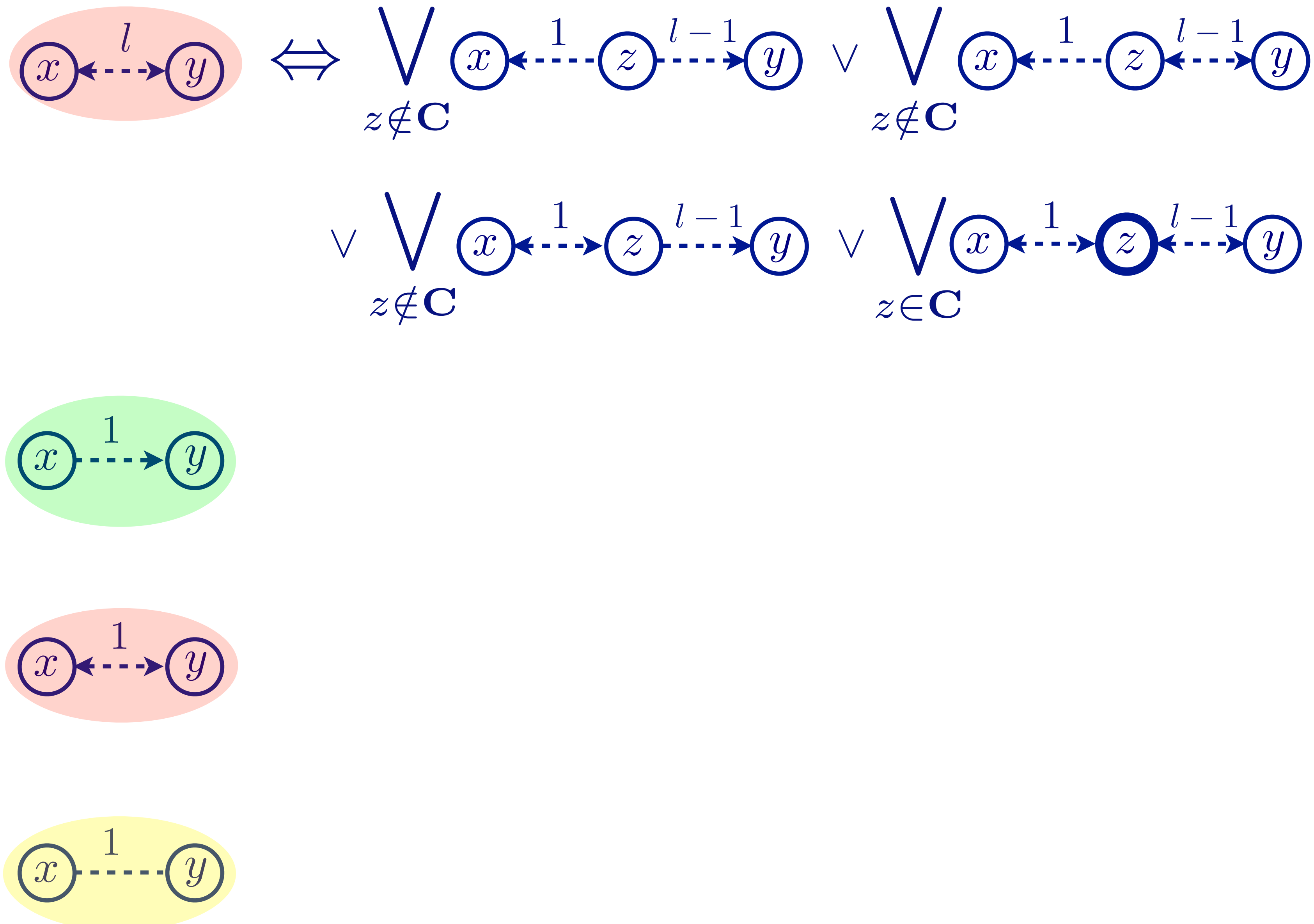
Encoding continued



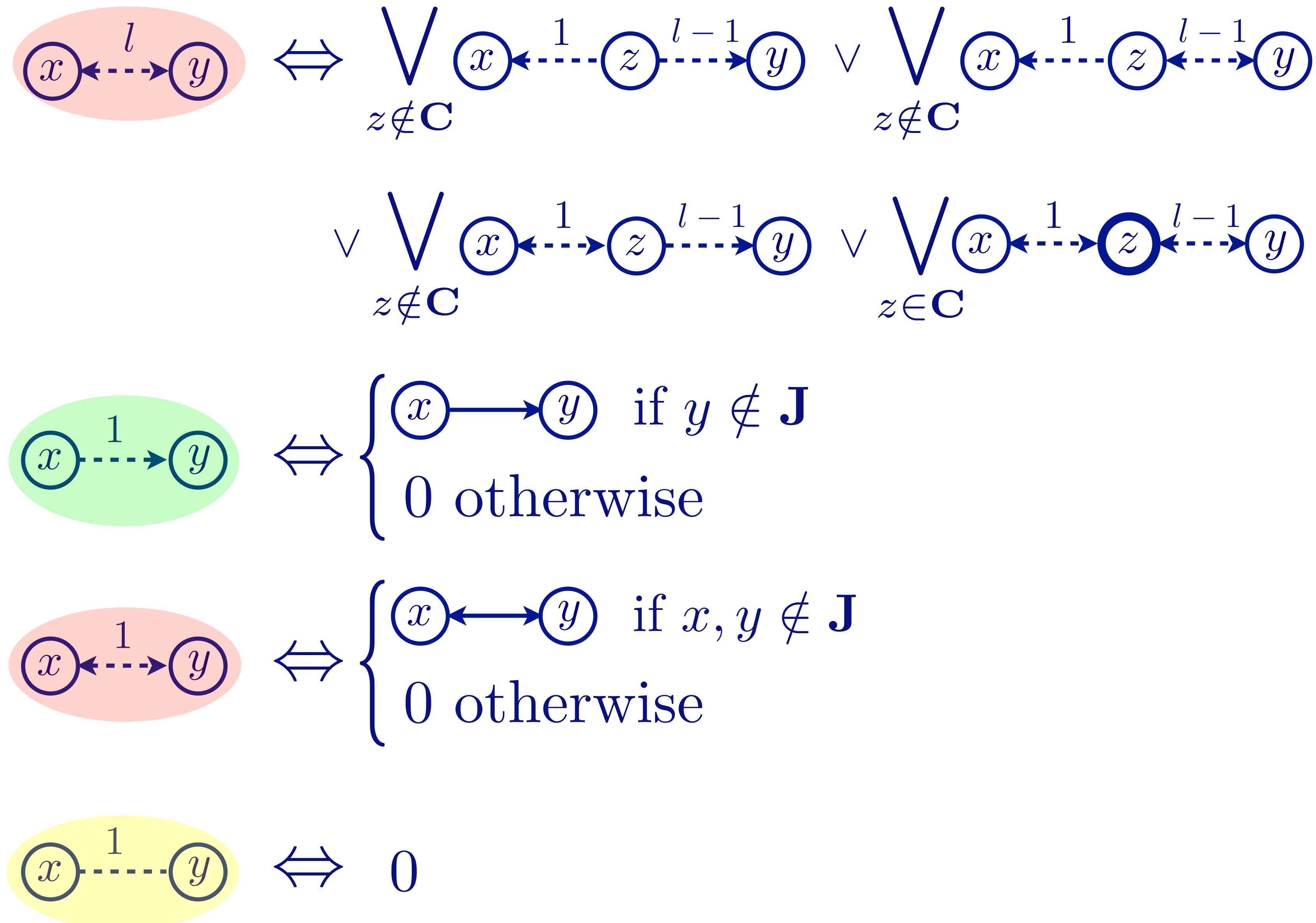
Encoding continued



Encoding continued



Encoding continued



$$[x \not\perp y \,|\, \mathbf{C} \,||\, \mathbf{J}] \quad \Leftrightarrow \quad \bigvee_{l=1}^{l_{\max}} \left([x \overset{l}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} y] \vee [y \overset{l}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} x] \vee [x \overset{l}{\underset{\mathbf{C},\mathbf{J}}{<\cdots>}} y] \vee [x \overset{l}{\underset{\mathbf{C},\mathbf{J}}{-\cdots-}} y] \right)$$

$$[x \overset{l}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} y] \quad \Leftrightarrow \quad \bigvee_{z \notin \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} y] \right) \vee \bigvee_{z \in \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C},\mathbf{J}}{<\cdots>}} y] \right)$$

$$\begin{aligned} [x \overset{l}{\underset{\mathbf{C},\mathbf{J}}{<\cdots>}} y] \quad \Leftrightarrow \quad & \bigvee_{z \notin \mathbf{C}} \left([z \overset{1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} x] \wedge [z \overset{l-1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} y] \right) \vee \bigvee_{z \notin \mathbf{C}} \left([z \overset{1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} x] \wedge [z \overset{l-1}{\underset{\mathbf{C},\mathbf{J}}{<\cdots>}} y] \right) \vee \\ & \bigvee_{z \notin \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{<\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} y] \right) \vee \bigvee_{z \in \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{<\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C},\mathbf{J}}{<\cdots>}} y] \right) \end{aligned}$$

$$[x \overset{l}{\underset{\mathbf{C},\mathbf{J}}{-\cdots-}} y] \quad \Leftrightarrow \quad \bigvee_{z \notin \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots-}} y] \right) \vee \bigvee_{z \in \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} z] \wedge [y \overset{l-1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} z] \right)$$

$$[x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots>}} y] \quad \Leftrightarrow \quad \begin{cases} [x \rightarrow y] & \text{if } y \notin \mathbf{J} \\ 0 & \text{otherwise} \end{cases}$$

$$[x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{<\cdots>}} y] \quad \Leftrightarrow \quad \begin{cases} [x \leftrightarrow y] & \text{if } x \notin \mathbf{J} \text{ and } y \notin \mathbf{J} \\ 0 & \text{otherwise} \end{cases}$$

$$[x \overset{1}{\underset{\mathbf{C},\mathbf{J}}{-\cdots-}} y] \quad \Leftrightarrow \quad 0$$

