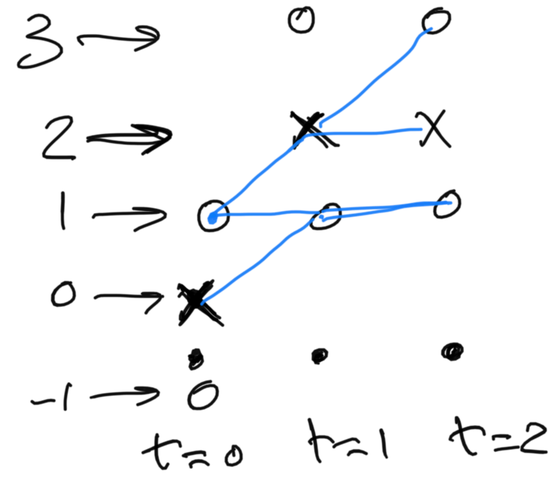
Modeling object tracking

Problem: object tracking

Noisy sensors report positions: $0, 2, 2$.

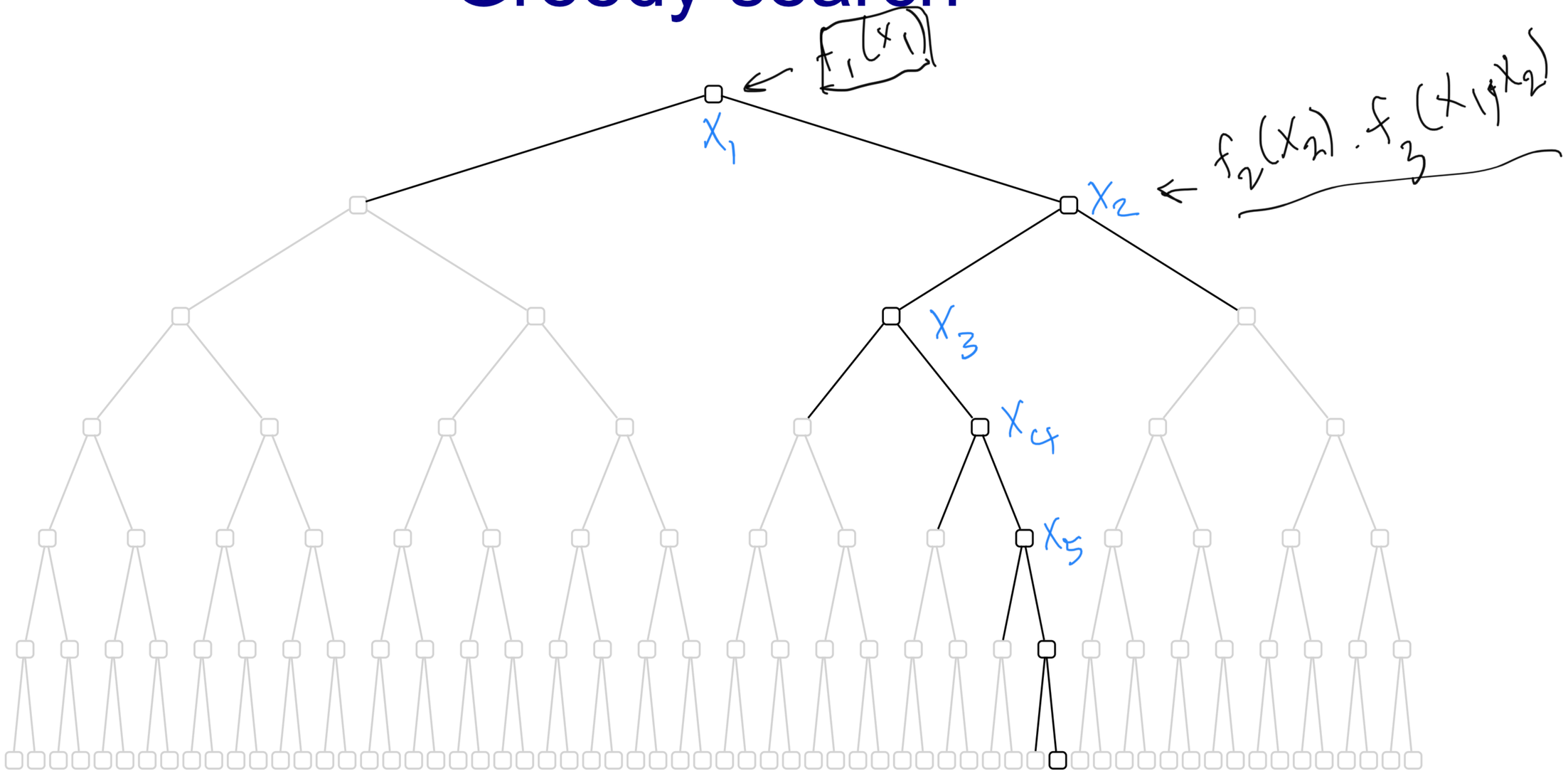
Objects don't move very fast.

