Optimal Verification of a Minimum-Weight Basis in an Uncertainty Matroid

Haya Diwan, Lisa Hellerstein
New York University

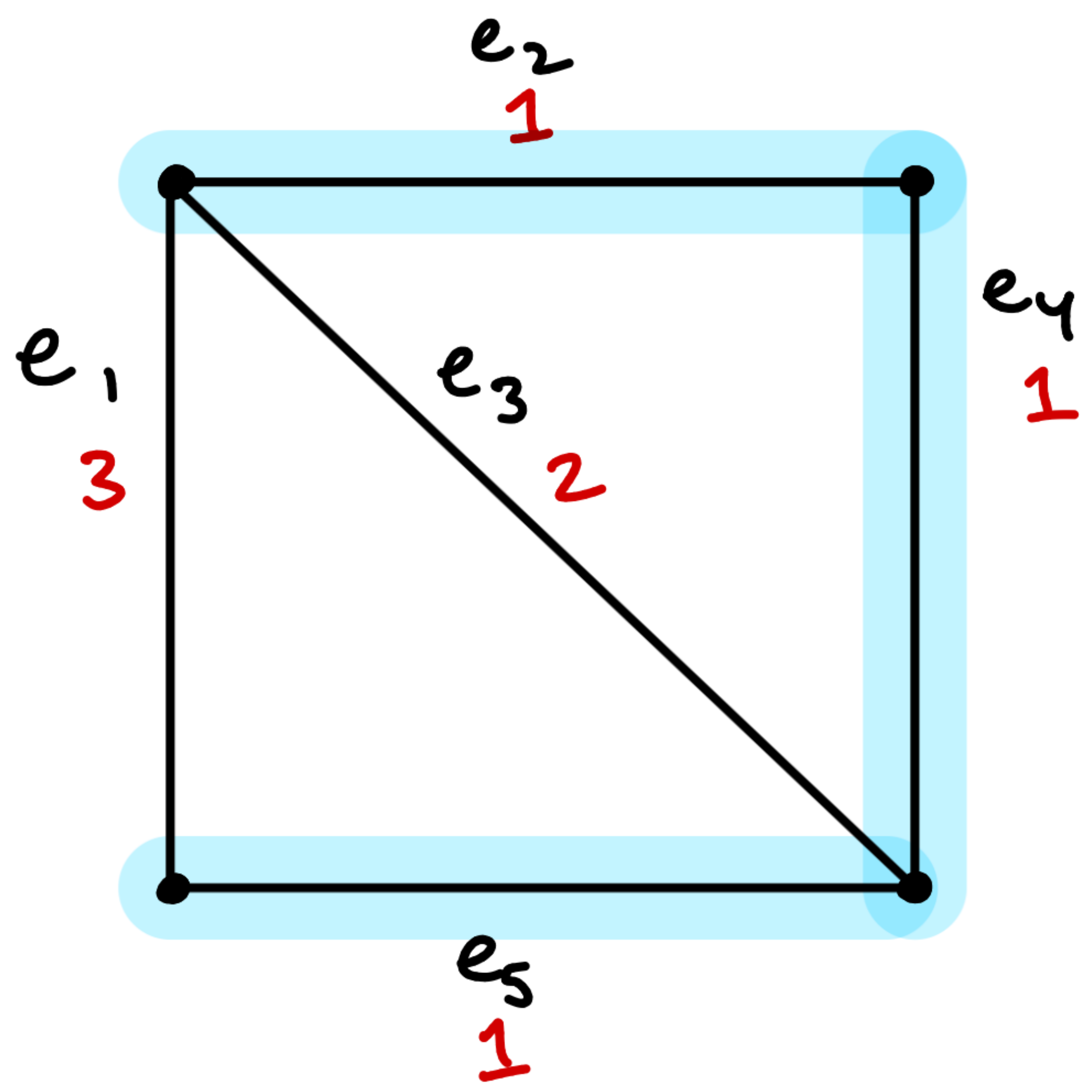
Nicole Megow
University of Bremen

Jens Schlöter
Centrum Wiskunde & Informatica (CWI)

Minimum Spanning Tree (MST) Problem

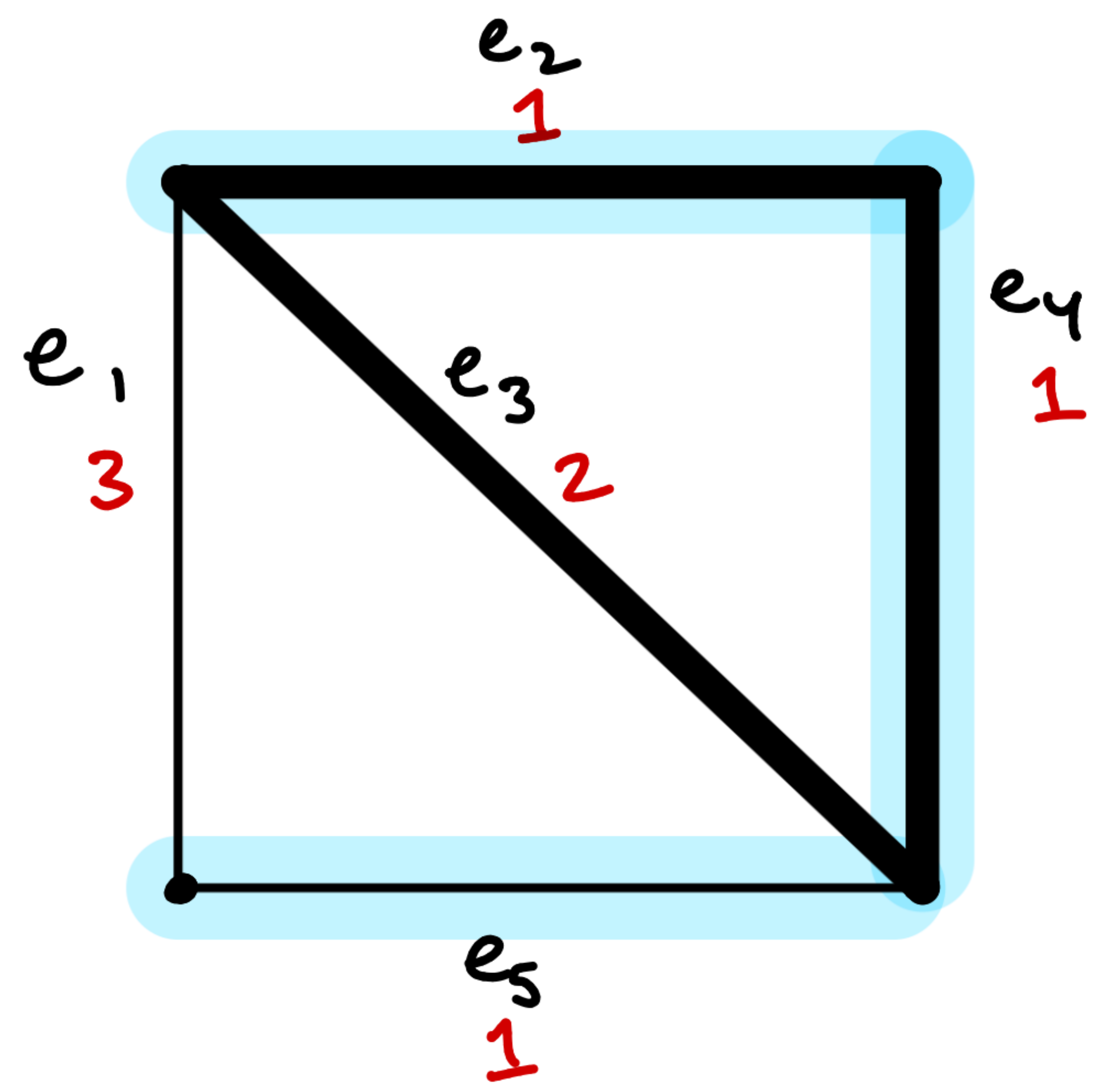
- Find an MST on the connected weighted graph $G = (V, E, w)$

- How can we easily verify that the highlighted edges form an MST?



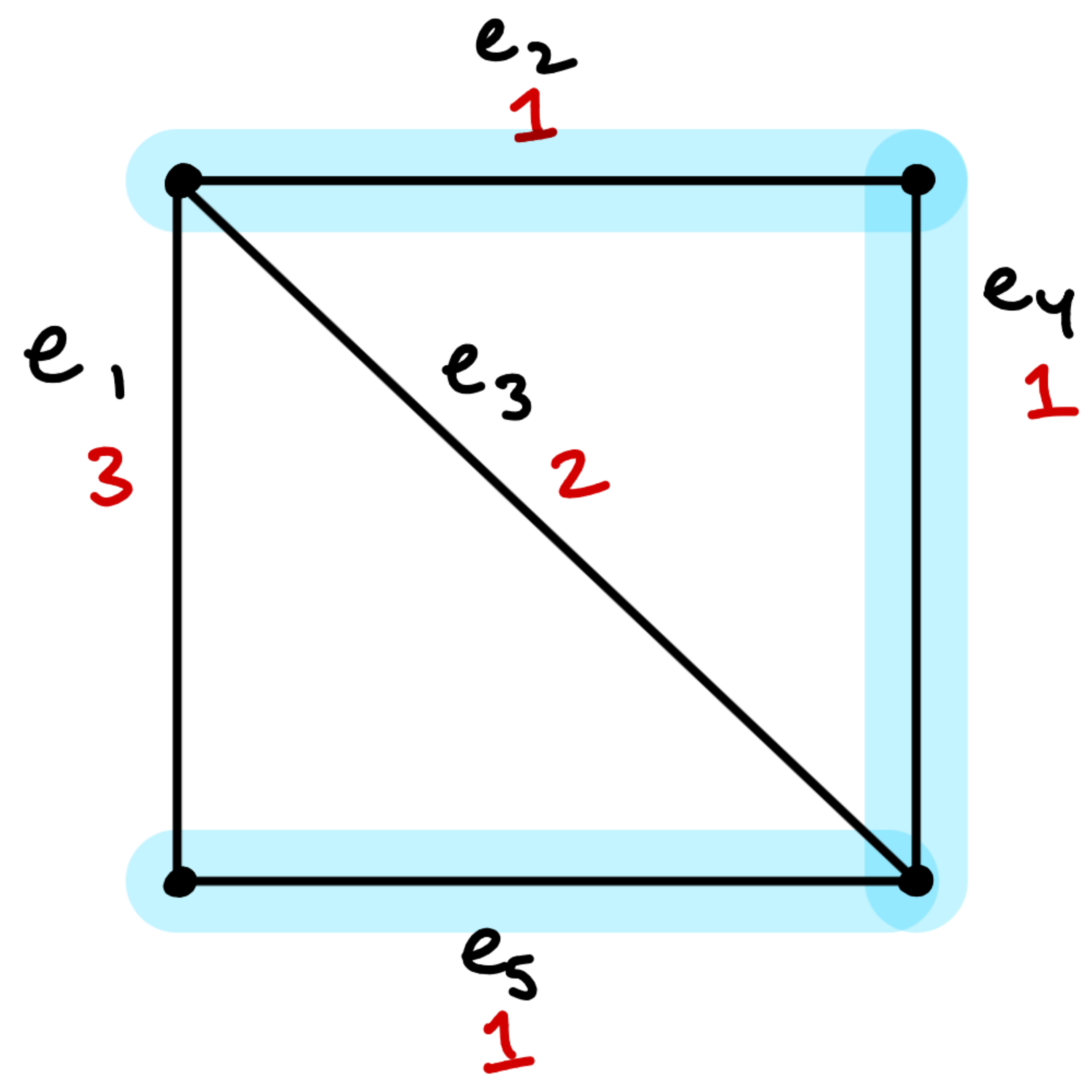
Minimum Spanning Tree (MST) Problem

- Find an MST on the connected weighted graph $G = (V, E, w)$
- How can we easily verify that the highlighted edges form an MST?
  - $e_3 = 2$ has maximum weight on cycle it forms with tree