$$[x \not\perp y \,|\mathbf{C}\,||\mathbf{J}] \;\Leftrightarrow\; \bigvee_{l=1}^{l_{\max}} \left([x \overset{l}{\underset{\mathbf{C,J}}{-\cdots>}} y] \vee [y \overset{l}{\underset{\mathbf{C,J}}{-\cdots>}} x] \vee [x \overset{l}{\underset{\mathbf{C,J}}{<\cdots>}} y] \vee [x \overset{l}{\underset{\mathbf{C,J}}{-\cdots-}} y] \right)$$

$$[x \overset{l}{\underset{\mathbf{C,J}}{-\cdots>}} y] \;\Leftrightarrow\; \bigvee_{z \notin \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C,J}}{-\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C,J}}{-\cdots>}} y] \right) \vee \bigvee_{z \in \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C,J}}{-\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C,J}}{<\cdots>}} y] \right)$$

$$\begin{aligned}[x \overset{l}{\underset{\mathbf{C,J}}{<\cdots>}} y] \;\Leftrightarrow\; & \bigvee_{z \notin \mathbf{C}} \left([z \overset{1}{\underset{\mathbf{C,J}}{-\cdots>}} x] \wedge [z \overset{l-1}{\underset{\mathbf{C,J}}{-\cdots>}} y] \right) \vee \bigvee_{z \notin \mathbf{C}} \left([z \overset{1}{\underset{\mathbf{C,J}}{-\cdots>}} x] \wedge [z \overset{l-1}{\underset{\mathbf{C,J}}{<\cdots>}} y] \right) \vee \\ & \bigvee_{z \notin \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C,J}}{<\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C,J}}{-\cdots>}} y] \right) \vee \bigvee_{z \in \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C,J}}{<\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C,J}}{<\cdots>}} y] \right)\end{aligned}$$

$$[x \overset{l}{\underset{\mathbf{C,J}}{-\cdots-}} y] \;\Leftrightarrow\; \bigvee_{z \notin \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C,J}}{-\cdots>}} z] \wedge [z \overset{l-1}{\underset{\mathbf{C,J}}{-\cdots-}} y] \right) \vee \bigvee_{z \in \mathbf{C}} \left([x \overset{1}{\underset{\mathbf{C,J}}{-\cdots>}} z] \wedge [y \overset{l-1}{\underset{\mathbf{C,J}}{-\cdots>}} z] \right)$$

$$[x \overset{1}{\underset{\mathbf{C,J}}{-\cdots>}} y] \;\Leftrightarrow\; \begin{cases} [x \rightarrow y] & \text{if } y \notin \mathbf{J} \\ 0 & \text{otherwise} \end{cases}$$

$$[x \overset{1}{\underset{\mathbf{C,J}}{<\cdots>}} y] \;\Leftrightarrow\; \begin{cases} [x \leftrightarrow y] & \text{if } x \notin \mathbf{J} \text{ and } y \notin \mathbf{J} \\ 0 & \text{otherwise} \end{cases}$$

$$[x \overset{1}{\underset{\mathbf{C,J}}{-\cdots-}} y] \;\Leftrightarrow\; 0$$

longest path that needs to be considered: l_max = 2n-4 where n = |V|

Example of Encoding

For a network of n=**10** variables

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)
- 2^135 ~ **10^40** different graphs

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)
- 2^135 ~ **10^40** different graphs

For a data set

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)
- 2^135 ~ **10^40** different graphs

For a data set

- 10*9/2*2^8 = **11520** different conditional dsep/dcon relations

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)
- 2^135 ~ **10^40** different graphs

For a data set

- 10*9/2*2^8 = **11520** different conditional dsep/dcon relations
- 2^10 = **1024** different intervention sets

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)
- 2^135 ~ **10^40** different graphs

For a data set

- 10*9/2*2^8 = **11520** different conditional dsep/dcon relations
- 2^10 = **1024** different intervention sets
- longest d-connecting path that needs to be considered is

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)
- 2^135 ~ **10^40** different graphs

For a data set

- 10*9/2*2^8 = **11520** different conditional dsep/dcon relations
- 2^10 = **1024** different intervention sets
- longest d-connecting path that needs to be considered is l_{max} = 2*n-4 = **16**

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)
- 2^135 ~ **10^40** different graphs

For a data set

- 10*9/2*2^8 = **11520** different conditional dsep/dcon relations
- 2^10 = **1024** different intervention sets
- longest d-connecting path that needs to be considered is l_{max} = 2*n-4 = **16**
- ~ **5million** path variables

Example of Encoding

For a network of n=**10** variables

- 10*9+10*9/2 = **135** possible edges (incl. confounders)
- 2^135 ~ **10^40** different graphs

For a data set

- 10*9/2*2^8 = **11520** different conditional dsep/dcon relations
- 2^10 = **1024** different intervention sets
- longest d-connecting path that needs to be considered is l_{max} = 2*n-4 = **16**
- ~ **5million** path variables
- **Gigabytes** of CNF formulas, but only define those that you need!