What path did the object take?

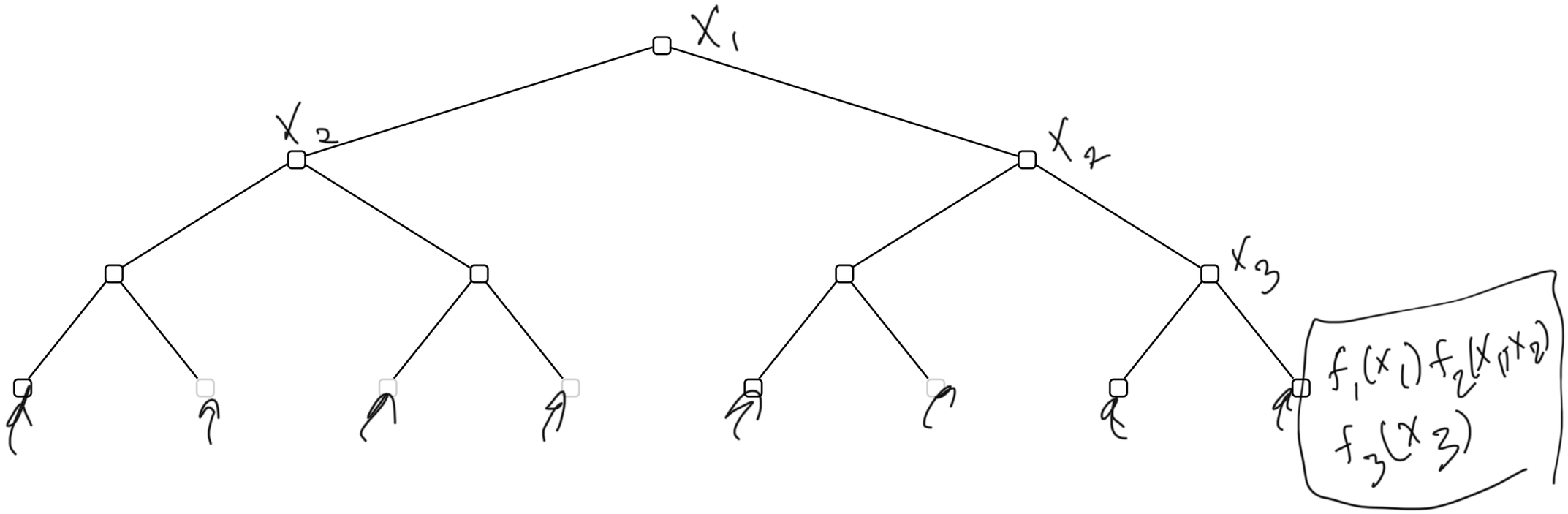


[whiteboard: trajectories over time]

Greedy search



Beam search



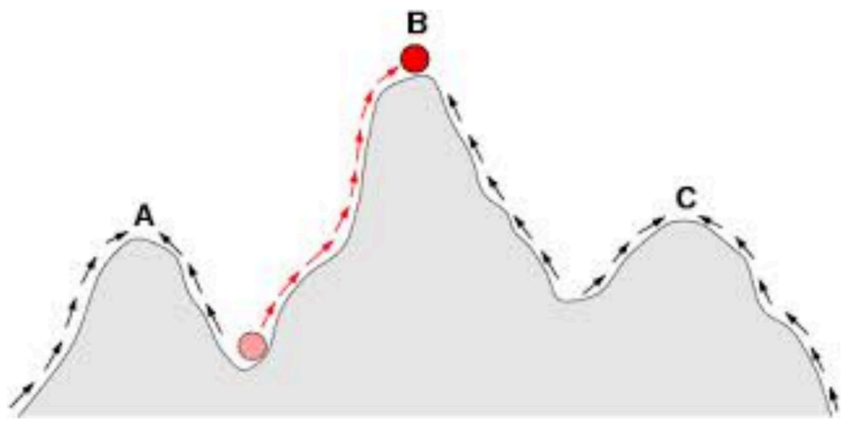
Beam size $K = 4$

Iterated conditional modes (ICM)