Minimum Spanning Tree (MST) Problem

- Find an MST on the connected weighted graph $G = (V, E, w)$
- How can we easily verify that the highlighted edges form an MST?
  - $e_3 = 2$ has maximum weight on cycle it forms with tree



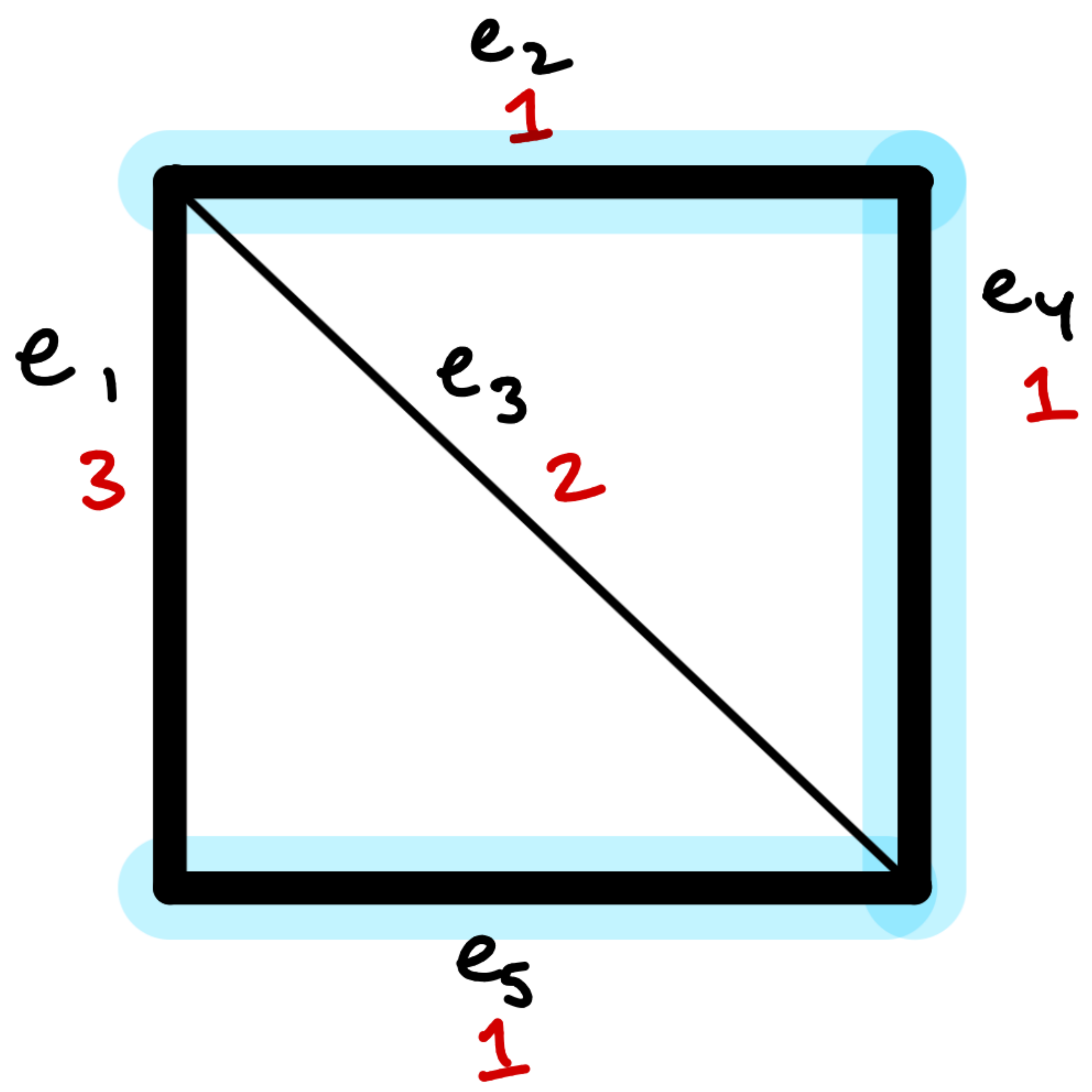
Minimum Spanning Tree (MST) Problem

- Find an MST on the connected weighted graph $G = (V, E, w)$

- How can we easily verify that the highlighted edges form an MST?

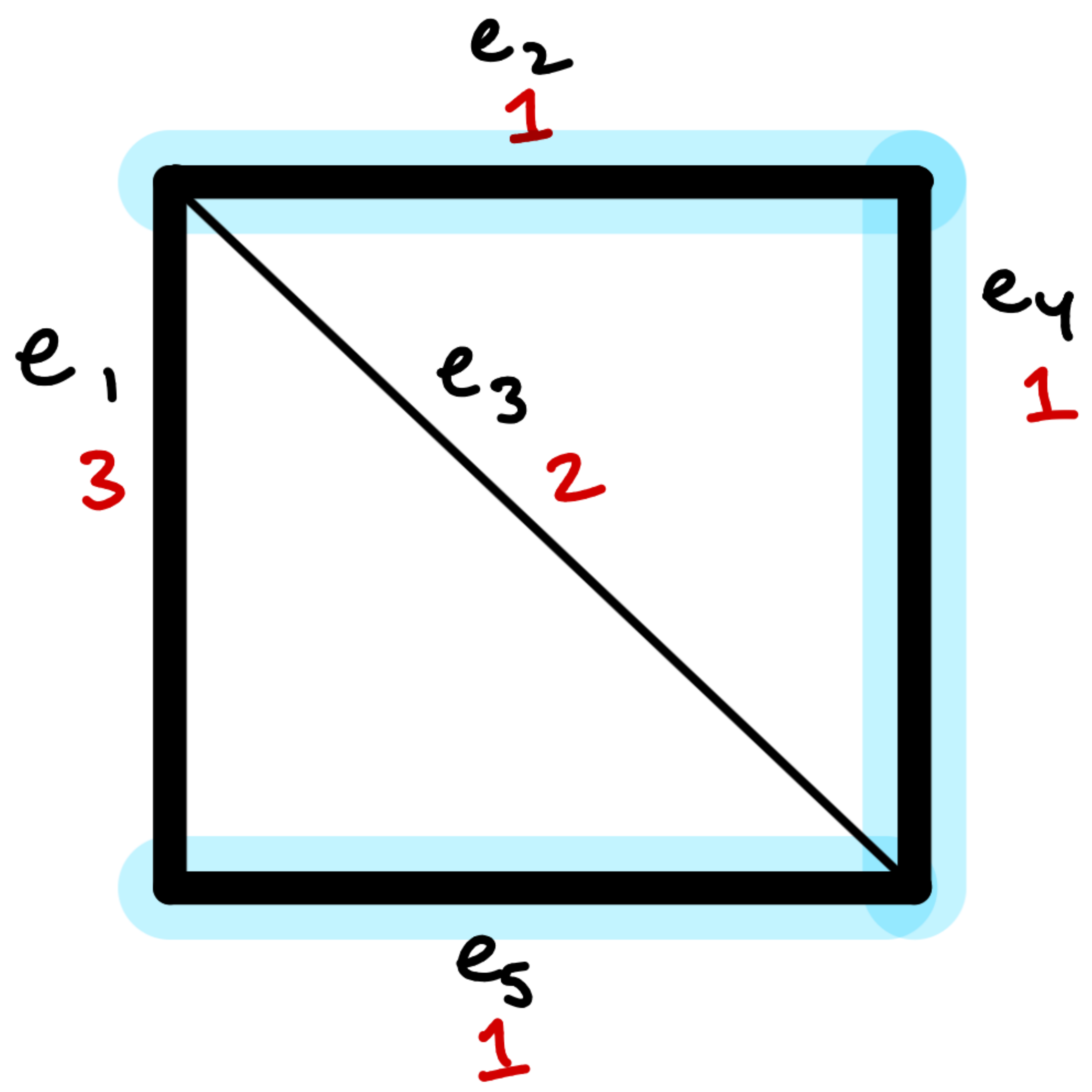
  - $e_3 = 2$ has maximum weight on cycle it forms with tree

  - $e_1 = 3$ has maximum weight on cycle it forms with tree



Minimum Spanning Tree (MST) Problem

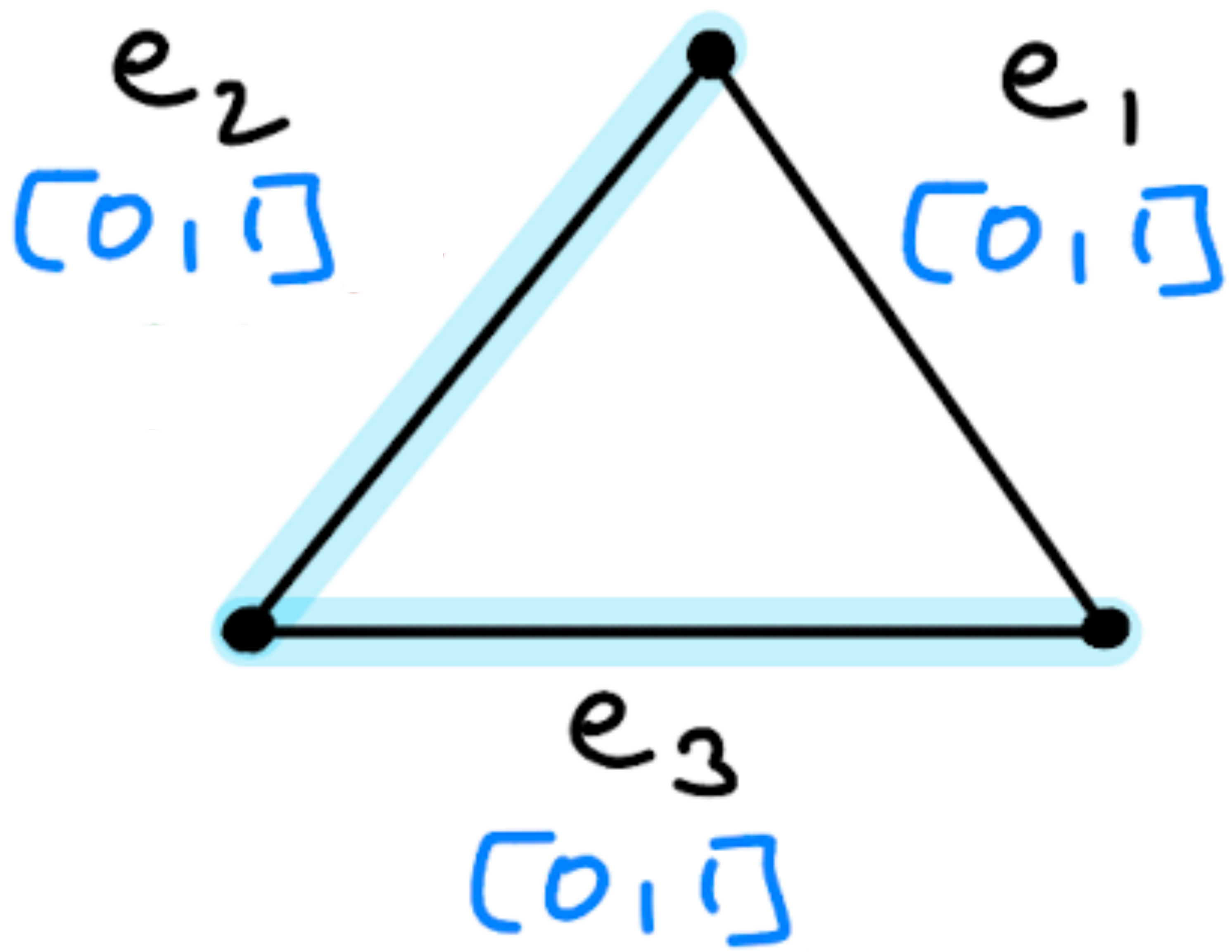
- Find an MST on the connected weighted graph $G = (V, E, w)$
- How can we easily verify that the highlighted edges form an MST?
  - $e_3 = 2$ has maximum weight on cycle it forms with tree
  - $e_1 = 3$ has maximum weight on cycle it forms with tree
  - Showed that each non-tree edge is a max-weight edge on the cycle it forms when added to the tree
    - Verified!



