

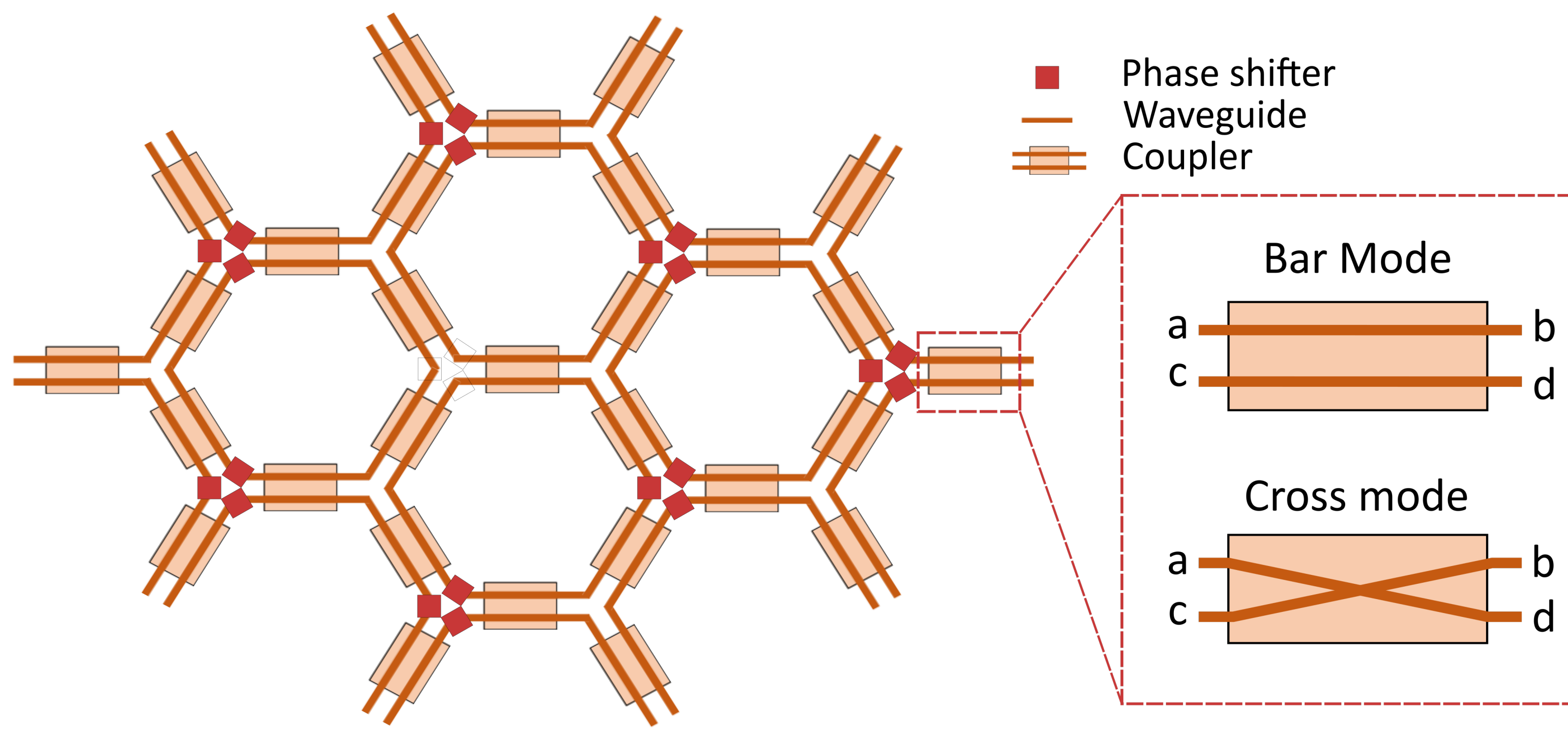
# ADAPTING ROUTING ALGORITHMS TO PROGRAMMABLE PHOTONIC CIRCUITS

Ferre Vanden Kerchove<sup>1</sup>, Xiangfeng Chen<sup>2</sup>, Didier Colle<sup>1</sup>, Wim Bogaerts<sup>2</sup>, Mario Pickavet<sup>1</sup>