Algorithm

Proceed in order of conditioning set size

- heuristically find unknown d-separation / d-connection relations and determine them.
- Encode the relations into the working formula F, including definitions as needed.
- Determine the "backbone" of F using the SAT-solver, i.e. for each pair of variables (x,y) in **V** and for each edge type determine whether it is
    - **present** in all causal structures consistent with the input.
    - **absent** in all causal structures consistent with the input.
    - **unknown**, i.e. present in some, and absent in other causal structures consistent with the input.

Algorithm

any **background knowledge** representable using encoding can be included

Proceed in order of conditioning set size

- heuristically find unknown d-separation / d-connection relations and determine them.
- Encode the relations into the working formula F, including definitions as needed.
- Determine the "backbone" of F using the SAT-solver, i.e. for each pair of variables (x,y) in **V** and for each edge type determine whether it is
  - **present** in all causal structures consistent with the input.
  - **absent** in all causal structures consistent with the input.
  - **unknown**, i.e. present in some, and absent in other causal structures consistent with the input.

Algorithm

Proceed in order of conditioning set size

- heuristically find unknown d-separation / d-connection relations and determine them.
- Encode the relations into the working formula F, including definitions as needed.
- Determine the "backbone" of F using the SAT-solver, i.e. for each pair of variables (x,y) in **V** and for each edge type determine whether it is
  - **present** in all causal structures consistent with the input.
  - **absent** in all causal structures consistent with the input.
  - **unknown**, i.e. present in some, and absent in other causal structures consistent with the input.

any **background knowledge** representable using encoding can be included

d-separation constraints can be treated separately from d-connection constraints

Algorithm

Proceed in order of conditioning set size

- heuristically find unknown d-separation / d-connection relations and determine them.
- Encode the relations into the working formula F, including definitions as needed.
- Determine the "backbone" of F using the SAT-solver, i.e. for each pair of variables (x,y) in **V** and for each edge type determine whether it is
  - **present** in all causal structures consistent with the input.
  - **absent** in all causal structures consistent with the input.
  - **unknown**, i.e. present in some, and absent in other causal structures consistent with the input.

any **background knowledge** representable using encoding can be included

d-separation constraints can be treated separately from d-connection constraints

you can compute the backbone over any graphical feature that you are interested in

Compare to FCI or CCD

- Both FCI and CCD use a representation of the current equivalence class to inform the choice of next independence test

Compare to FCI or CCD

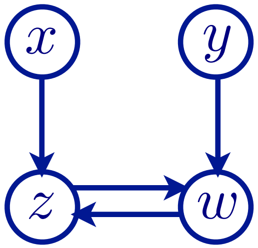
- Both FCI and CCD use a representation of the current equivalence class to inform the choice of next independence test
- For our search space we have no such representation

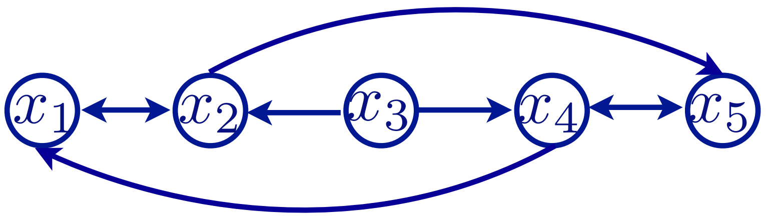
Compare to FCI or CCD

- Both FCI and CCD use a representation of the current equivalence class to inform the choice of next independence test
- For our search space we have no such representation
  - recall the blow-up of the equivalence class in the search space of **acyclic** models in overlapping data sets for the ION and IOD algorithm (see Tillman et al. 2009, Triantafillou et al. 2010)

Completeness

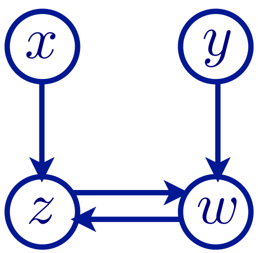
- The procedure is d-separation complete: it determines all d-separation relations that can be determined given the data sets (without, in general, doing all possible tests).

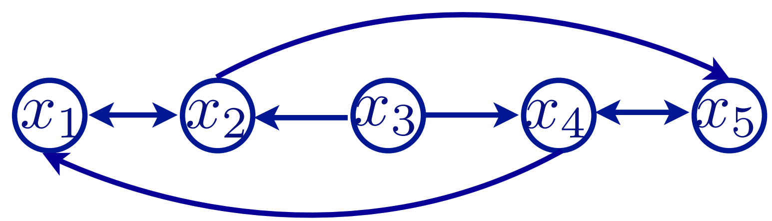




Completeness

- The procedure is d-separation complete: it determines all d-separation relations that can be determined given the data sets (without, in general, doing all possible tests).





  - in restricted search spaces we can copy the test schedules of e.g. PC, FCI, CCD or ION for efficiency (while integrating background or experimental constraints)

Handling Statistical Errors

- do what any other constraint based methods do
  - retract and return “don’t knows” when conflicts arise
  - focus on reliable tests first and stop
  - different cut-offs for d-separation vs. d-connection
  - try doing some type of false discovery rate control

➡ use weighted maxSAT techniques

Conclusion

- a constraint based inference procedure for a search space that includes causal models **with latents and cycles**
- combination of input obtained from **experimental** or **observational overlapping** data sets
- inclusion of **wide variety of background knowledge**
- change to a query based approach to causal discovery
- code package availabe

References to Related Work

Claassen, T. and Heskes, T. (2011). A logical characterization of constraint-based causal discovery. In Proc. UAI 2011.

