Inferring pathways in metabolic networks via optimal factories and hyperpaths

John Kececioglu
The University of Arizona

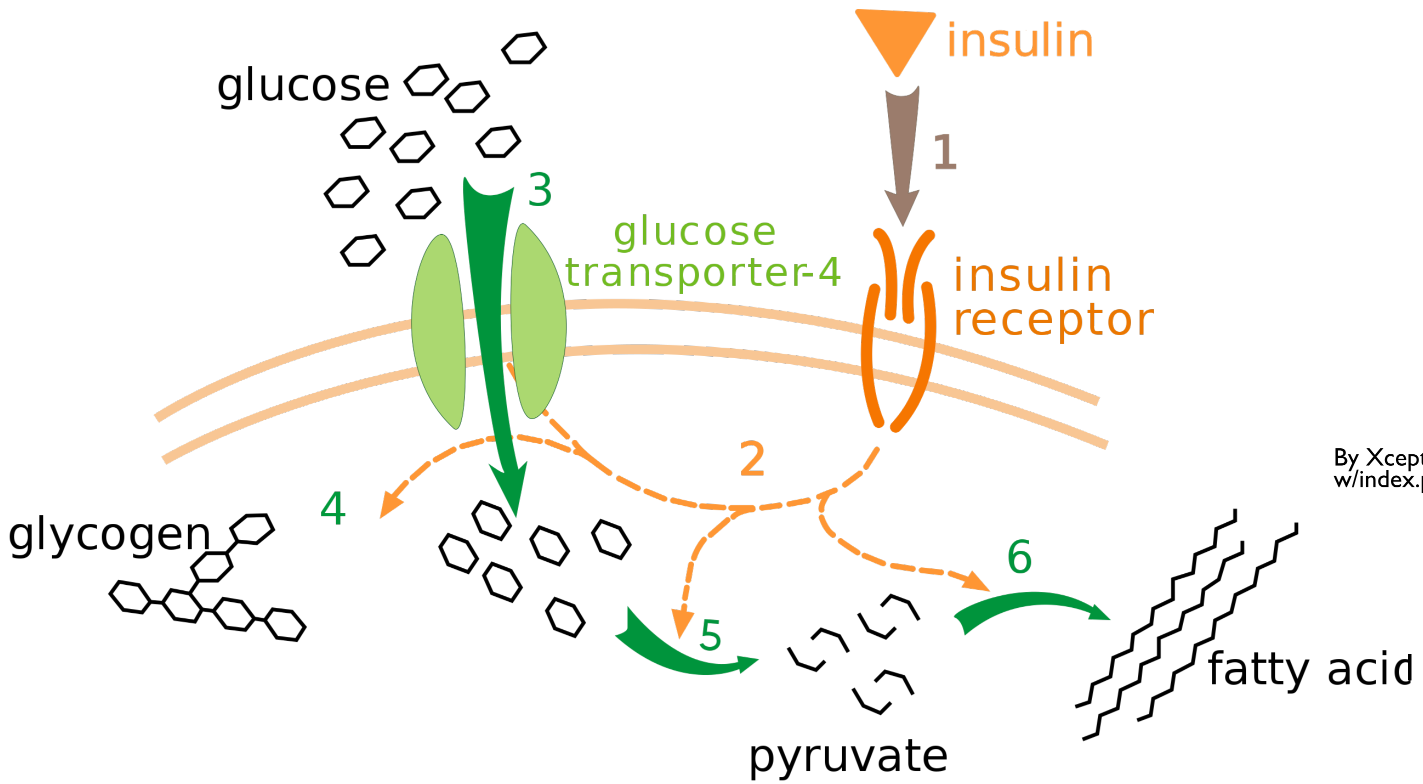
Spencer Krieger
Carnegie Mellon University