[demo: `iteratedConditionalModes()`]

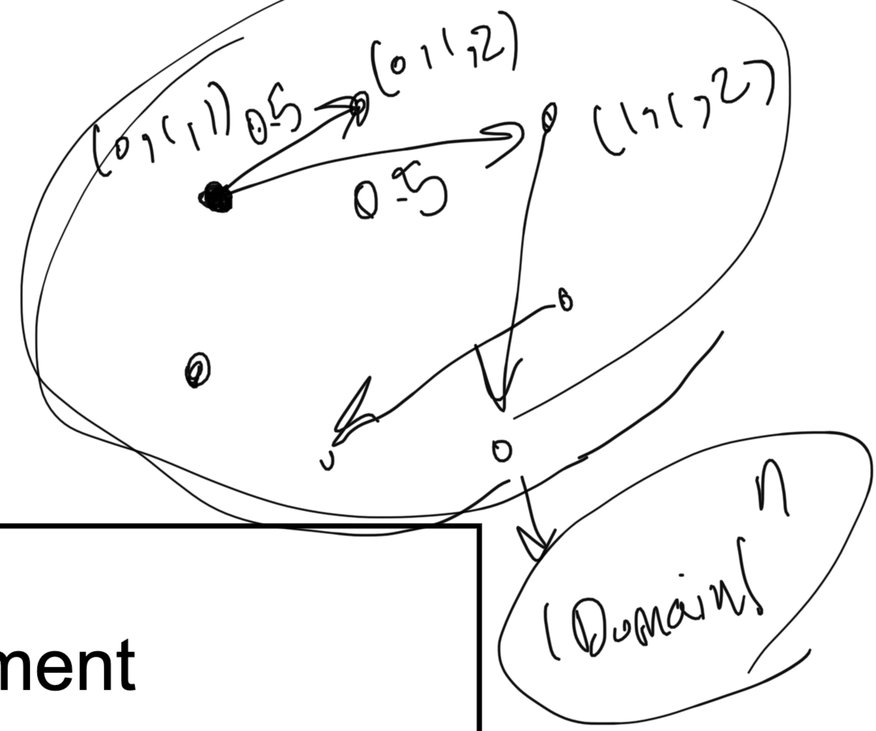
- $\mathbf{Weight}(x)$ increases or stays the same each iteration
- Converges in a finite number of iterations
- Can get stuck in **local optima**

- Not guaranteed to find optimal assignment!



Gibbs sampling

[demo: `gibbsSampling()`]



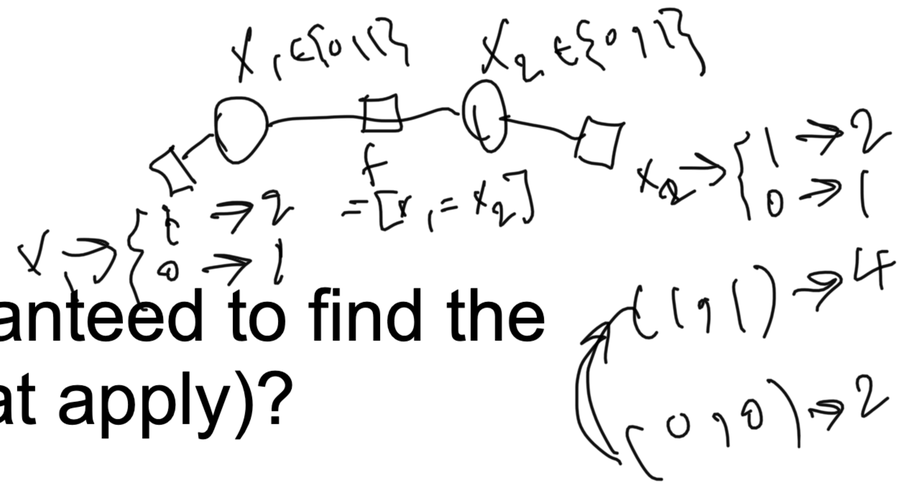
Algorithm: Gibbs sampling

Initialize x to a random complete assignment
Loop through $i = 1, \ldots, n$ until convergence:
  Compute weight of $x_v = x \cup \{X_i : v\}$ for each v
  Choose $x \leftarrow x_v$ with probability prop. to its weight