<sup>1</sup> IDLab, Ghent University - imec

<sup>2</sup> Photonic Research Group, Ghent University - imec  
ferre.vandenkerchove@ugent.be

## Programmable photonic mesh



## Objective function

Optimisation criteria:

1. Insertion loss (IL)
2. Basic unit length (BUL)
3. Power consumption (P)

Edge weight  $b_w$ , where  $c_1, c_2, c_3$  are freely chosen

$$b_w = c_1 \cdot IL_w + c_2 \cdot BUL_w + c_3 \cdot P_w$$

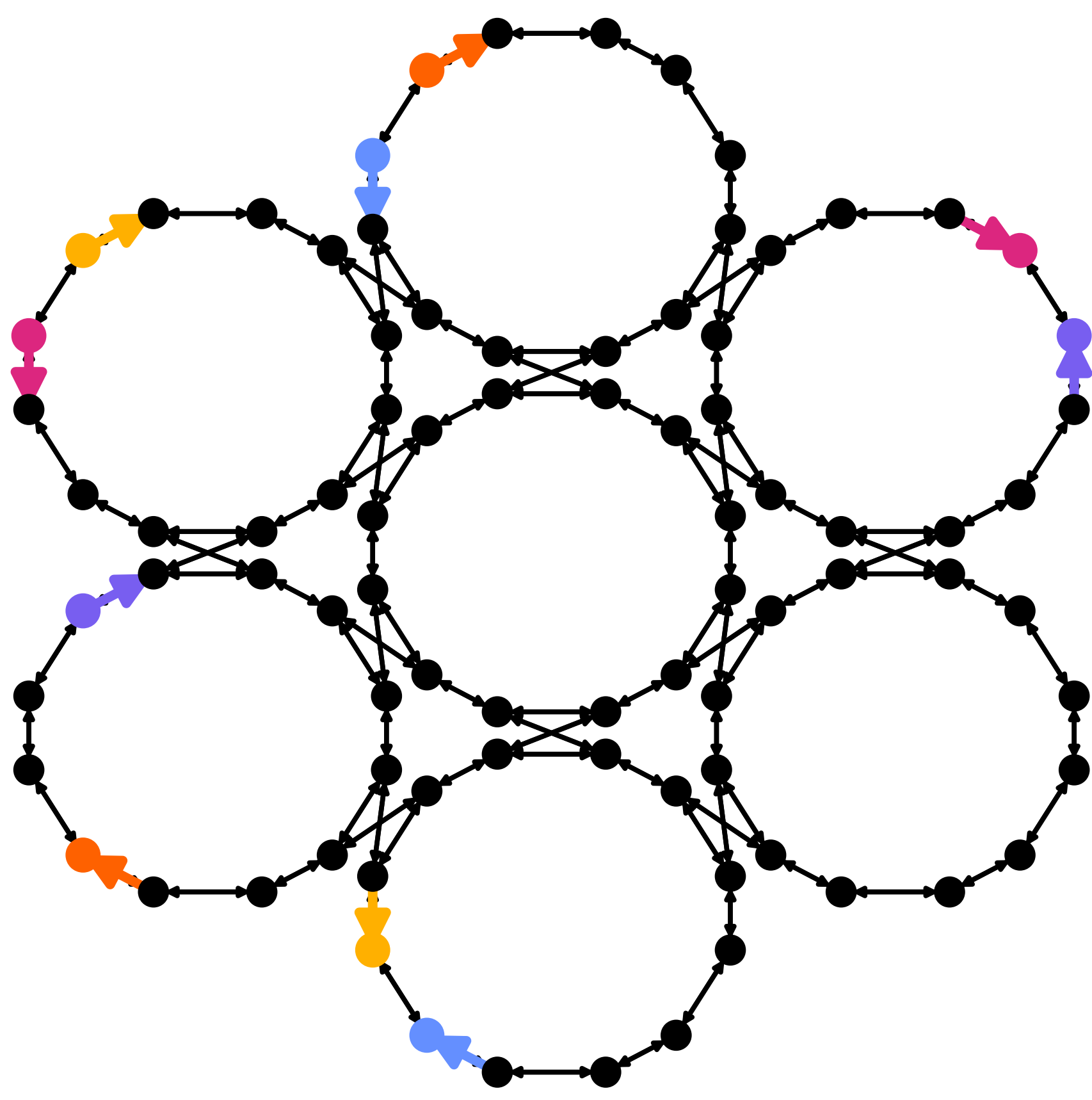
Optimisation goal:

