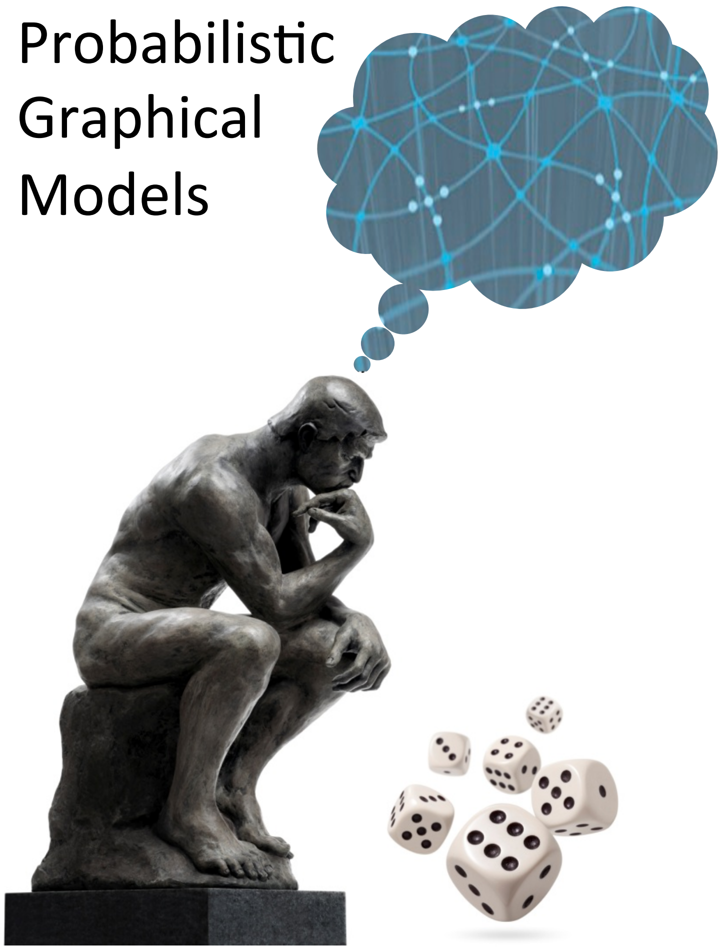
Probabilistic
Graphical
Models



Inference

Variable Elimination

Finding Elimination Orderings

Finding Elimination Orderings

- Theorem: For a graph H, determining whether there exists an elimination ordering for H with induced width ≤ K is NP-complete

- Note: This NP-hardness result is distinct from the NP-hardness result of inference
  - Even given the optimal ordering, inference may still be exponential

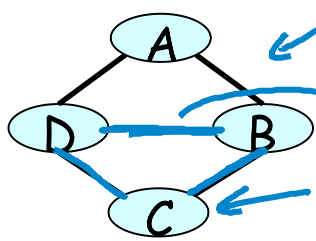
Finding Elimination Orderings

- Greedy search using heuristic cost function
  - At each point, eliminate node with smallest cost
- Possible cost functions:
  - min-neighbors: # neighbors in current graph
  - min-weight: weight (# values) of factor formed
  - min-fill: number of new fill edges
  - weighted min-fill: total weight of new fill edges (edge weight = product of weights of the 2 nodes)

smallest factor

Finding Elimination Orderings

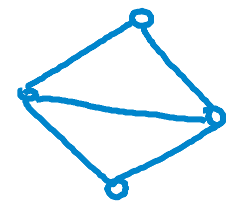
- Theorem: The induced graph is triangulated
  - No loops of length > 3 without a "bridge"