Can escape from local optima (not always easy though)!

answer in chat

Question



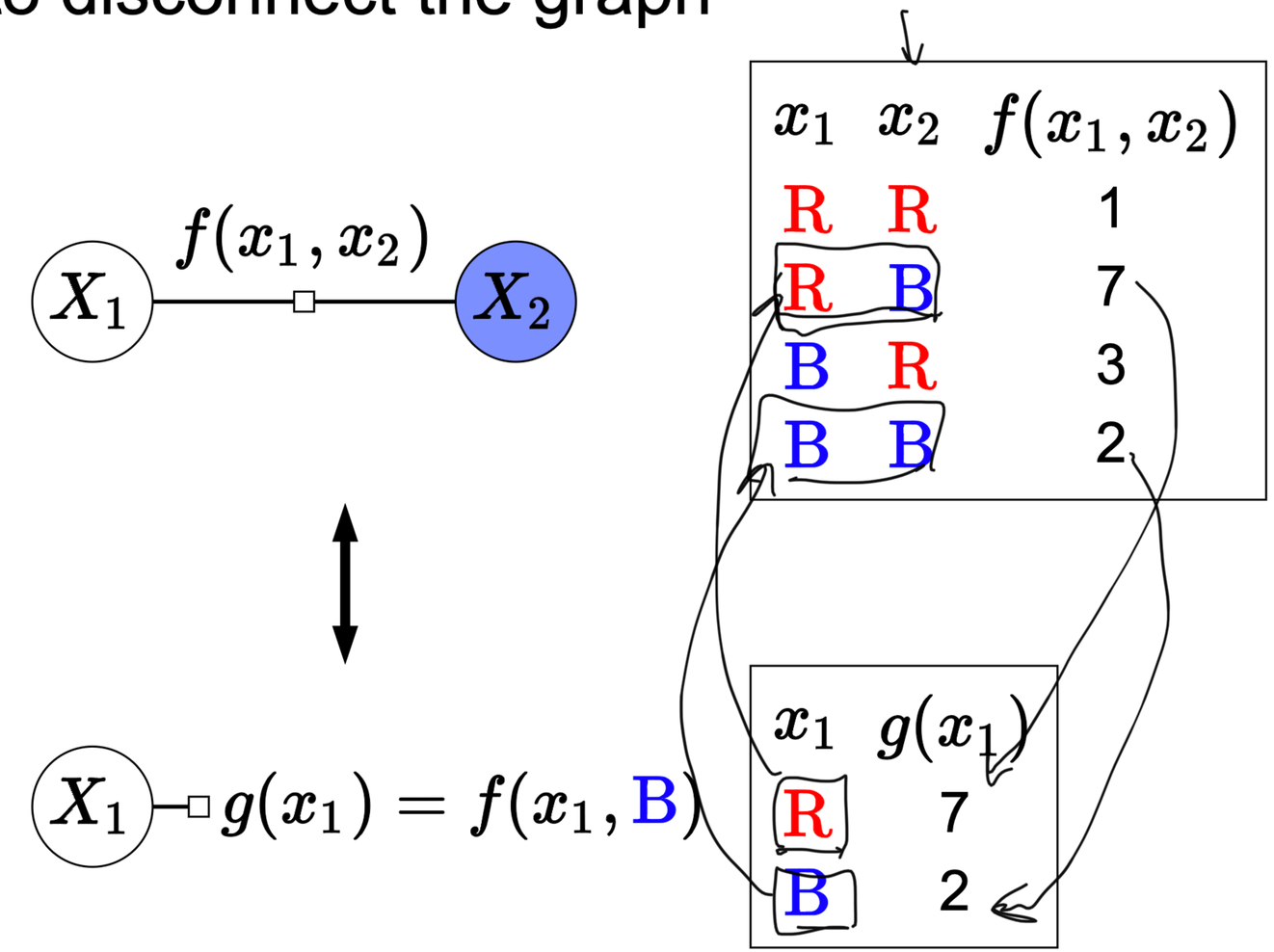
Which of the following algorithms are guaranteed to find the maximum weight assignment (select all that apply)?