ARIZONA

Carnegie Mellon University

Background: Pathways

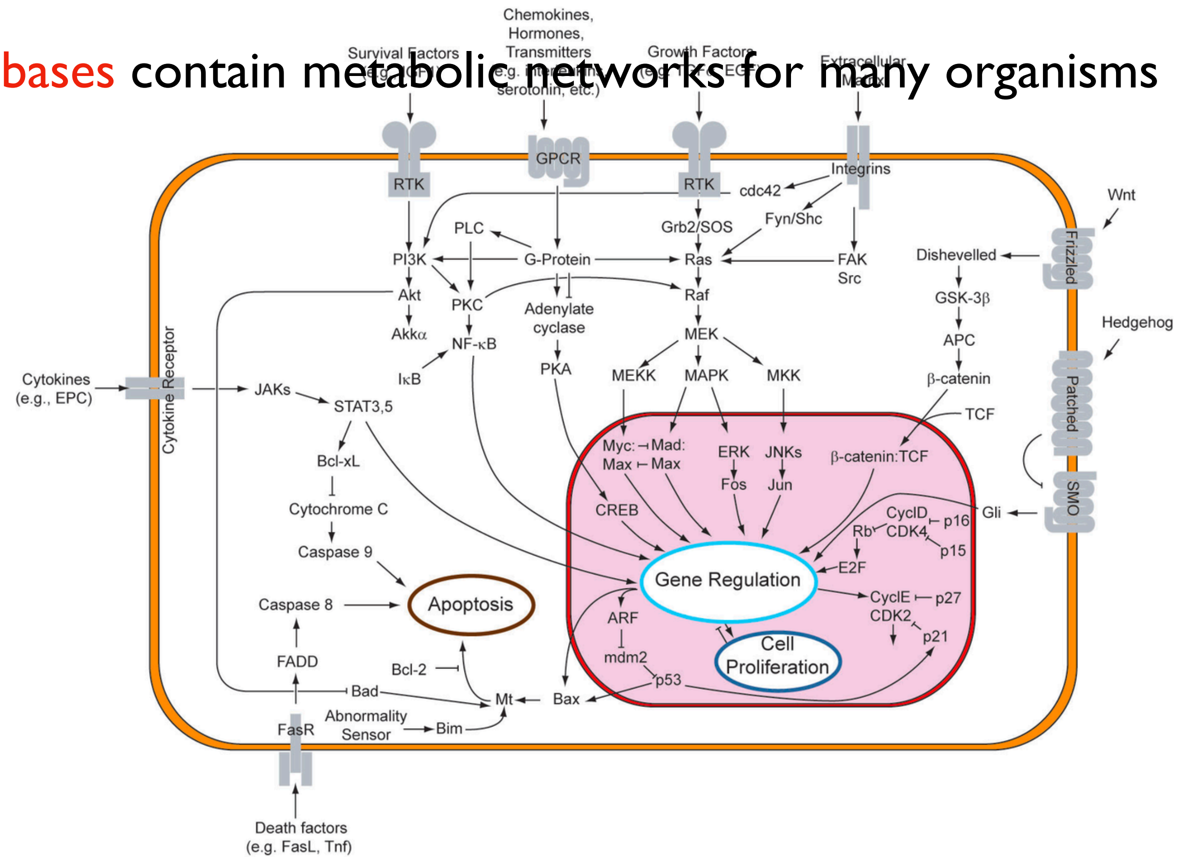
- A pathway is a collection of reactions culminating in a specific cellular response



By XcepticZP, commons.wikimedia.org/w/index.php?curid=4436369

Background: Metabolic networks

- Metabolic networks contain reactions from many annotated pathways
- Network databases contain metabolic networks for many organisms