Problem Example

edge label

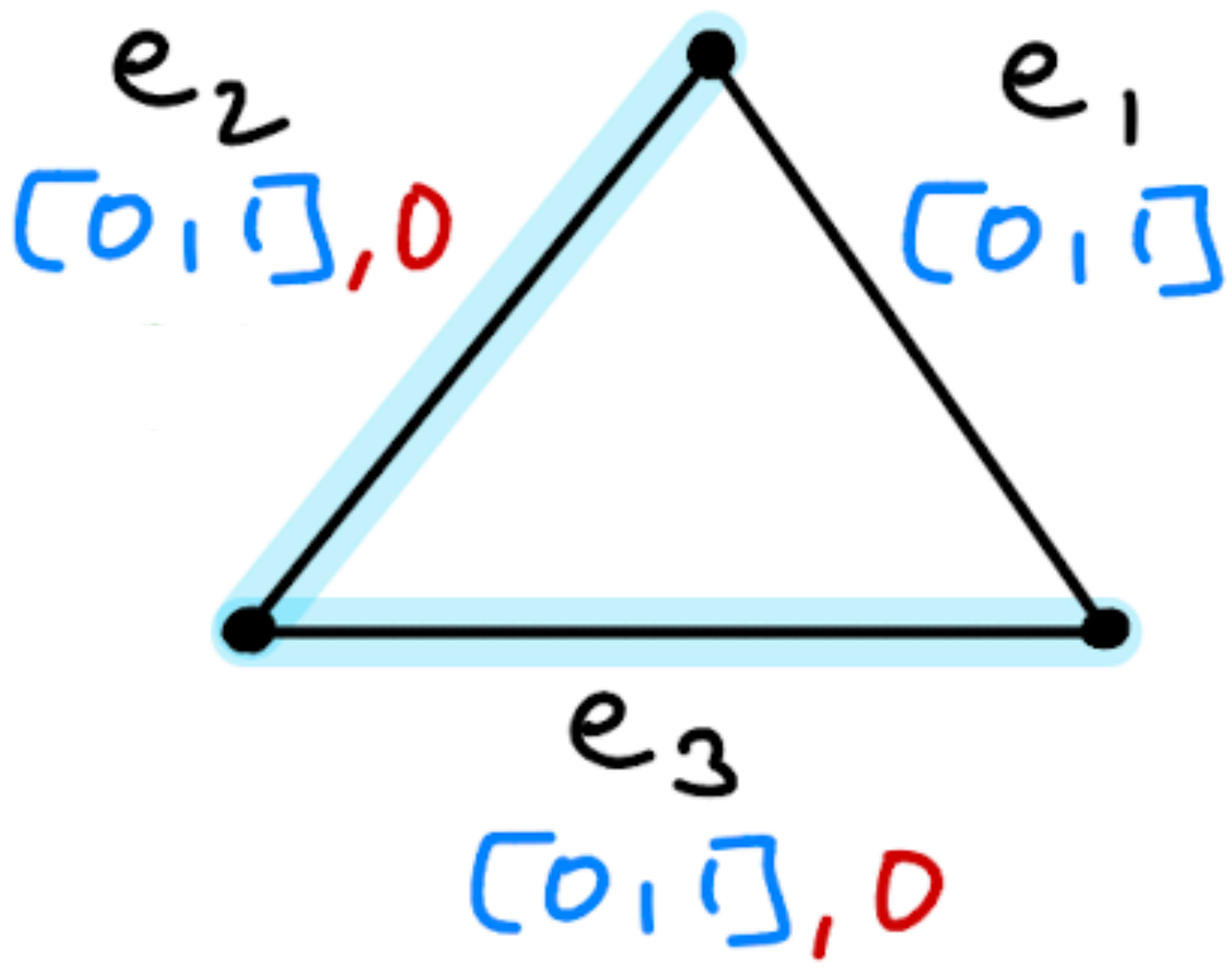
uncertainty area



Problem Example

edge label

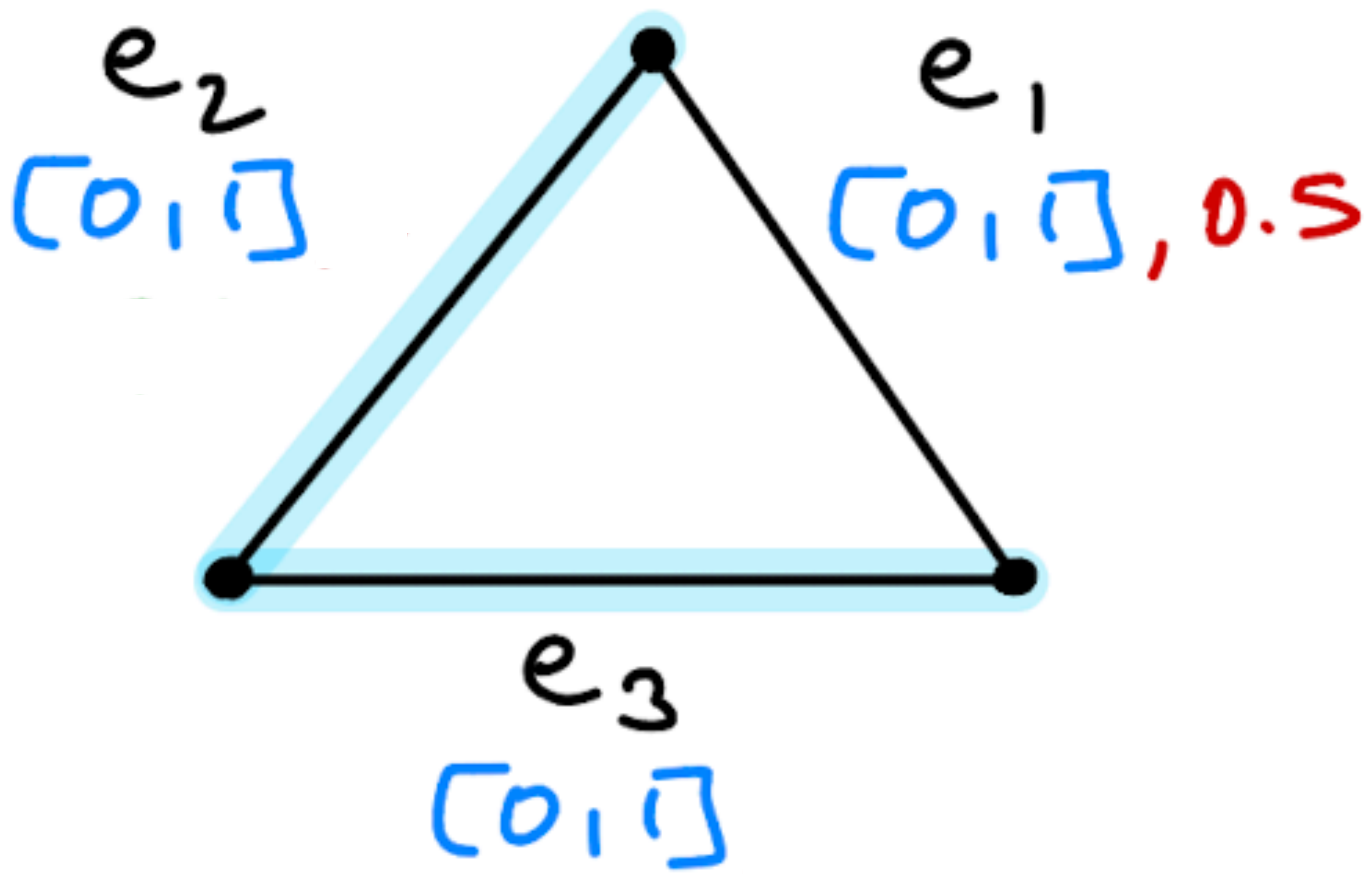
uncertainty area, edge weight



Problem Example

edge label

uncertainty area, edge weight

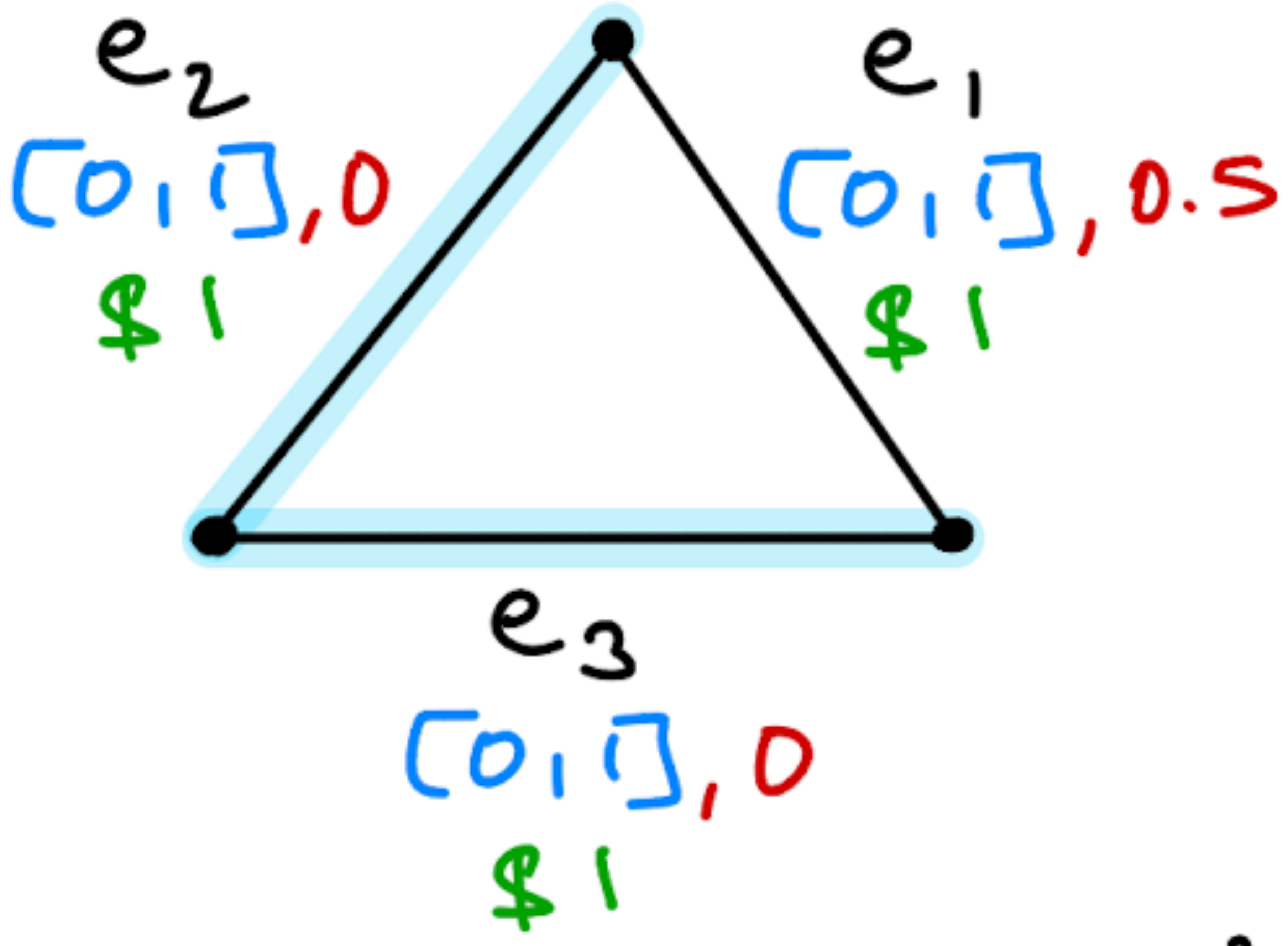


Problem Example

edge label