$$\text{Minimize } \sum_{w \in W} b_w$$

$W$  = all the waveguides in the solution

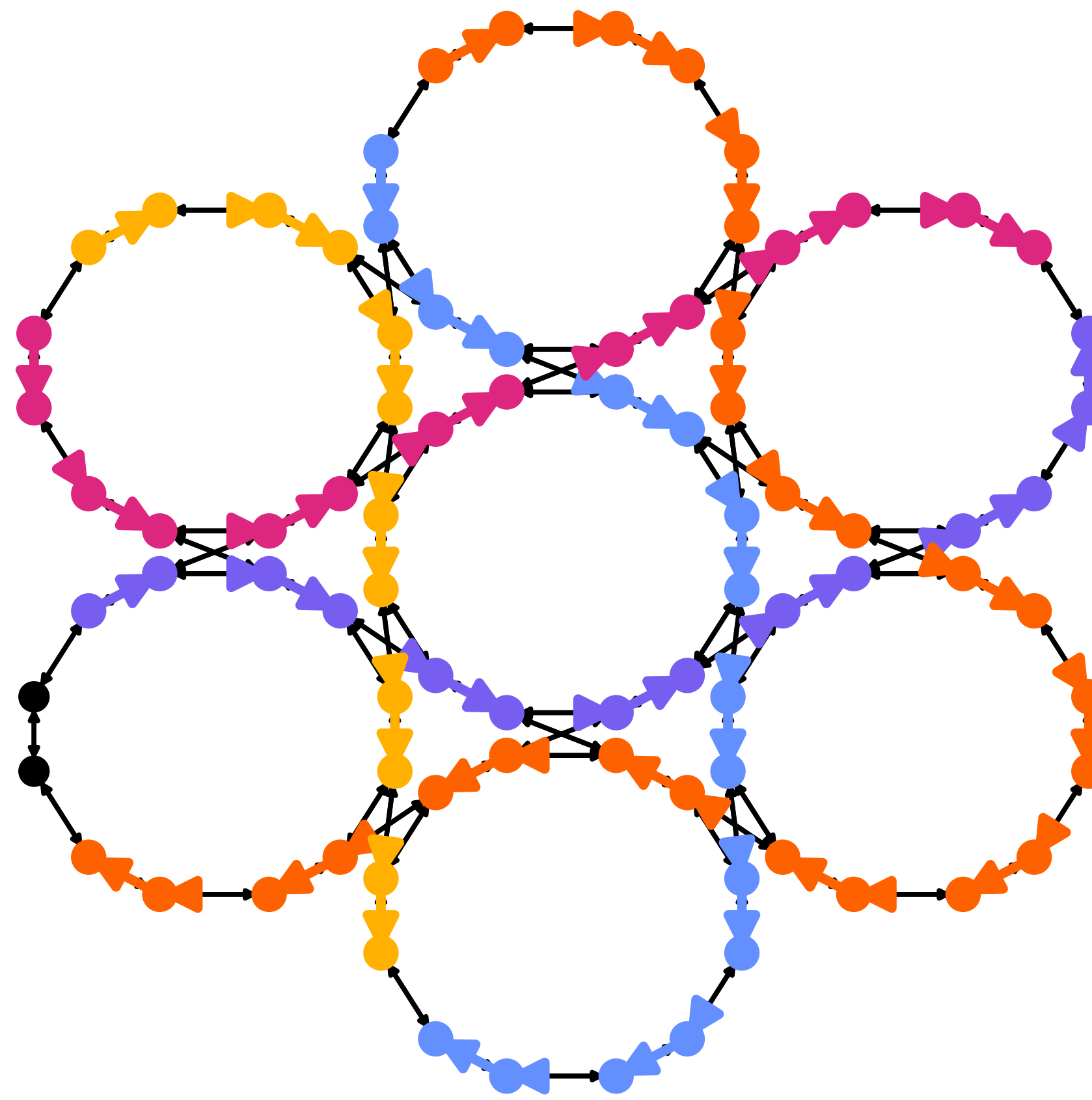
A solution is the set of paths assigned to the commodities.