- backtracking search
- greedy search
- beam search
- iterated conditional modes
- Gibbs sampling

activate deactivate reset report

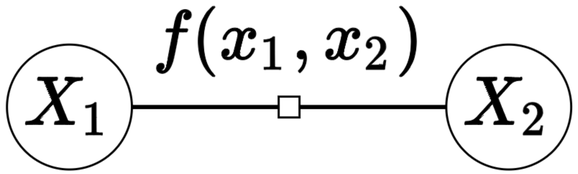
Conditioning

Goal: try to disconnect the graph



Condition on $X_2 = \text{B}$: remove X_2, f and add g

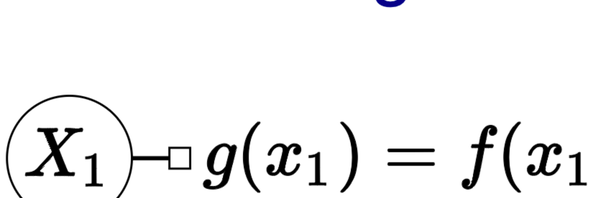
Conditioning versus elimination



x_1	x_2	$f(x_1, x_2)$
R	R	1
R	B	7
B	R	3
B	B	2

Conditioning:

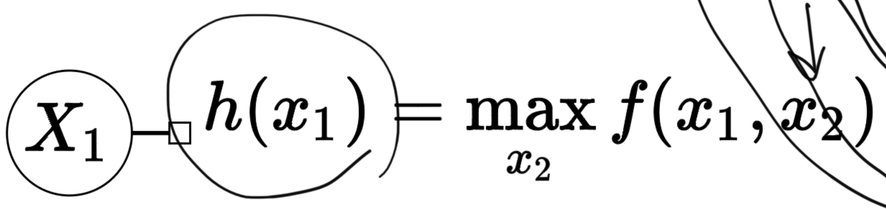
consider **one** value ($X_2 = \mathrm{B}$)



x_1	$g(x_1)$
R	7
B	2

Elimination:

consider **all** values



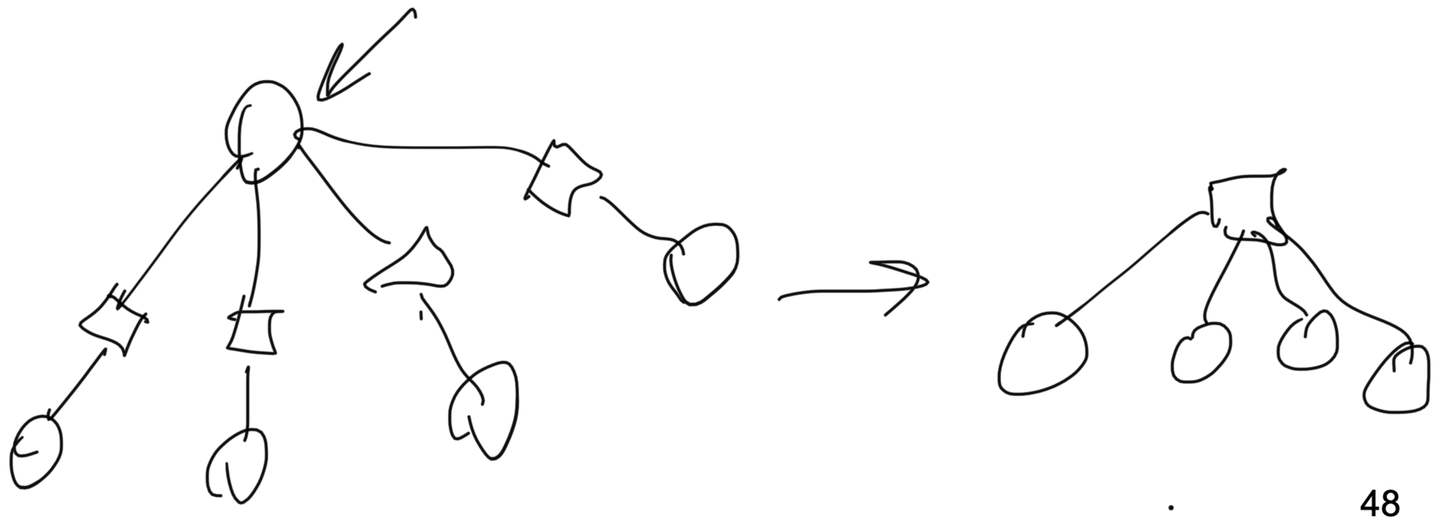
x_1	$h(x_1)$
R	7
B	3

Variable elimination algorithm

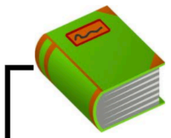
Algorithm: variable elimination

For $i = 1, \ldots, n$:
    Eliminate X_i (produces new factor $f_{\text{new},i}$).
For $i = n, \ldots, 1$:
    Set X_i to the maximizing value in $f_{\text{new},i}$.