Assume we have a loop of size > 3

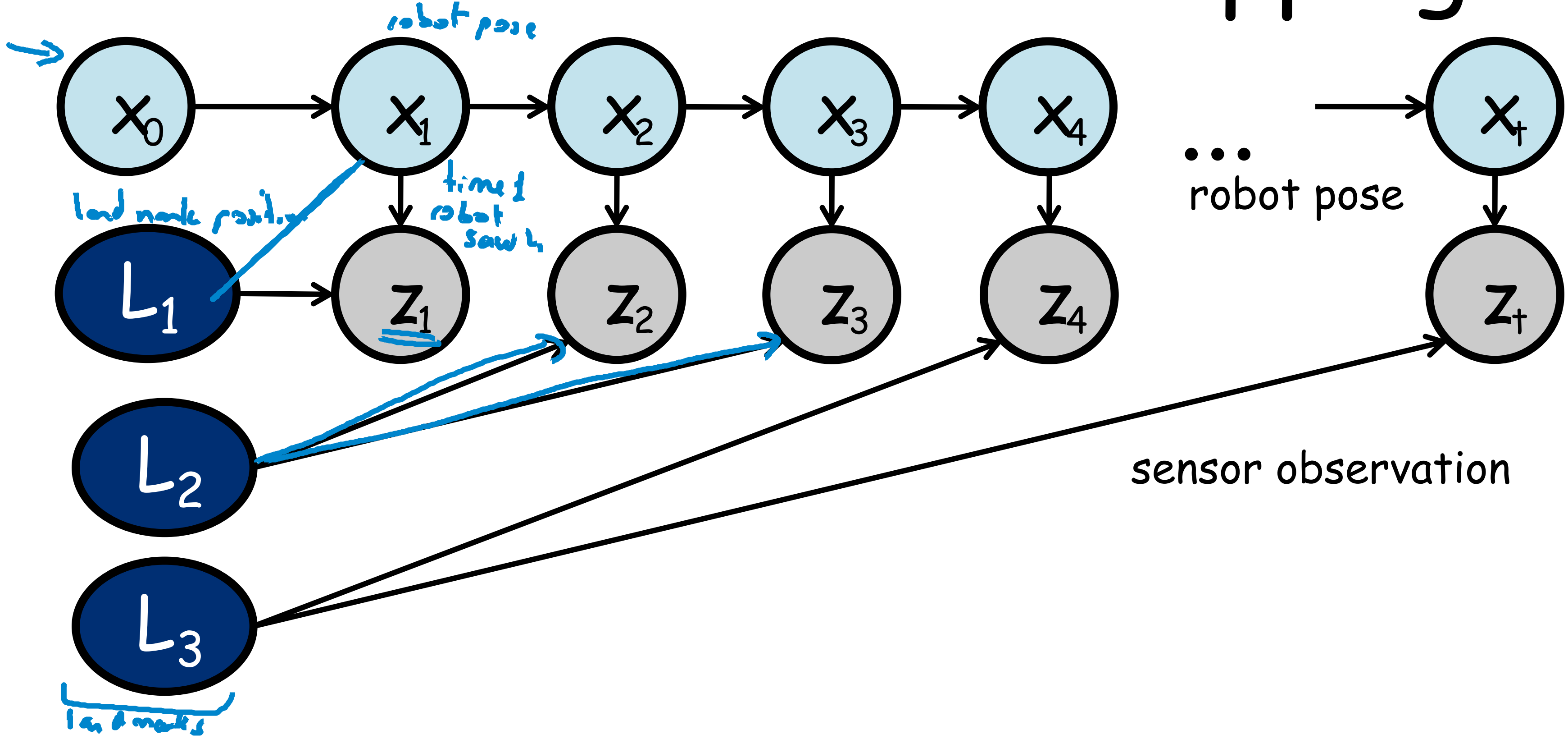
fill edge must be added

first to be eliminated



- Can find elimination ordering by finding a low-width triangulation of original graph H_Φ