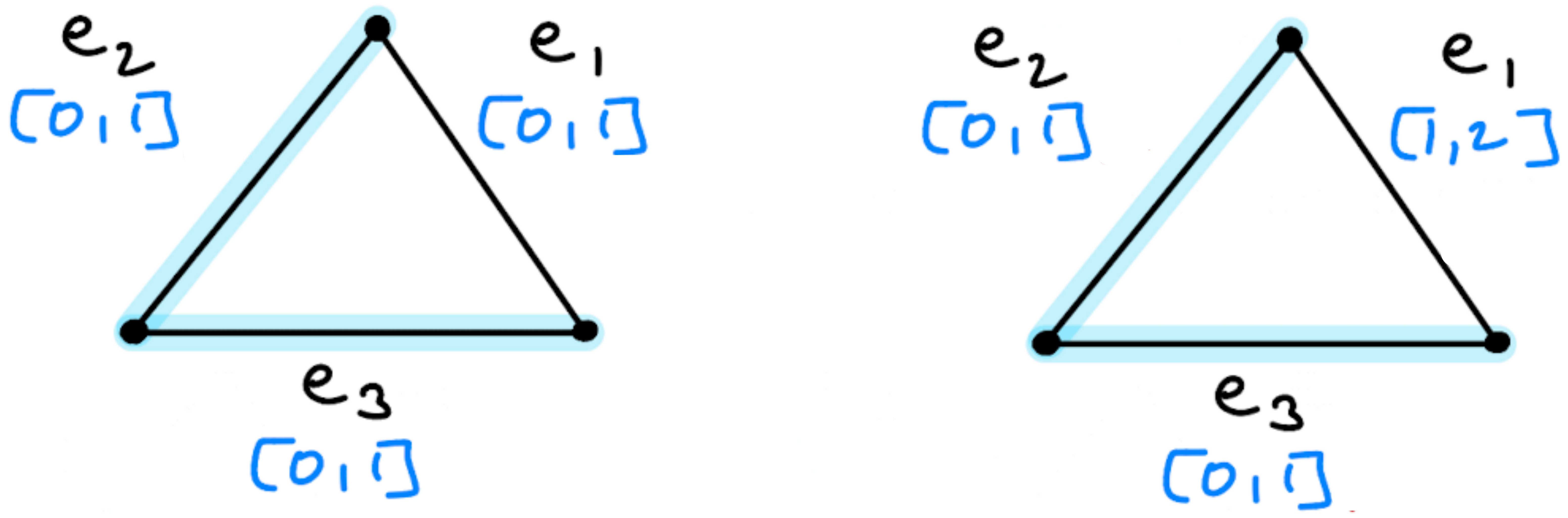
uncertainty area, edge weight

Cost to reveal weight

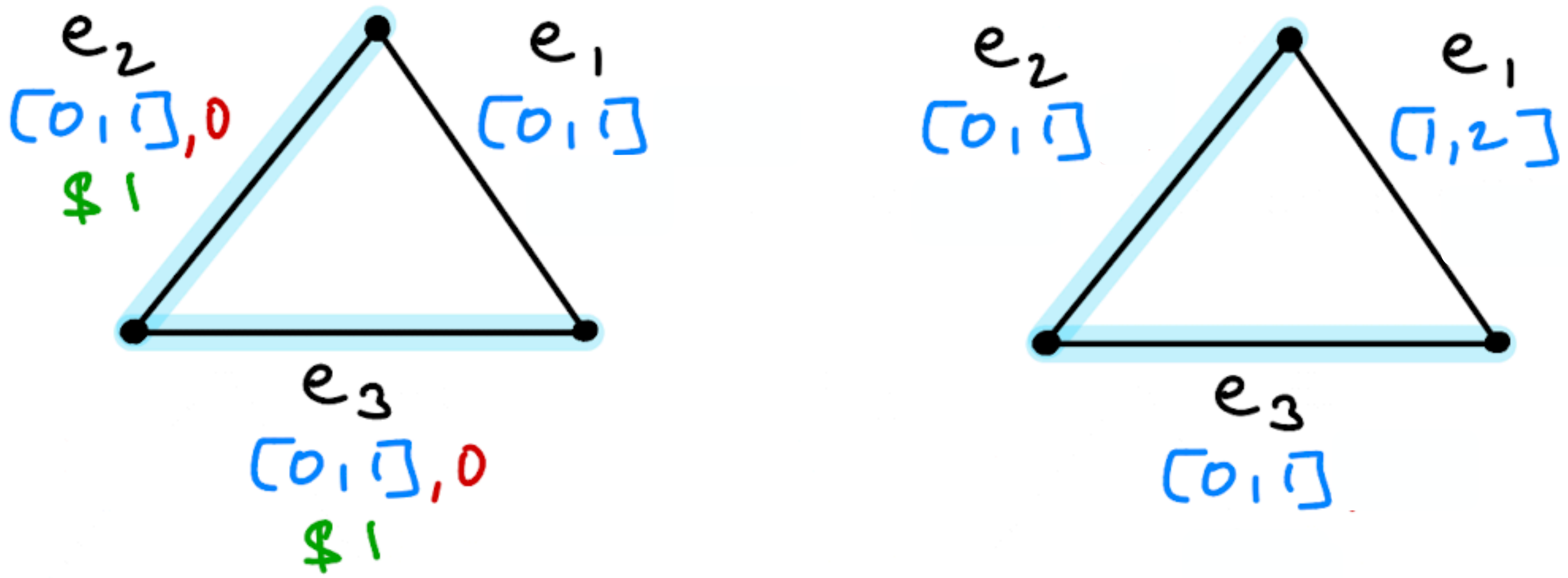


certificate : $\{e_2 = 0, e_3 = 0\}$

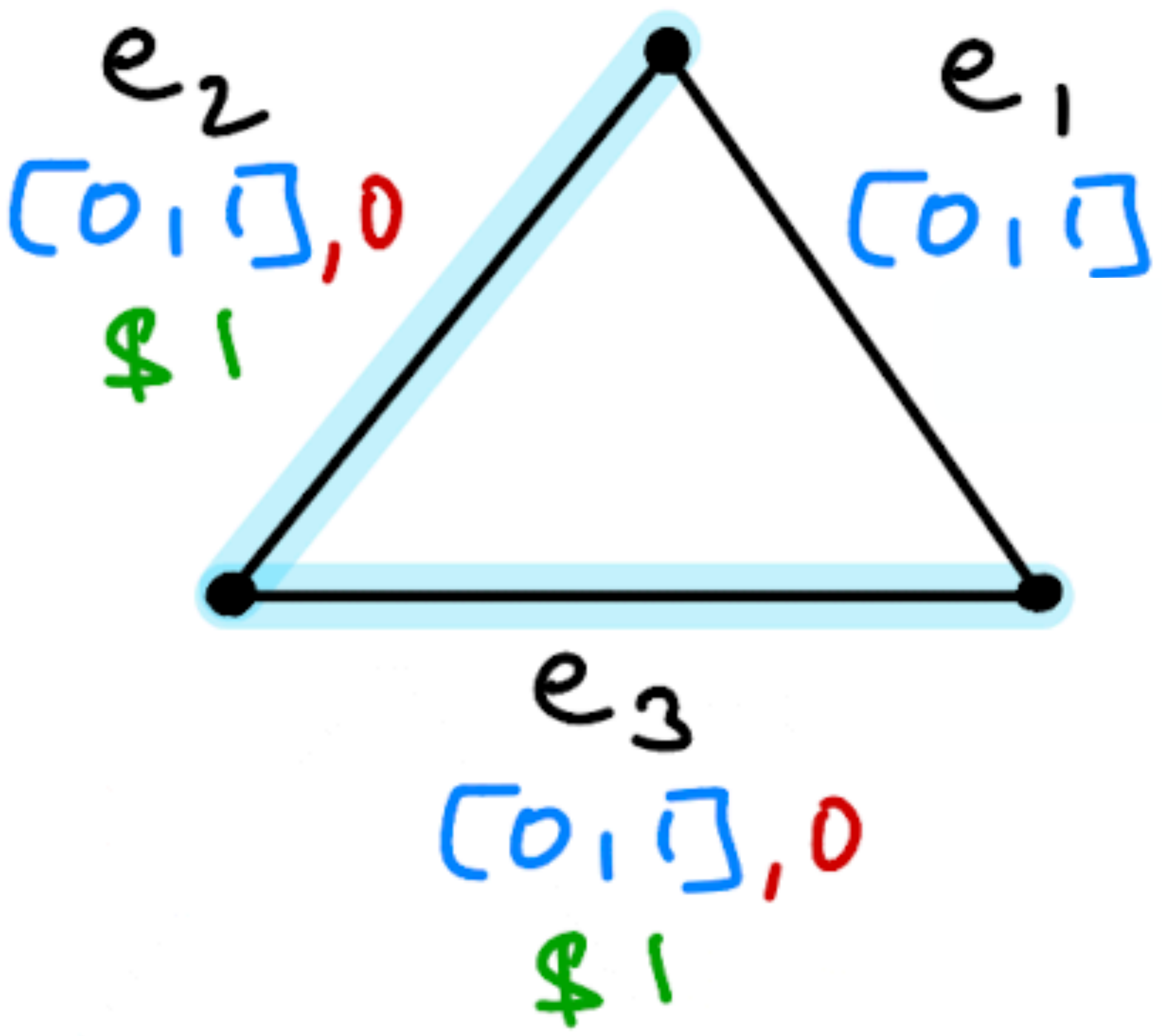
More Examples



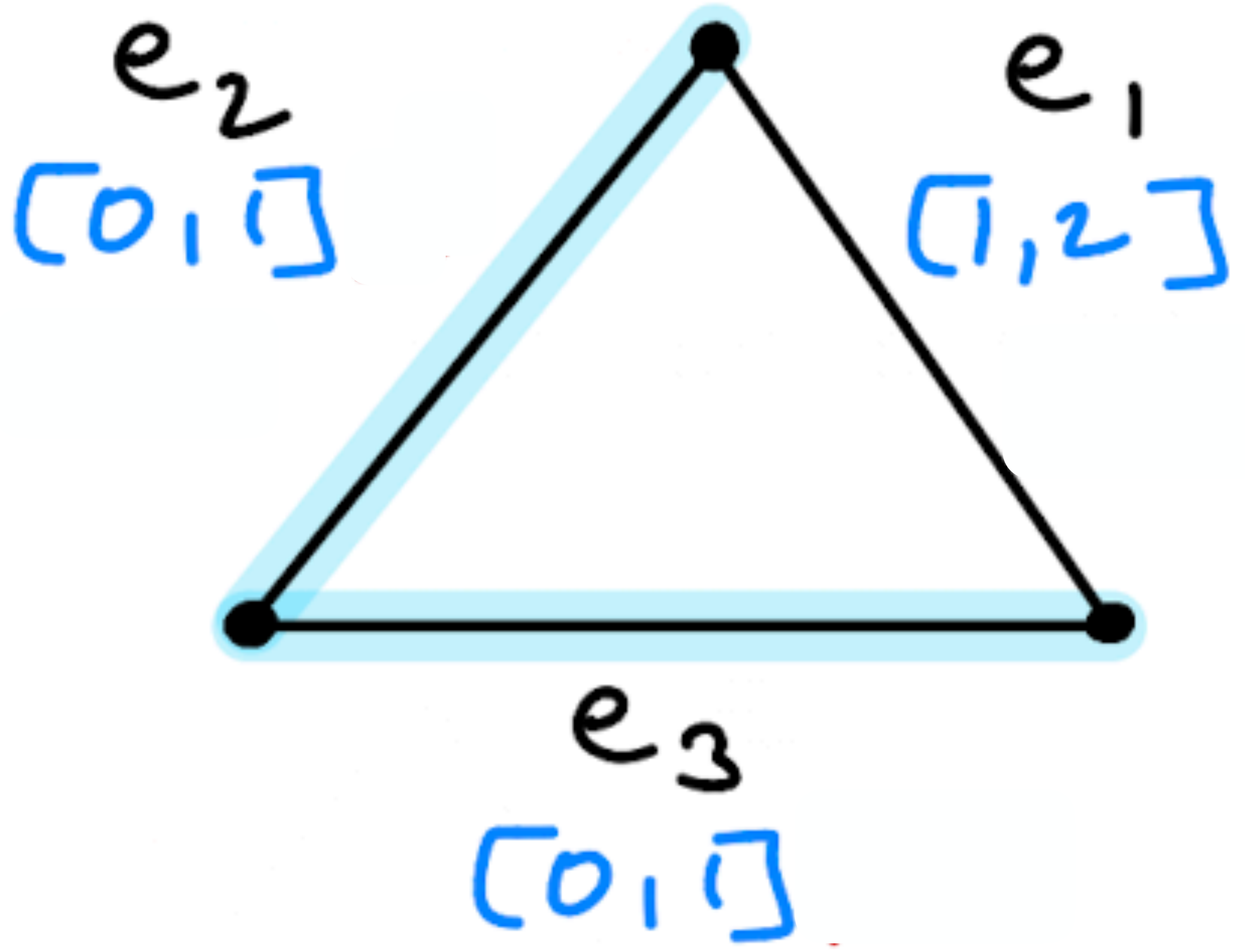
More Examples



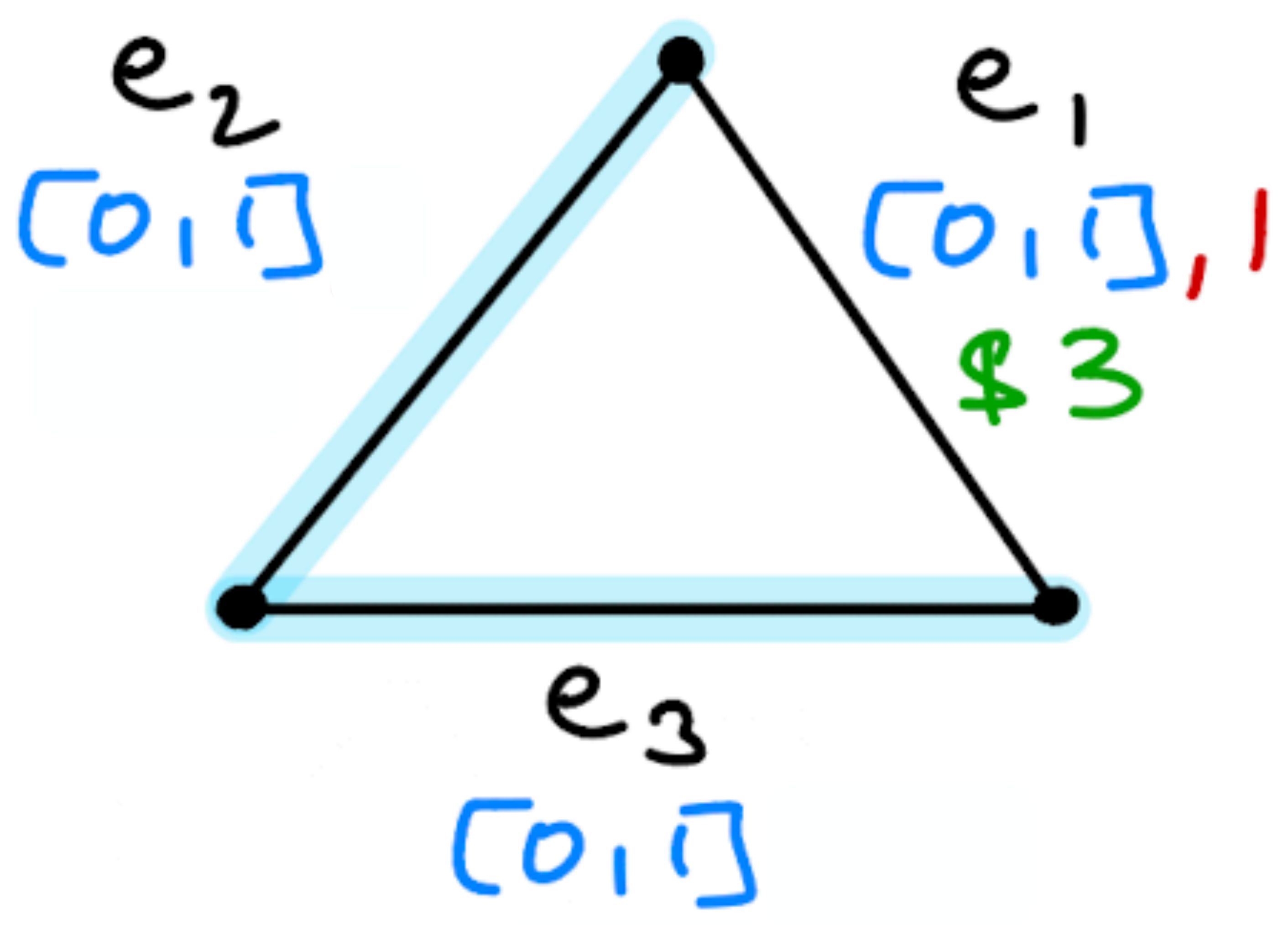