[demo: `query(''); maxVariableElimination()`]



Treewidth



Definition: treewidth

The **treewidth** of a factor graph is the maximum arity of any factor created by variable elimination with the **best** variable ordering.

[whiteboard]

- Treewidth of a chain is 1.

Giant Sequoia

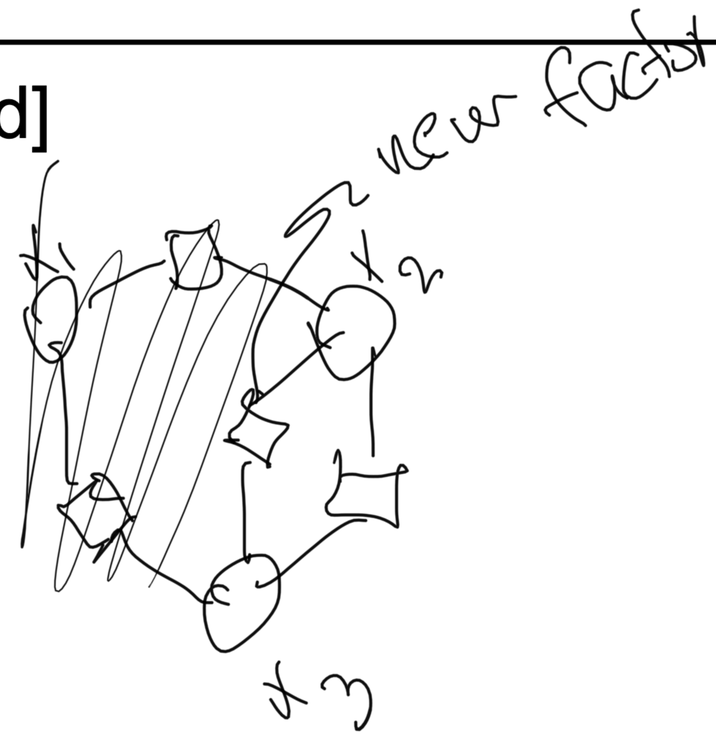
Treewidth

Definition: treewidth

The **treewidth** of a factor graph is the maximum arity of any factor created by variable elimination with the **best** variable ordering.

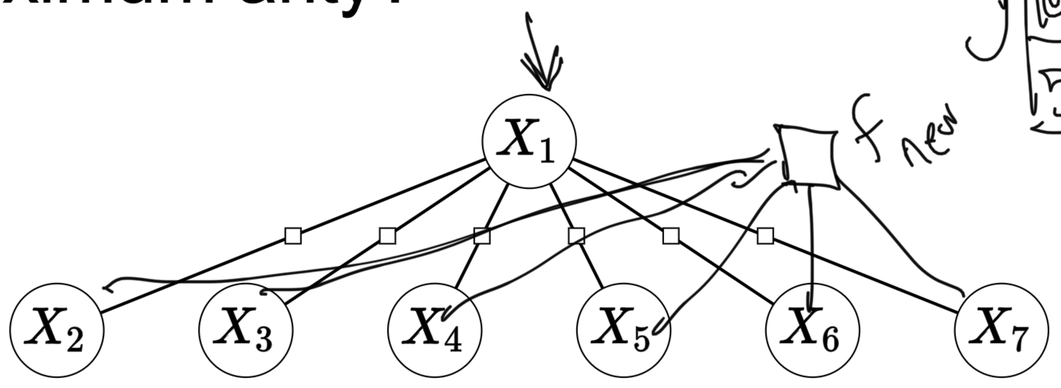
[whiteboard]

- Treewidth of a chain is 1.
- Treewidth of a tree is 1.
- Treewidth of simple cycle is 2.



Variable ordering

What's the maximum arity?



If eliminate leaves first, all factors have arity 1 (good)