## Problem statement

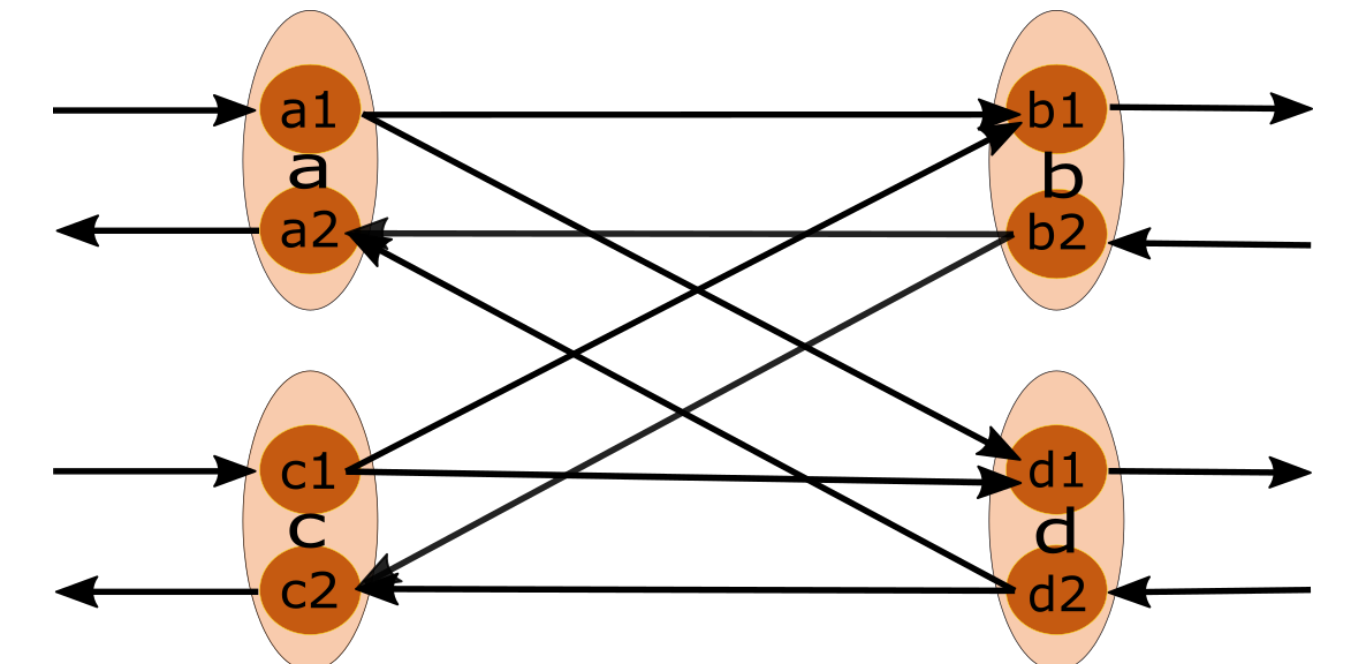


- Photonic mesh as **graph**  $G$
- **Commodities** are source-destination pairs

## Optimal solution



## Coupler - graph representation



See Xiangfeng et al. "Graph Representations for Programmable Photonic Circuits."