Claassen, T. and Heskes, T. (2010). Causal discovery discovery in multiple models from different experiments. In Proc. NIPS, pages 415–423.

Lagani, V., Tsamardinos, I., and Triantafilou, S. (2012). Learning from mixture of experimental data: A constraint-based approach. In Proc. SETN, pages 124–131.

Tillman, R. E., Danks, D., and Glymour, C. (2009). Integrating locally learned causal structures with overlapping variables. In Proc. NIPS 2008, pages 1665–1672.

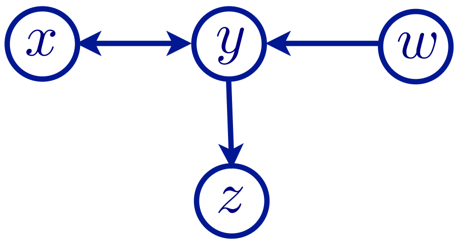
Triantafillou, S., Tsamardinos, I., and Tollis, I. G. (2010). Learning causal structure from overlapping variable sets. In Proc. AISTATS, pages 860–867.

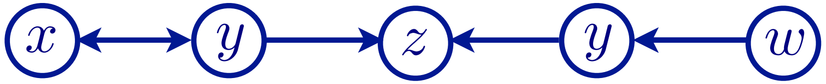
Tsamardinos, I., Triantafillou, S., and Lagani, V. (2012). Towards integrative causal analysis of heterogeneous data sets and studies. Journal of Machine Learning Research, 13:1097–1157.

Our research was supported by the Academy of Finland, HIIT and the James S. McDonnell Foundation.

d-separation

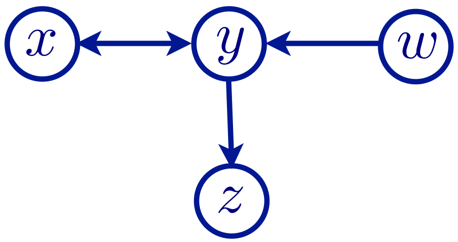
path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.

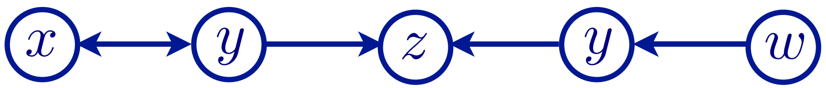




d-separation

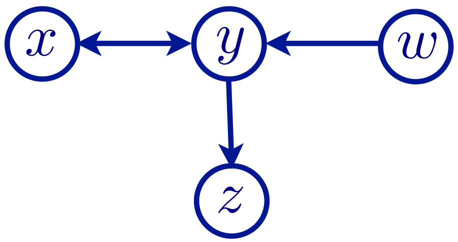
path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.

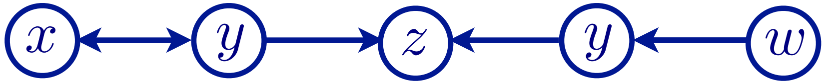




d-separation

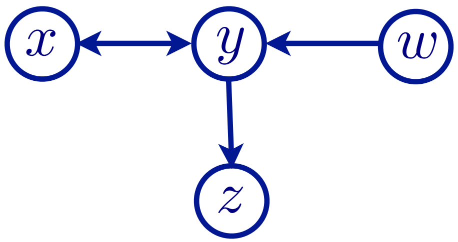
path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.

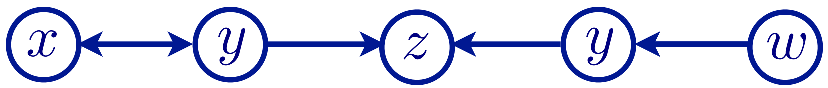




d-separation

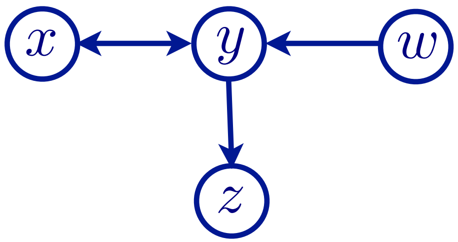
path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.

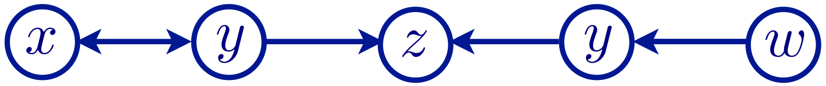




d-separation

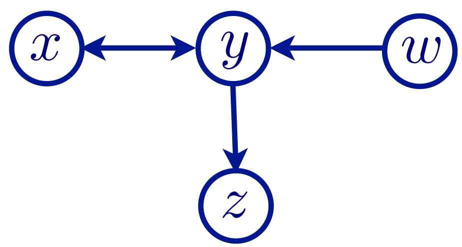
path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.