More Examples

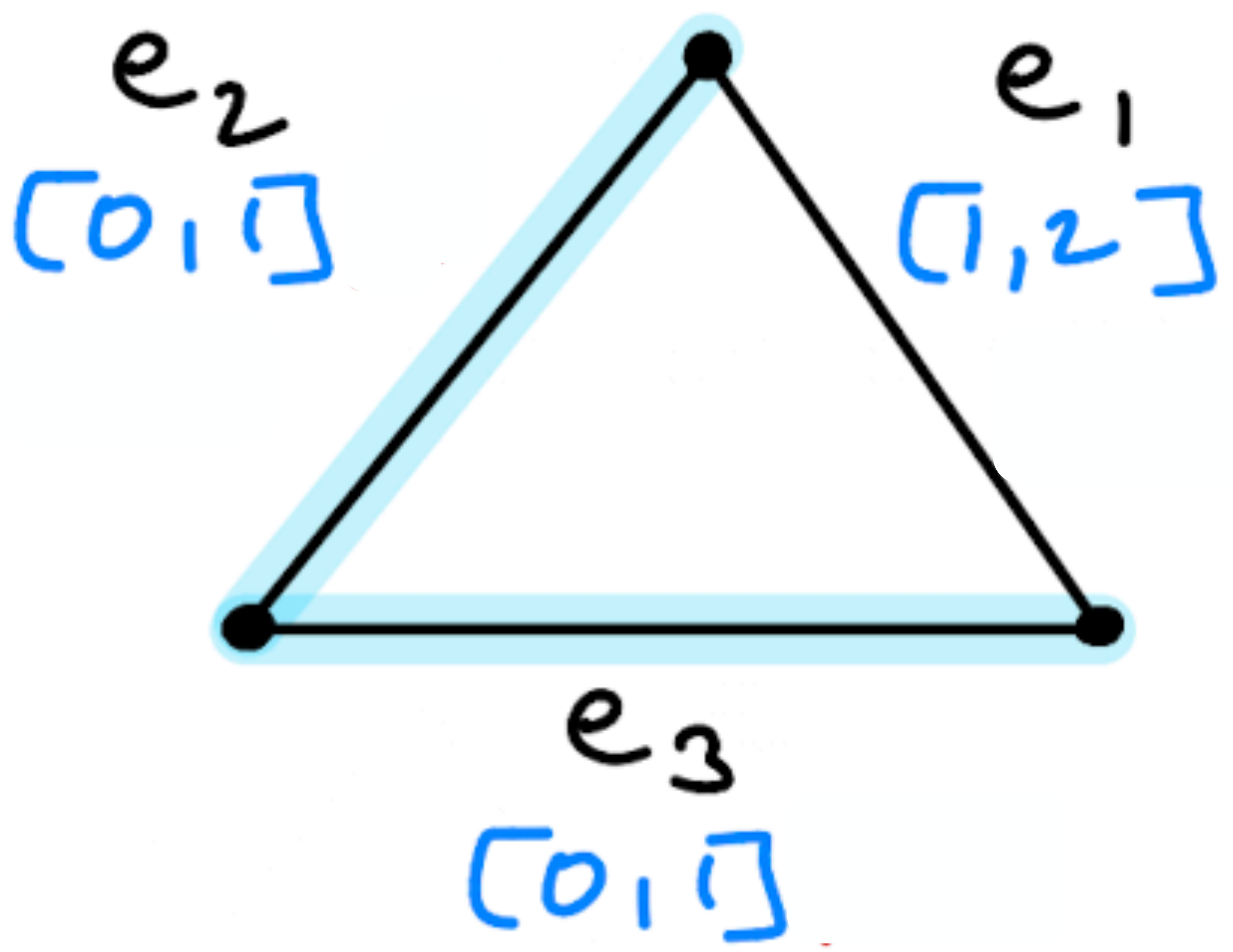


certificate : $\{e_2=0, e_3=0\}$



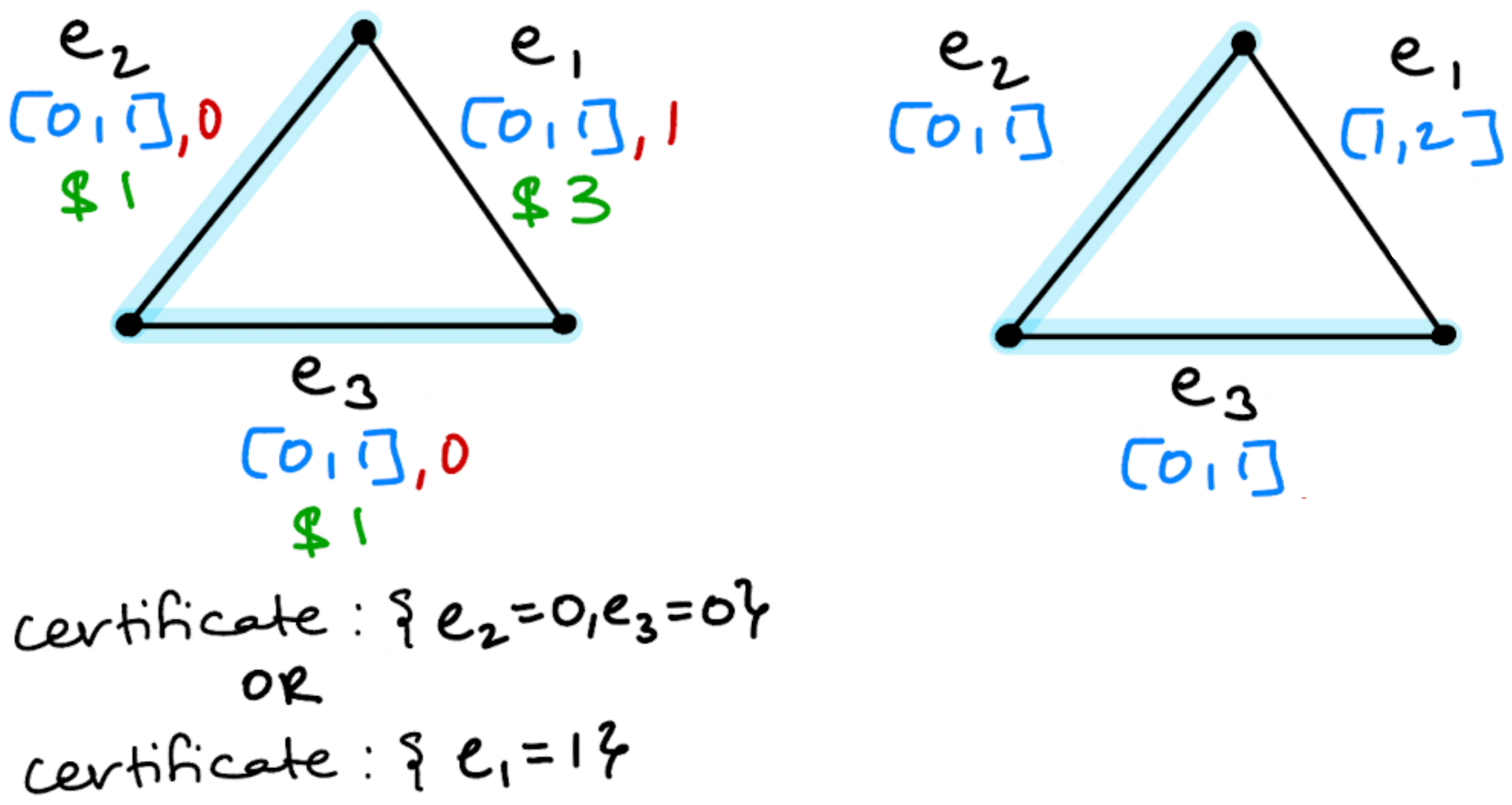
More Examples



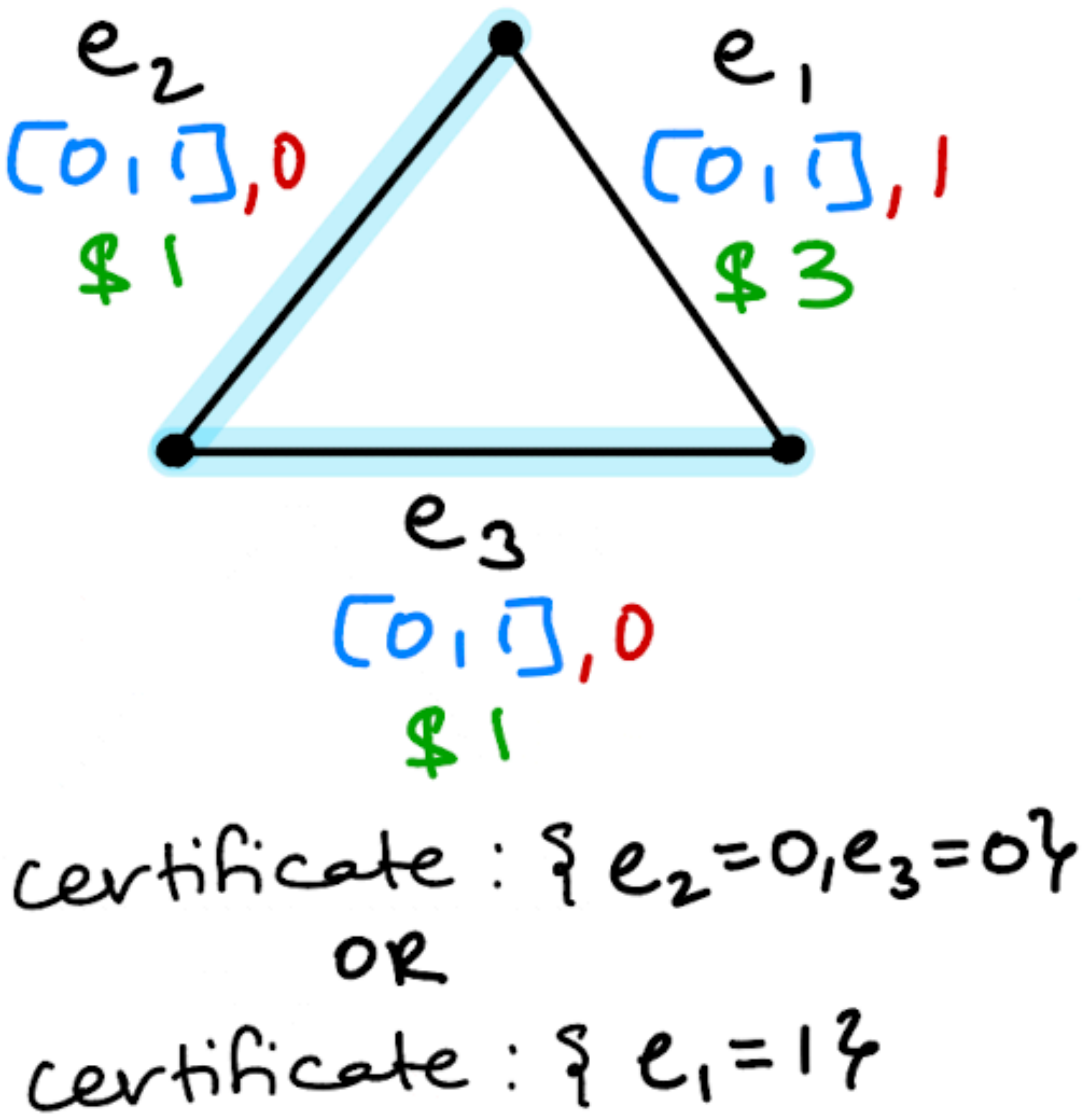


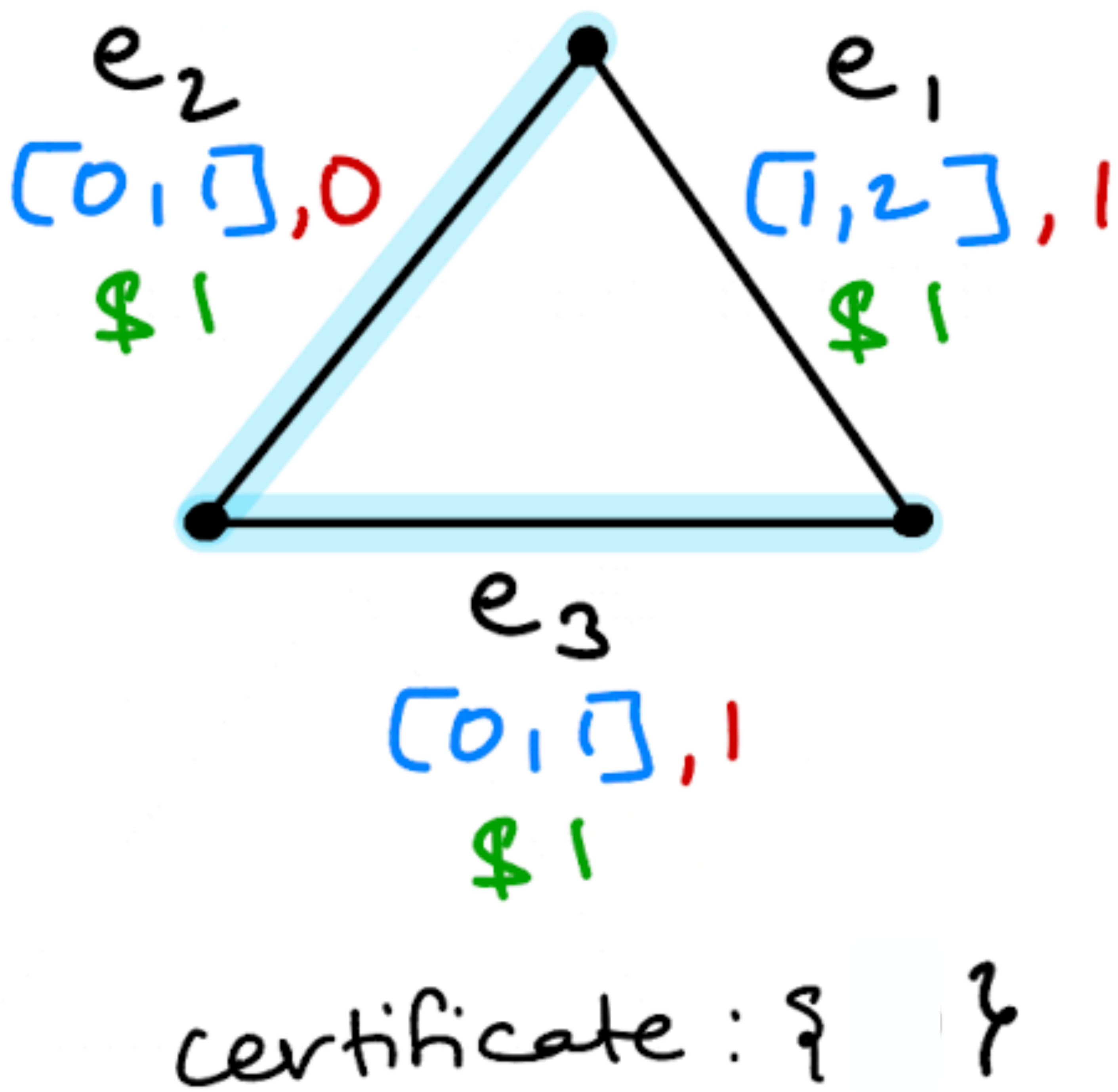
certificate : { $e_1 = 1$ }

More Examples



More Examples





The Verification Objective

- Find an MST T and an associated min-cost certificate Q
  - Q must verify that T is an MST assuming that for edges not in Q, only their uncertainty areas are known (not their actual weights)
  - T is a cheapest-to-verify MST among all MSTs of G
    - no other MST of G has a cheaper certificate