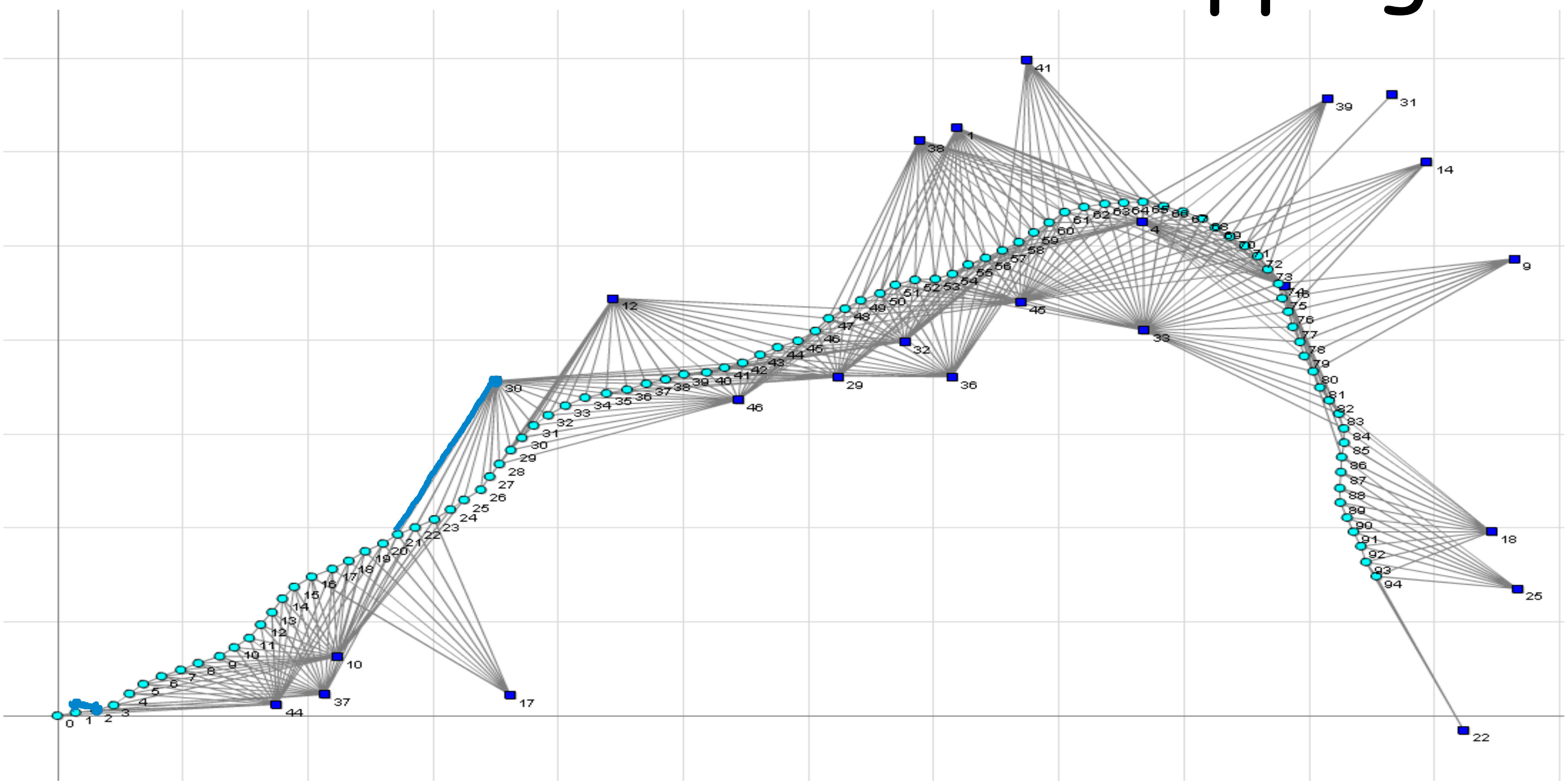
Robot Localization & Mapping



Daphne Koller

