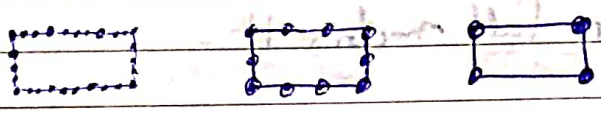
(5)

Hierarchical Pose-Graph SLAM for Online Mapping

Page No.
Date: / /

★ Hierarchical Pose-Graph

"There is no need to optimize the whole graph whenever a new observation is obtained"



Bottom layer (Input data) | First Layer | Second Layer | Top Layer

⇒ SLAM front-end seeks for loop-closures.

↳ This requires to know in which part of the graph to search for data associations.

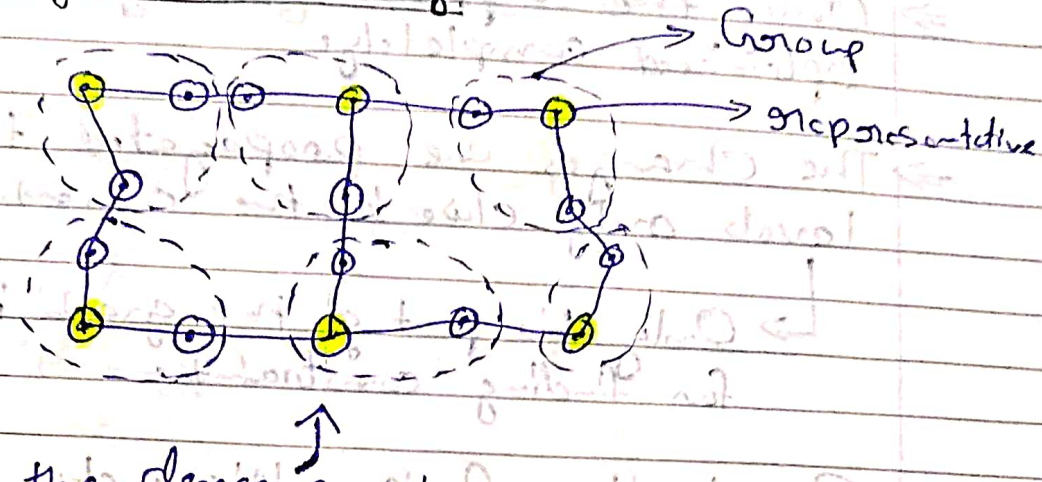
★ Hierarchical Approach

Insight: To find loop closing points, one does not need the perfect global map.

Idea: Connect only the core structure of the scene, not the overall graph.

Page No.
Date: / /

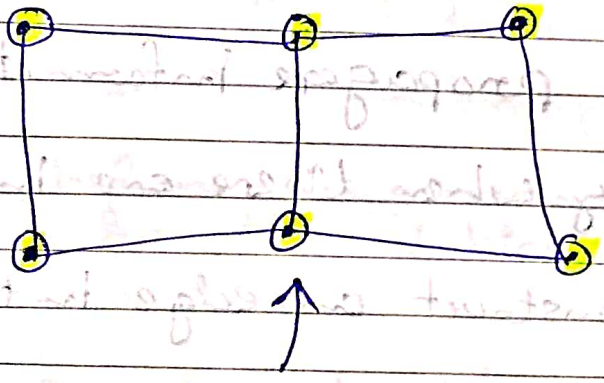
★ Key Idea of the Hierarchy:



⇒ Input is the dense graph.

⇒ Group the nodes of the graph based on their local connectivity.

⇒ For each group select one node as a representative.



⇒ The representatives are the nodes in a new sparsified graph (upper level)

⇒ Edges of the sparse graph are determined by the connectivity of the groups of nodes.

⇒ The parameters of the sparse edges are estimated via local optimization.

⇒ Only the upper level of the hierarchy is optimized completely.

⇒ The changes are propagated to the bottom levels only close to the current robot position
  ↳ Only this part of the graph is relevant for finding constraints.

★ Construction of the Hierarchy

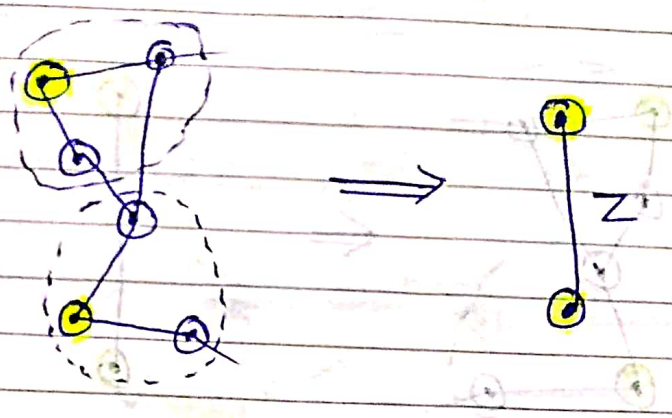
① When and how to generate a new group?
  ↳ A (Simple) distance-based decision
  ↳ The first node of a new group is the representative.

② When to propagate Information downwards?
  ↳ Only when there are inconsistencies.

③ How to Construct an edge in the Sparsified graph?
  TODO

④ How to propagate information downwards?
  TODO

★ Determining edge parameters



⇒ How to compute a virtual observation Z and the information matrix Ω for the new edge?

⇒ Optimize the two sub groups independently from the rest.

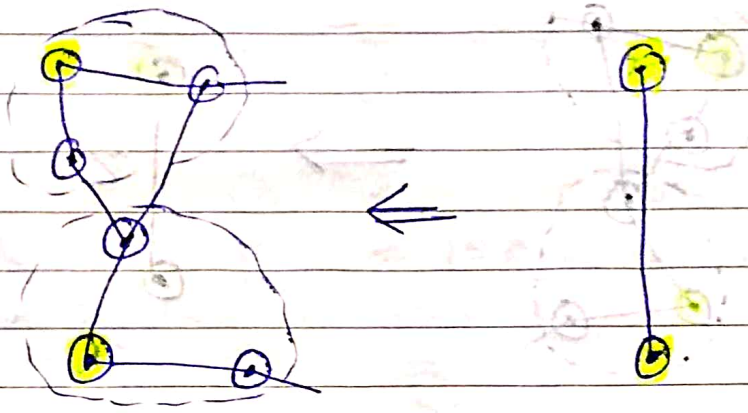
- → The observation is the relative transformation between the two representatives.
- → The information matrix is computed from the diagonal block of the matrix H:

$$\Omega_{ab} = (H^{-1}_{[b,b]})^{-1}$$

↓

{Inverse of the [b,b] block of H^{-1}}

★ Propagating Information Downwards



⇒ All representatives are nodes from the lowest (bottom) level

⇒ Information is propagated downwards by transforming the group at the lower level using a rigid body transformation.

⇒ Only if the lower level became inconsistent optimize at the lower level.

★ For the Best Possible Map

⇒ Run the optimization on the lowest level (at the end)

⇒ For offline processing with all constraints the hierarchy helps convergence faster in case of large error.