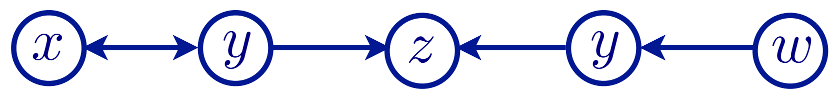




d-separation

path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.

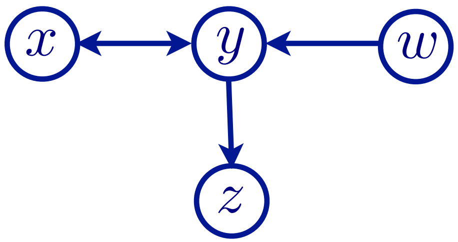


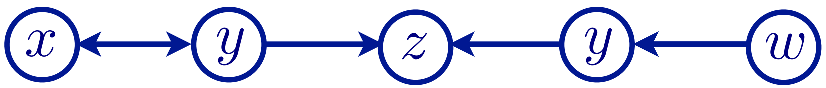


d-separation: a path is d-connecting with respect to a conditioning set **C** if every collider c on the path is in **C** and no other nodes on the path are in **C**, otherwise the path is d-separated. Two nodes are d-connected if there is a d-connecting path between the two nodes.

d-separation

path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.



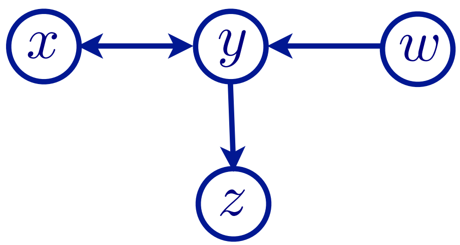


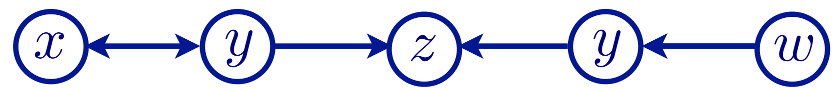
d-separation: a path is d-connecting with respect to a conditioning set **C** if every collider c on the path is in **C** and no other nodes on the path are in **C**, otherwise the path is d-separated. Two nodes are d-connected if there is a d-connecting path between the two nodes.

Nb: Equivalent to standard definition of d-separation, see Studeny (1998), and Koster (2002).

d-separation

path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.





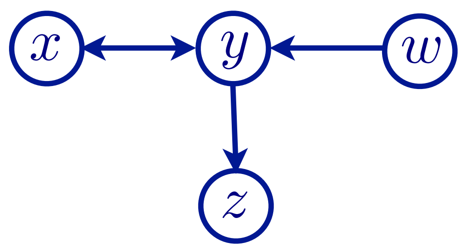
d-separation: a path is d-connecting with respect to a conditioning set **C** if every collider c on the path is in **C** and no other nodes on the path are in **C**, otherwise the path is d-separated. Two nodes are d-connected if there is a d-connecting path between the two nodes.

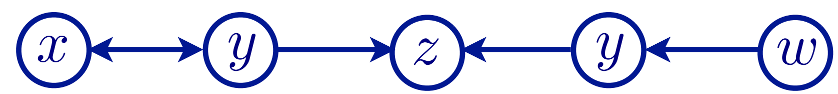
Nb: Equivalent to standard definition of d-separation, see Studeny (1998), and Koster (2002).

$$x \not\perp y \mid \mathbf{C} \mid\mid \mathbf{J}$$

d-separation

path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.





d-separation: a path is d-connecting with respect to a conditioning set **C** if every collider c on the path is in **C** and no other nodes on the path are in **C**, otherwise the path is d-separated. Two nodes are d-connected if there is a d-connecting path between the two nodes.

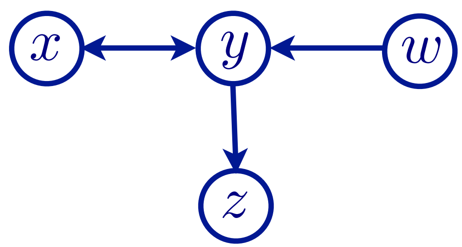
d-connection

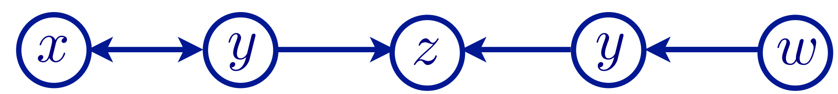
$$x \not\perp y \mid \mathbf{C} \mid\mid \mathbf{J}$$

Nb: Equivalent to standard definition of d-separation, see Studeny (1998), and Koster (2002).

d-separation

path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.





d-separation: a path is d-connecting with respect to a conditioning set **C** if every collider c on the path is in **C** and no other nodes on the path are in **C**, otherwise the path is d-separated. Two nodes are d-connected if there is a d-connecting path between the two nodes.

Nb: Equivalent to standard definition of d-separation, see Studeny (1998), and Koster (2002).