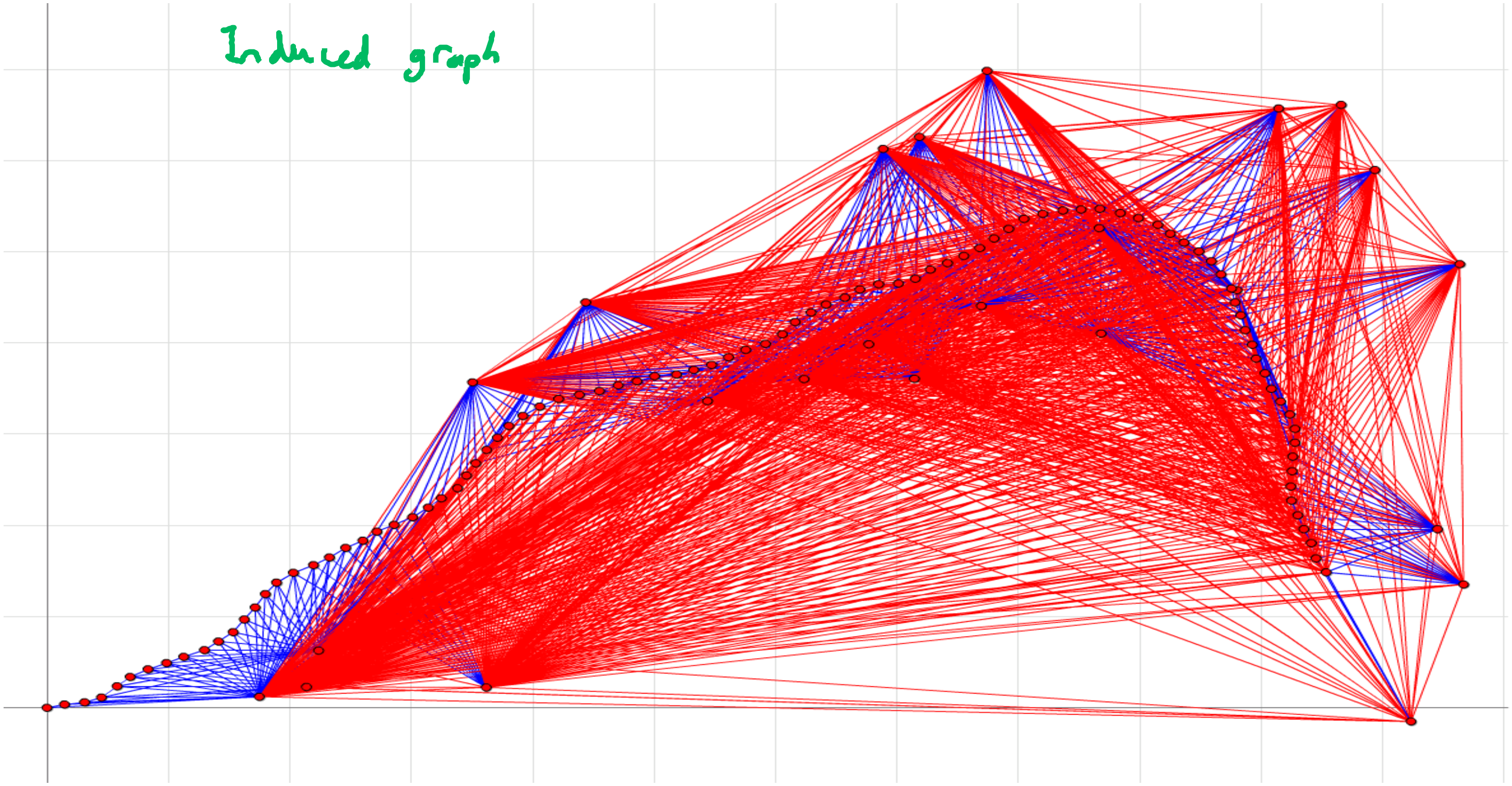
Square Root SAM, F. Dellaert and M. Kaess, IJRR, 2006