## Algorithm weights

Cost  $c_w$  to use waveguide  $w$ :

$$c_w = (b_w + h_w) \cdot p_w$$

- $b_w$ : edge weight
- $h_w$ : increase each iteration when **congested**
- $p_w$ : the number of other paths that use this waveguide

## Iterative congestion-negotiation algorithm

1. Route all commodities by least-weighted path
2. Return solution if found
3. Increase congestion history  $h_w$  of congested waveguides
4. Go back to step 1.

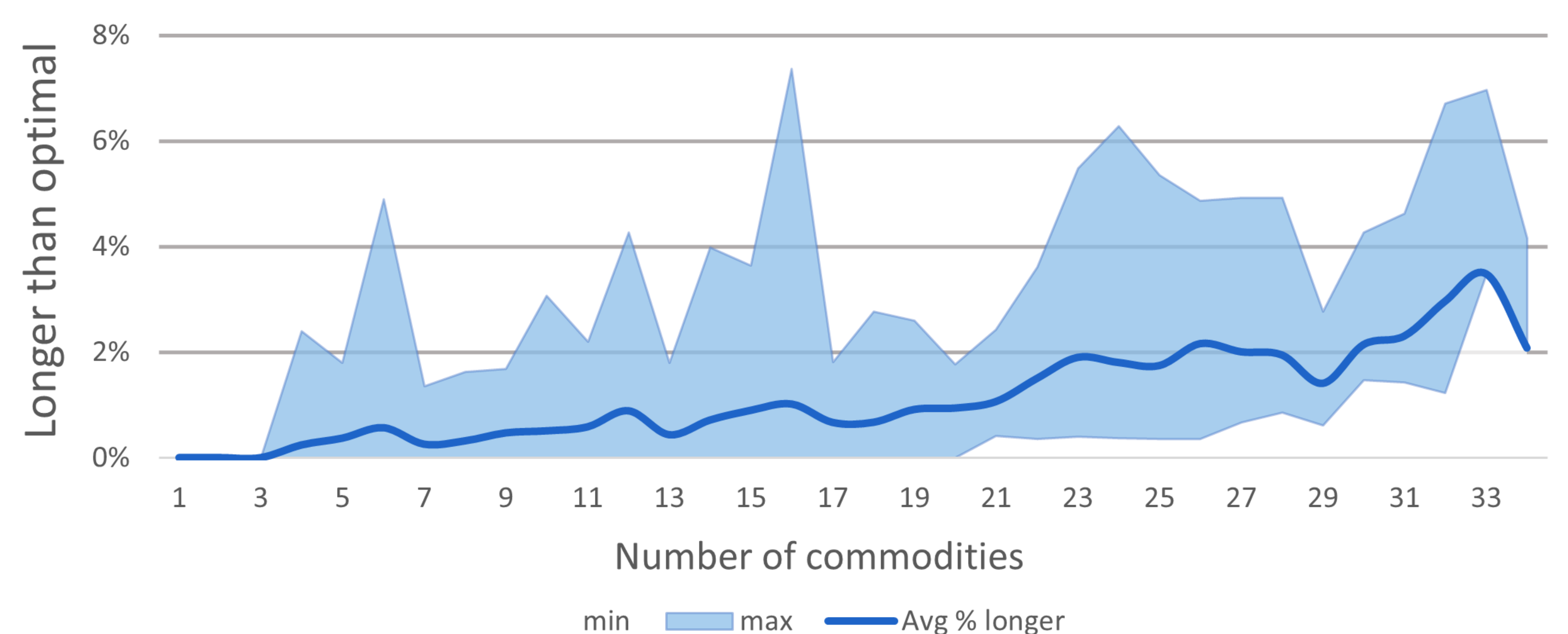
## Additional routines

- Preprocessing
- Loop detection
- Length-based restart

## Conclusion

- On average within **4%** of optimal value
- **10-100**× faster than an integer program.

## Comparison to optimal value



## Future work

- Extension to commodities with multiple destinations
- Synthesis and placement of various interferometric structures