MST Verification Problem

- **Input:** Connected graph $G = (V, E)$, with the following values for each edge $e \in E$
  - uncertainty area $A_e = [L_e, U_e]$ (a closed real interval)
  - weight w_e such that $w_e \in A_e$
  - cost c_e such that $c_e \geq 0$
- **Output:** T, Q where
  - T is a cheapest-to-verify MST of G
  - Q is a min-cost certificate for T

Theorem: There is a polynomial-time algorithm that solves the MST verification problem

MWB Verification Problem

- **Input:** Matroid $M = (E, \mathcal{I})$, with the following values for each element $e \in E$
  - uncertainty area $A_e = [L_e, U_e]$ (a closed real interval)
  - weight w_e such that $w_e \in A_e$
  - cost c_e such that $c_e \geq 0$
- **Output:** B, Q where
  - B is a cheapest-to-verify MWB of M
  - Q is a min-cost certificate for B

Theorem: There is a polynomial-time algorithm that solves the MWB verification problem

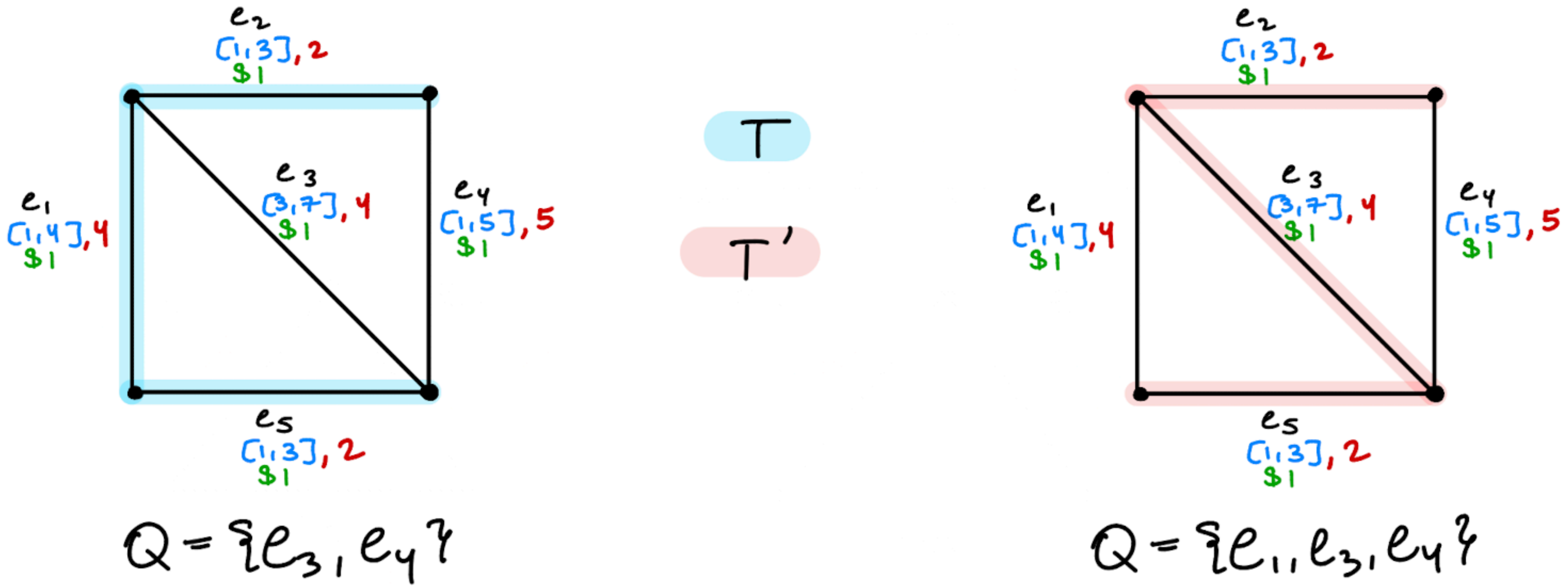
Related Work

- [Erlebach and Hoffman, 2014] gave a polynomial time algorithm for a related MST verification problem:
  - Open (L_e, U_e) uncertainty intervals
- The MST problem for closed uncertainty intervals (our problem) is significantly more challenging
  - Any algorithm that works for closed intervals also works for open intervals
  - With open intervals [Erlebach and Hoffman, 2014] all MSTs have the same verification cost
    - Can output arbitrary MST T, doesn't matter which one
  - With closed intervals different MSTs can have different verification costs!
    - Need to find a cheapest-to-verify MST

MST Verification in Online Settings

- **Worst case online setting:** Competitive ratio compares online query costs to a minimum-cost offline certificate
    - **Open Intervals:** 2-competitive ratio [Erlebach et. al, 2016]
    - **Closed Intervals:** Ratio degrades to n [Gupta et. al, 2016]
- In worst case online setting, we achieved 2-competitive ratio for closed intervals by providing the algorithm with a cheapest-to-verify MWB
    - Complexity in closed interval case stems from the search for a cheapest-to-verify MWB, not the search for its min-cost certificate
- Leveraged these structural insights to develop an algorithm that utilizes untrusted MWB predictions to beat worst-case bounds

Impact of Closed Intervals: Different Verification Costs



MST Verification Algorithm Overview

- **Phase 1: Compute a cheapest-to-verify MST T of G**
  - Contract or delete extreme case edges ($w_e = L_e$ or $w_e = U_e$) according to certain rules
    - Each rule application transforms the current graph into a smaller resulting graph: $G \to G' \to G'' \to \ldots \to G^*$
    - Let K be the set of contracted edges

    # Remaining edges in G^* satisfy $L_e < w_e < U_e$

    # Equivalent to just having open intervals as in [Erlebach and Hoffman, 2014]
  - Compute any MST T^* of G^*
  - $T := T^* \cup K$ is a cheapest-to-verify MST of G
- **Phase 2: Compute a min-cost certificate for T**
  - Reduce to a weighted bipartite vertex cover problem [Erlebach and Hoffman, 2014]
    - minimum weight vertex cover gives a min-cost certificate that verifies T

Phase 1

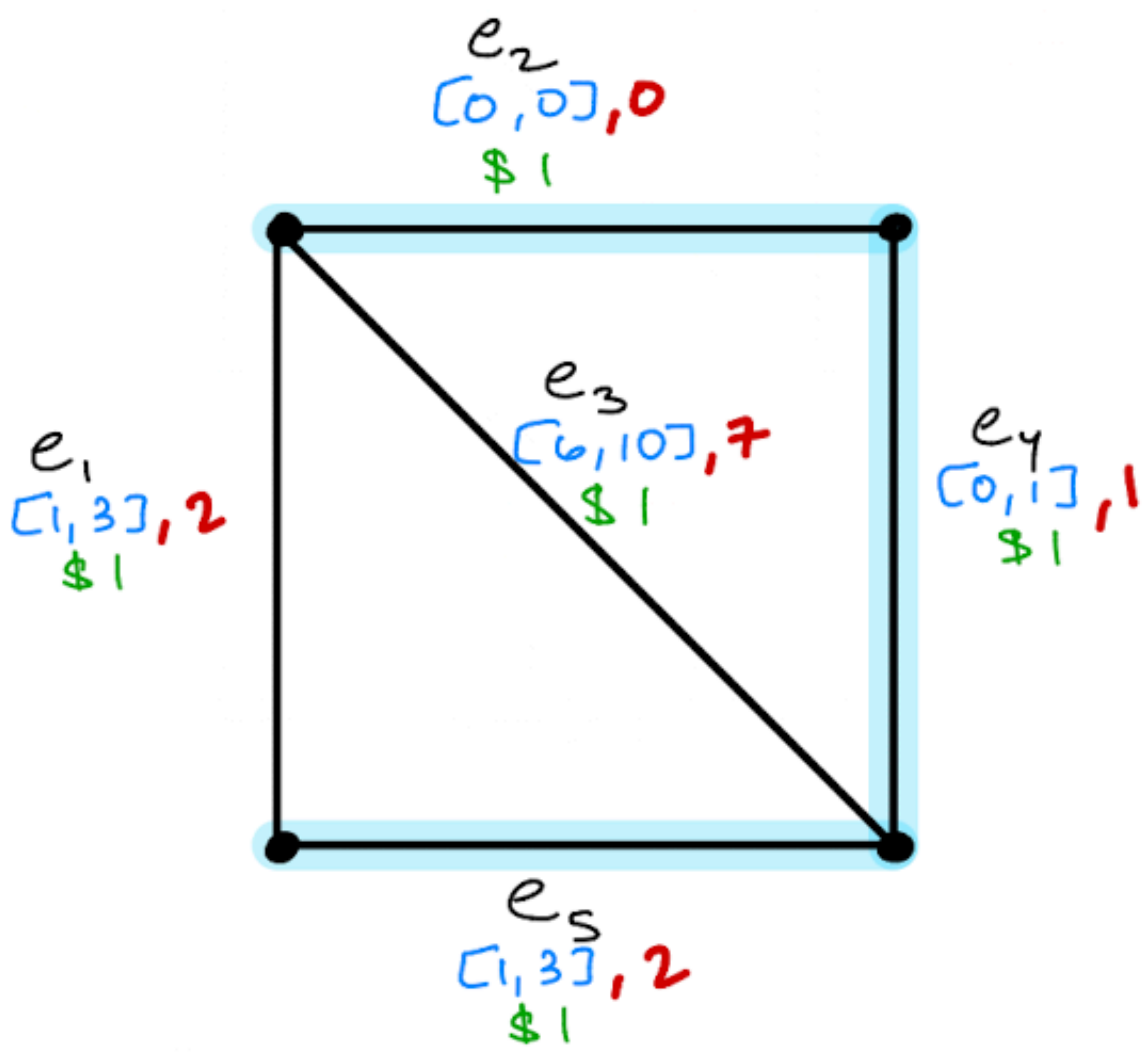
Phase 1: Contractions and Deletions

- We use properties of MSTs to derive contraction (K) and deletion (D) rules
- Phase 1 applies K-rules and D-rules to eliminate all extreme edges
  - Edges in K will be included in output MST
  - Edges in D will be excluded
  - Tie-break using query costs
- Rules ensure contractions and deletions are consistent with a cheapest-to-verify MST