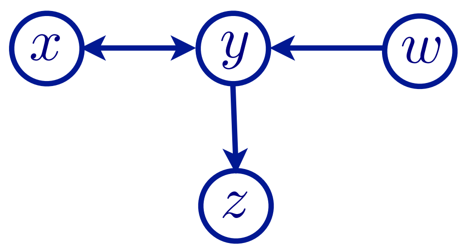
d-connection

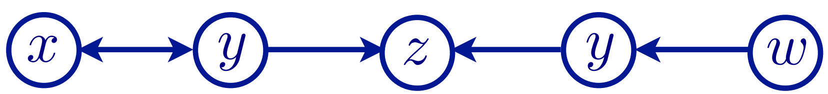
$$x \not\perp y \mid \mathbf{C} \mid\mid \mathbf{J}$$

conditioning set

d-separation

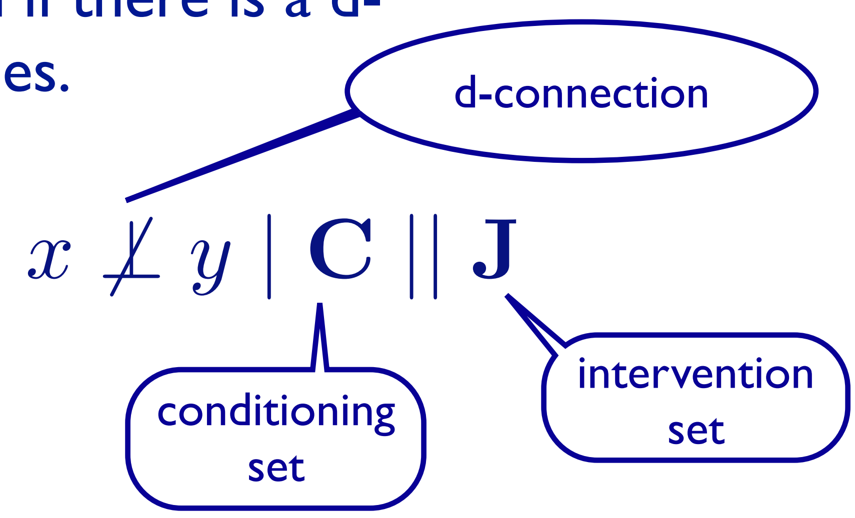
path: sequence of consecutive edges in the graph, without any restrictions on the types or orientations of the edges involved.



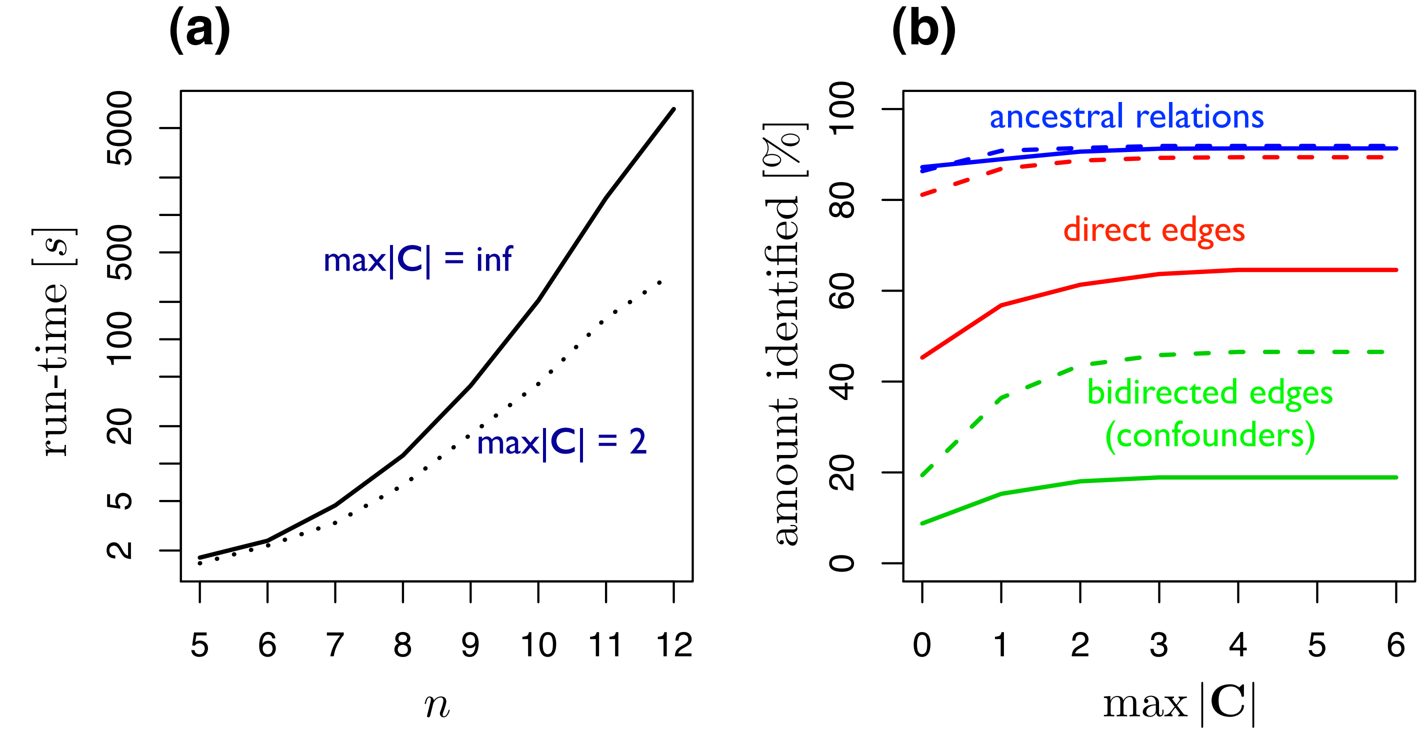


d-separation: a path is d-connecting with respect to a conditioning set **C** if every collider c on the path is in **C** and no other nodes on the path are in **C**, otherwise the path is d-separated. Two nodes are d-connected if there is a d-connecting path between the two nodes.