Robot Localization & Mapping



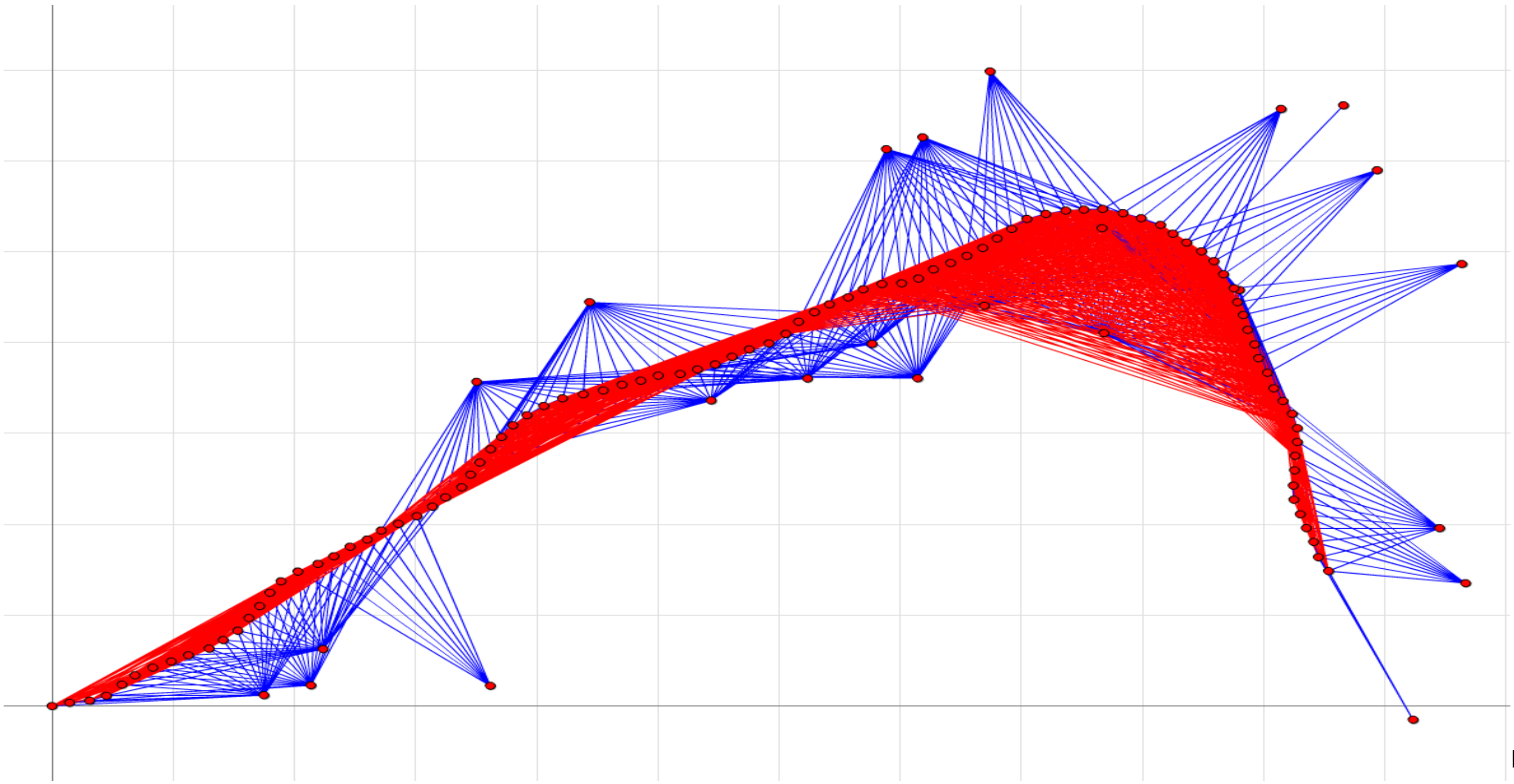
Square Root SAM, F. Dellaert and M. Kaess, IJRR, 2006

Eliminate Poses then Landmarks



Square Root SAM, F. Dellaert and M. Kaess, IJRR, 2006

Eliminate Landmarks then Poses



hne Koller

Summary

- Finding the optimal elimination ordering is NP-hard
- Simple heuristics that try to keep induced graph small often provide reasonable performance