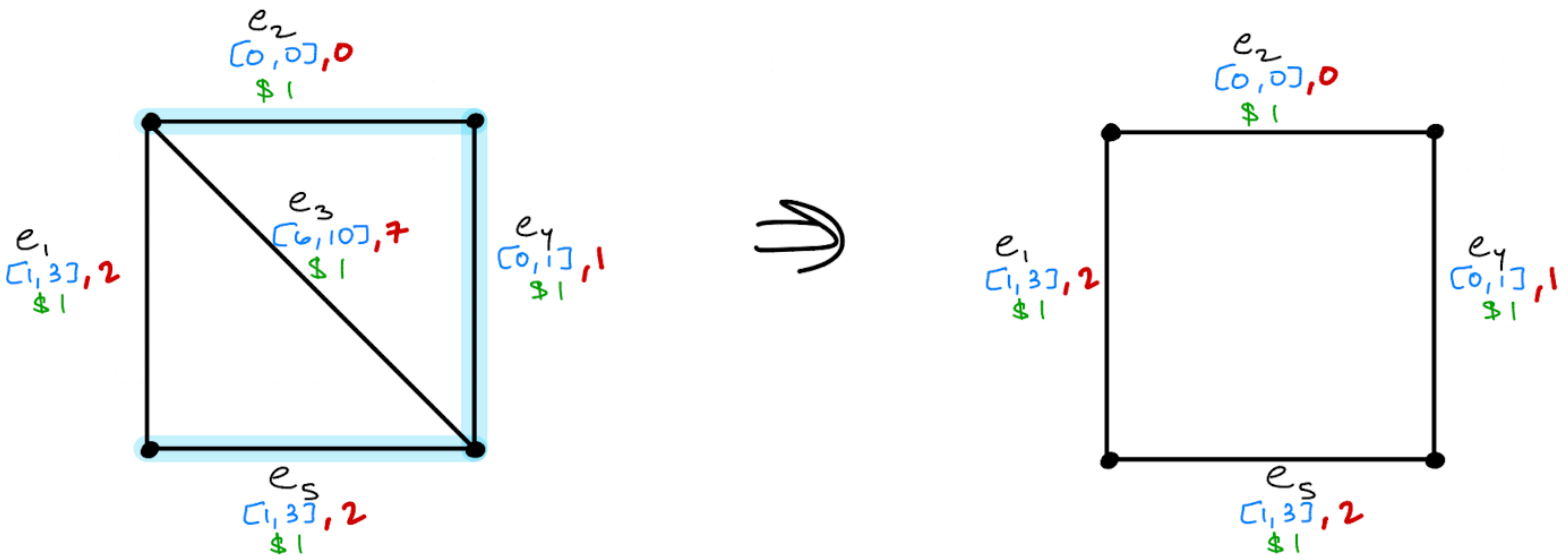
Example Rules

- **Example K-rule (1):** Contract edge e (add to set K) if
  - It belongs to some MST T of current induced multigraph G'
  - $w_e = U_e$ and $U_e \neq L_e$
  - e is a min-weight edge in the set of edges that can replace e to form a new Spanning Tree
  - e has highest query cost among similar extreme edges that can replace e to form a new Spanning Tree
- **Example K-rule (2):** Contract an edge e (add to set K) if
  - It belongs to some MST T of current induced multigraph G'
  - e is the unique minimum weight edge connecting the two components formed by $T \backslash \{e\}$

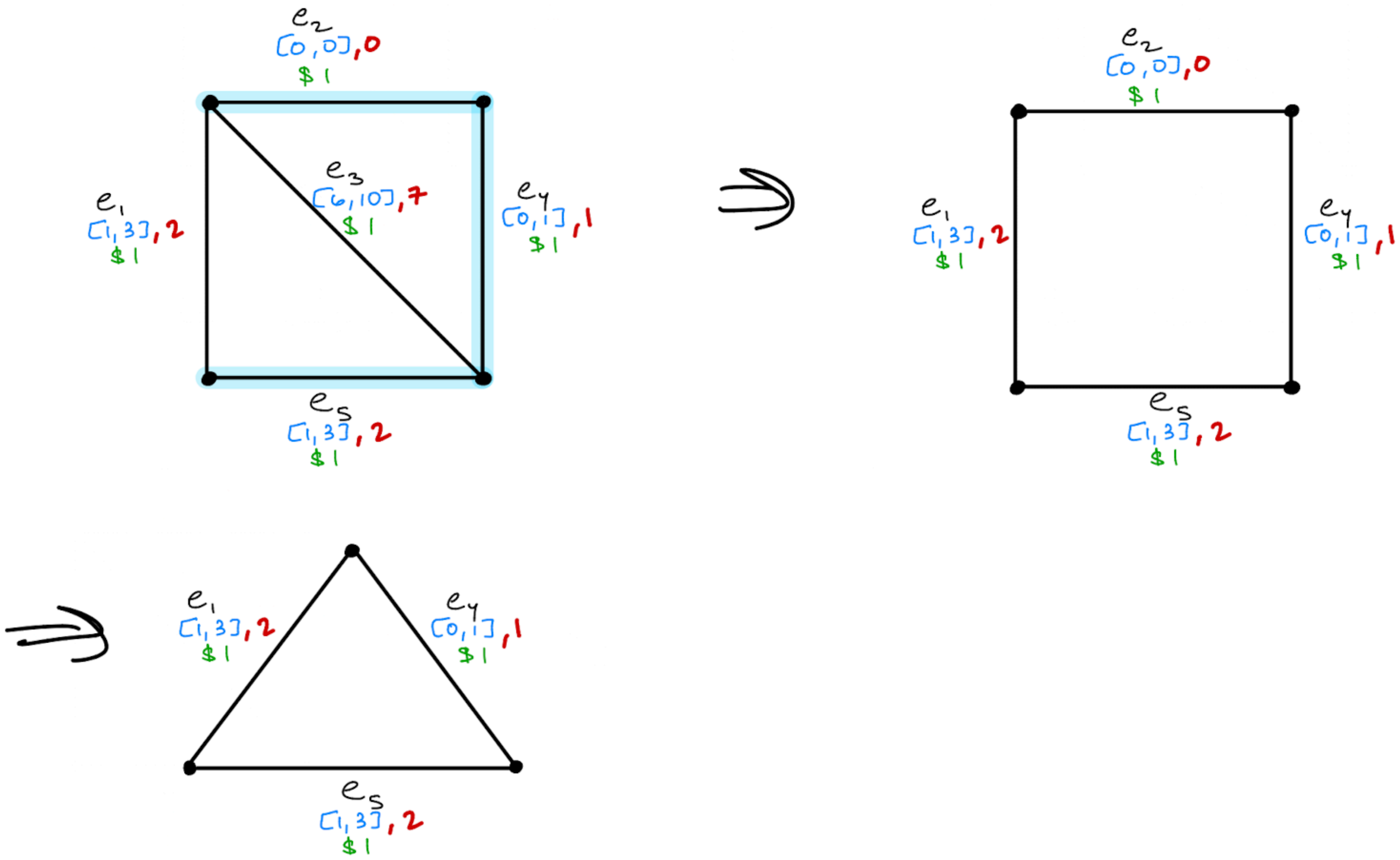
Contraction and Deletion Example



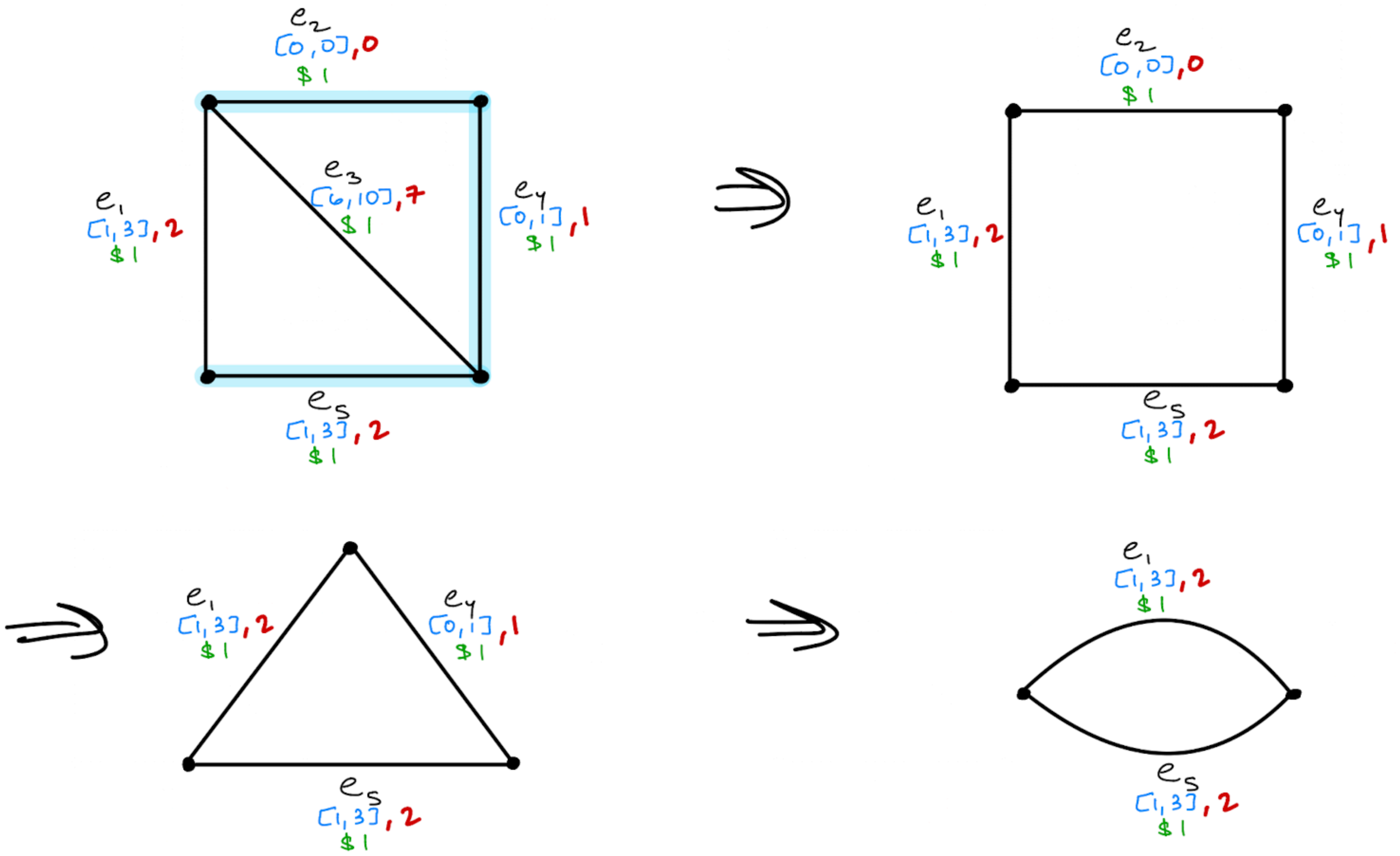
Contraction and Deletion Example



Contraction and Deletion Example



Contraction and Deletion Example



Algorithm Runs in Polynomial Time

- **Phase 1:**
  - Checking the conditions for applying K and D rules can be done in polynomial time
  - Output is a cheapest-to-verify MST T
- **Phase 2:**
  - Constructs an instance of Bipartite Weighted Vertex Cover from G and T
  - Solves it using known poly-time algorithm
  - Solution gives min-cost certificate for T

From Offline to Online

- Solving the (offline) verification problem is often a prerequisite to solving online versions of the problem
- **Open problem:** develop a polynomial time algorithm solving the stochastic version of the MST problem
  - Weight of an edge comes from a known distribution
    - Edge weights are independent
  - Need to query an edge to reveal its weight
  - **Goal:** minimize expected query cost

Questions?

References

- [1] Thomas Erlebach and Michael Hoffmann. Minimum spanning tree verification under uncertainty. In Dieter Kratsch and Ioan Todinca, editors, Graph-Theoretic Concepts in Computer Science, volume 8747 of Lecture Notes in Computer Science, pages 164–175. Springer International Publishing, Cham, 2014. doi:10.1007/978-3-319-12340-0_14.

- [2] T. Erlebach, M. Hoffmann, D. Krizanc, M. Mihalák, R. Raman, Computing minimum spanning trees with uncertainty, in: S. Albers, P. Weil (Eds.), 25th International Symposium on Theoretical Aspects of Computer Science, STACS'08, in: LIPIcs, vol. 1, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Germany, 2008, pp. 277–288.

- [3] Thomas Erlebach, Michael Hoffmann, and Frank Kammer. Query-competitive algorithms for cheapest set problems under uncertainty. Theor. Comput. Sci., 613:51–64, 2016. doi:10.1016/J.TCS.2015.11.025.

- [4] Manoj Gupta, Yogish Sabharwal, and Sandeep Sen. The update complexity of selection and related problems. Theory Comput. Syst., 59(1):112–132, 2016.