Optimal Edge Fault-Tolerant Embedding of a Star over a Cycle

Tadashi Akagi

Joint work with
Eduardo A. Canale,
Claudio E. Risso

Research work supported by Facultad Politécnica,
Universidad Nacional de Asunción
Problem and contributions

• Problem
  – Optimal Edge Fault-Tolerant Embedding of a Star over a Cycle

• Contributions
  – We find an equivalent problem, and
  – compute the optimal cost.
Outline

• Problem: Optimal Edge Fault-Tolerant Embedding of a Star over a Cycle.
  – Embedding
  – Demand, logical and physical graph
  – Edge Fault-tolerance

• Equivalent problem

• Optimal cost of the embedding
  – Lower bound
  – Upper bound
● Problem: Optimal Edge Fault-Tolerant Embedding of a Star over a Cycle.
  – Embedding
  – Demand, logical and physical graph
  – Edge Fault-tolerance

● Equivalent problem

● Optimal cost of the embedding
  – Lower bound
  – Upper bound
Embedding

- Edge $\{x, y\}$ associated to $x$-$y$ path

- The dilatation is the length of the path
Embedding (cont’d)

- If we embed every edge in $G$ to $H$, we have an embedded graph.
Embedding (cont’d)

- Embedding is a function

G’s edge  H’s path
Fiber optic network

- A fiber-optic communication can be modeled using 3 embedded graphs.
  - The first graph is a physical graph,
  - second graph is a logical graph, and
  - third graph is a demand graph.
Demand graph and Logical graph

- Given a number $b$, we say a logical graph satisfies the demand graph for that $b$ if there exist an embedding $\rho$ such that:

  - The number of demand edges passing through a logical edge must not exceed the constraint number $b$.

  - Number of demand edges passing through a logical edge is the congestion

\[
\text{cong}(\{x, y\}) \leq b
\]
Given a number $b$, we say a logical graph satisfies the demand graph for that $b$ if there exist an embedding $\rho$ such that:

- The number of demand edges passing through a logical edge must not exceed the constraint number $b$.
- Number of demand edges passing through a logical edge is the congestion $cong(\{x_a y\}) \leq b$.
Logical graph and Physical graph

- Like a “virtual” network over a real network.
- The cost of this embedding is the sum of the dilatation of every logical edge.
• Removing a physical edge may delete multiple logical edges at the same time.
Edge Fault-Tolerance

• An embedded logical graph is **Edge Fault-Tolerant** if:
  
  – For every possible faulty edge in a physical graph, the remainder logical graph satisfies the demand graph.
We study the following problem:
- **Physical graph**: cycle graph $C_n$
- **Demand graph**: star graph $K_{1,n-1}$

Problem: construct an embedded logical graph where
- The graph is **edge fault-tolerant**, and
- the **embedding** is optimal
Outline

• Problem: Optimal Edge Fault-Tolerant Embedding of a Star over a Cycle.
  – Embedding
  – Demand, logical and physical graph
  – Edge Fault-tolerance

• Equivalent problem

• Optimal cost of the embedding
  – Lower bound
  – Upper bound
Equivalent problem

- Central node: center of demand graph.
- In every optimal solution of the first problem, there is no logical edge passing through the central node → can split the central vertex.
Equivalent problem (cont’d)

- Line version of the problem
  - Physical graph: path graph $P_{n+1}$
  - Demand graph: complete bipartite $K_{2,n-1}$

- The cut of physical edge will split the demand graph

- Problem: construct an embedded logical graph where
  - The graph is edge fault-tolerant, and
  - the embedding is optimal
Lower bound

- The worst case cut is near the central vertex
Outline

- Problem: Optimal Edge Fault-Tolerant Embedding of a Star over a Cycle.
  - Embedding
  - Demand, logical and physical graph
  - Edge Fault-tolerance

- Equivalent problem

- Optimal cost of the embedding
  - Lower bound
  - Upper bound
Lower bound

- The worst case cut is near the central vertex
Lower bound (cont’d)

- Worst case lower bound

Physical graph (n = 7, b = 3)

Demand graph (Embedded)
Lower bound (cont’d)

Physical graph \((n = 7, b = 3)\)

- **first lower bound:**

  - **worst case 1 lower bound:**

  - **worst case 2 lower bound:**

- **First lower bound:**

  \[
  \sum_{i=0}^{n-1} \max \left\{ \left\lfloor \frac{i}{b} \right\rfloor, \left\lfloor \frac{n-i-1}{b} \right\rfloor \right\} \leq C
  \]
Lower bound (cont’d)

- Better lower bound:
  - If $n \not\equiv 2 \ (mod \ b)$
    \[
    \sum_{i=0}^{n-1} \max \left\{ \left\lfloor \frac{i}{b} \right\rfloor, \left\lfloor \frac{n - i - 1}{b} \right\rfloor \right\} \leq C
    \]
  - If $n \equiv 2 \ (mod \ b)$
    \[
    \sum_{i=0}^{n-1} \max \left\{ \left\lfloor \frac{i}{b} \right\rfloor, \left\lfloor \frac{n - i - 1}{b} \right\rfloor \right\} + 2 \leq C
    \]
Upper bound

• We obtain a **tight upper bound**

• We construct an **algorithm** where
  – **Input** is the **number of vertices** $n$ and the **capacity** $b$
  – **Output** is an **optimal logical graph**

• Is divided in **two steps**:
  1) Generates two graphs
  2) Merge the generated graph
1) Generate two labeled tree grouped by \( b \) vertices and connected to central vertex \( v_1 \) and \( v_{n+1} \).

2) Copy the edge with length 1 and merge short edge and large edges.
Upper bound (cont’d)

• An optimal solution! $n = 9$, $b = 3$
Thank you for your attention!