Nb: Equivalent to standard definition of d-separation, see Studeny (1998), and Koster (2002).



Simulations: Complexity & Identifiability

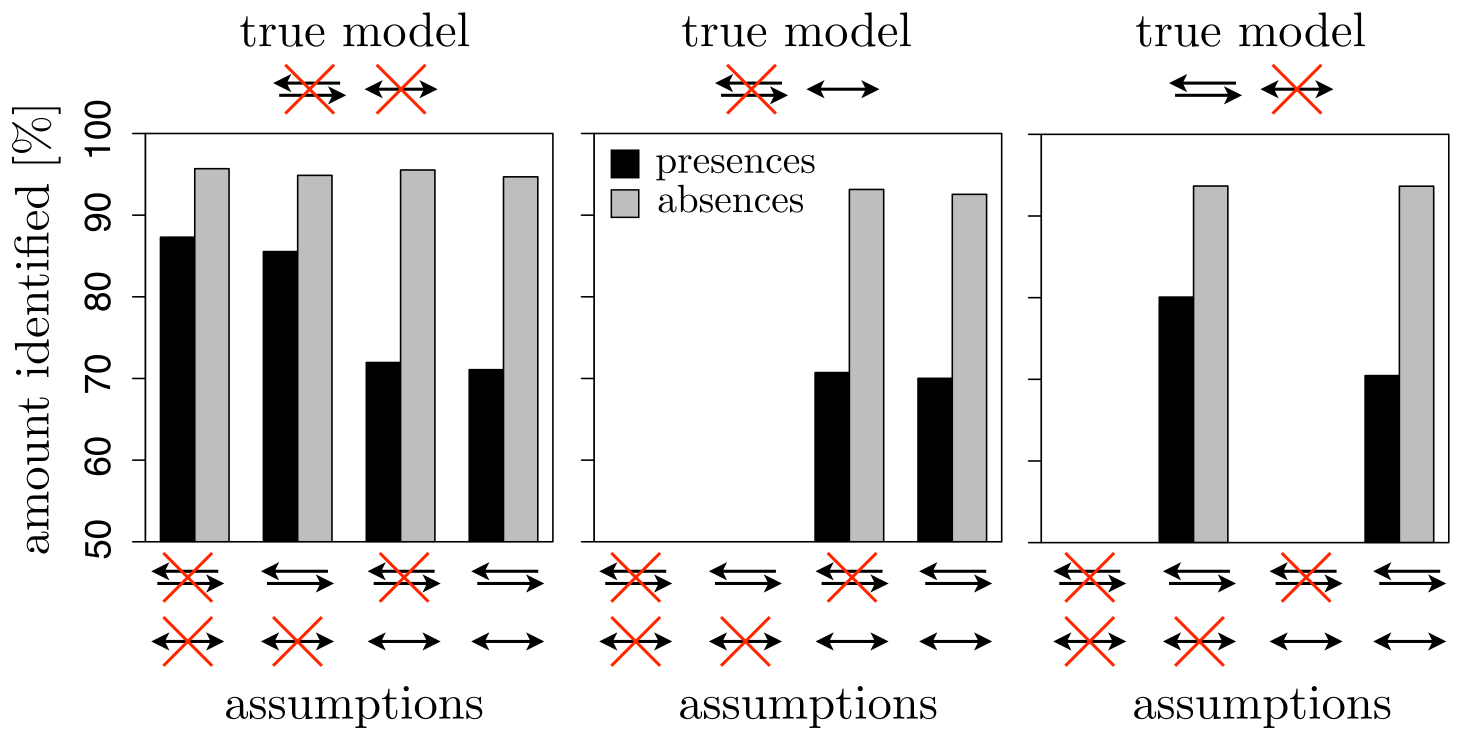


(a) Median runtime of the procedure as a function of the total number of nodes in the model.

(b) Proportion of edges (solid lines) and absences of edges (dashed) identified as a function of max |C|

100 graphs, edge probability 0.2, 10 overlapping experiments with equal probability for each node to be observed, intervened or hidden.

Simulations: Search Space Assumptions



- Proportion of directed edge presences and absences identified, under various model space assumptions, for acyclic true models without latent variables (left), acyclic models with latents (center), and cyclic models without latents (right).

Test Pruning Heuristic

- Given an intermediary solution of **present, absent & unknown** edges:
  - Consider a minimal model: all unknown edges absent
  - Consider a maximal model: all unknown edges present
  - Search for d-separation relations in which the minimal and maximal model differ
  - Omit tests that contain nodes that cannot be on a d-connecting path (e.g. nodes known to be disconnected)