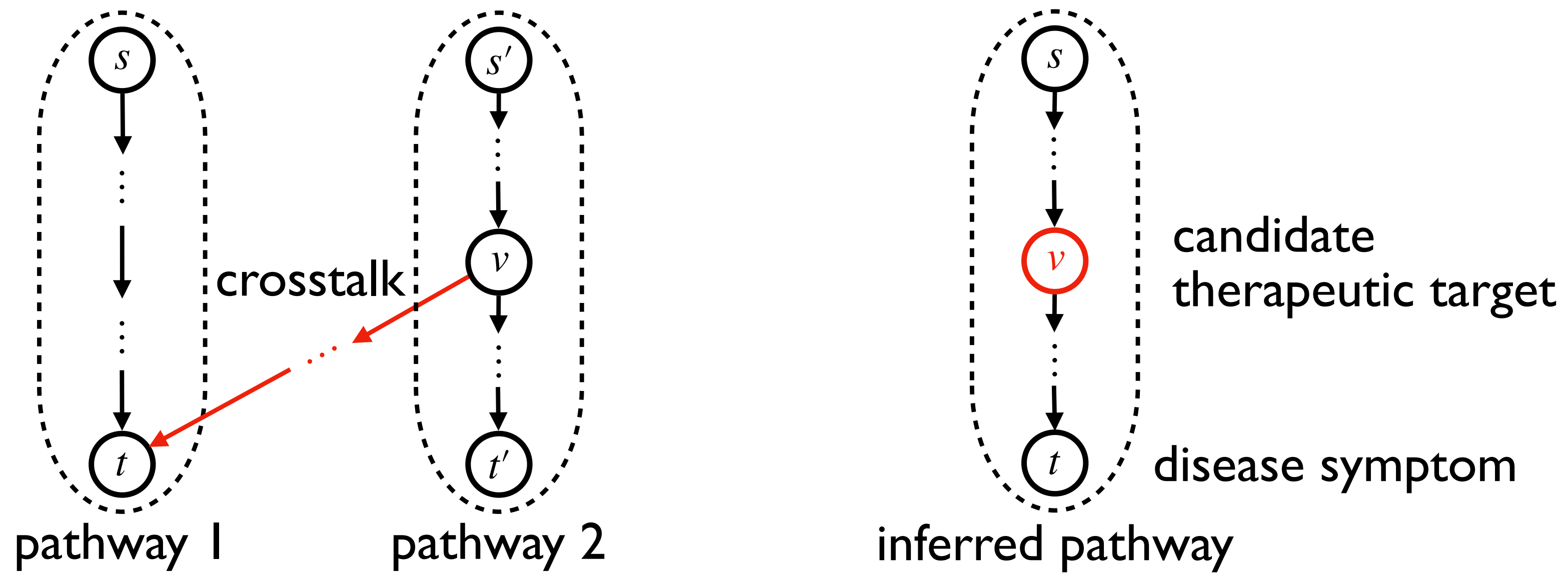


Background: Pathway inference

The pathway inference task is:

> **Given** a set of available source compounds and a set of target molecules,
> **find** a collection of reactions producing the targets from the sources.

Pathway inference can identify crosstalk and therapeutic targets



Background: Reactions

- General form of a reaction:

$$wA + \cdots + xB \longrightarrow yC + \cdots + zD$$

Background: Reactions

- General form of a reaction:

$$wA + \cdots + xB \longrightarrow yC + \cdots + zD$$

reactants

Background: Reactions

- General form of a reaction:

$$wA + \cdots + xB \longrightarrow yC + \cdots + zD$$

reactants products

Background: Reactions

- General form of a reaction:

$$wA + \cdots + xB \longrightarrow yC + \cdots + zD$$

Background: Reactions

- General form of a reaction:

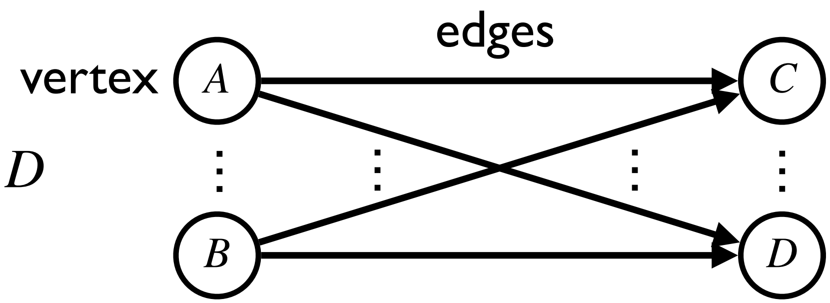
$$wA + \cdots + xB \longrightarrow yC + \cdots + zD$$

reactants products

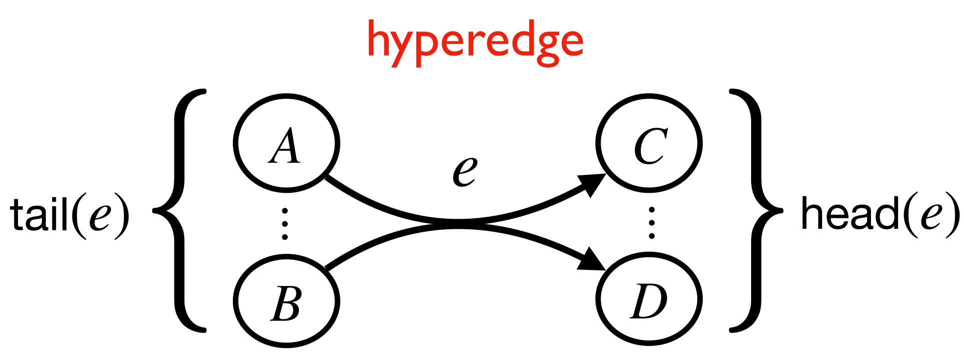
- A reaction consumes all reactants and produces all products
- To infer pathways, we represent reactions as edges in graphs

Background: Graphs vs. hypergraphs

$$A + \cdots + B \;\rightarrow\; C + \cdots + D$$

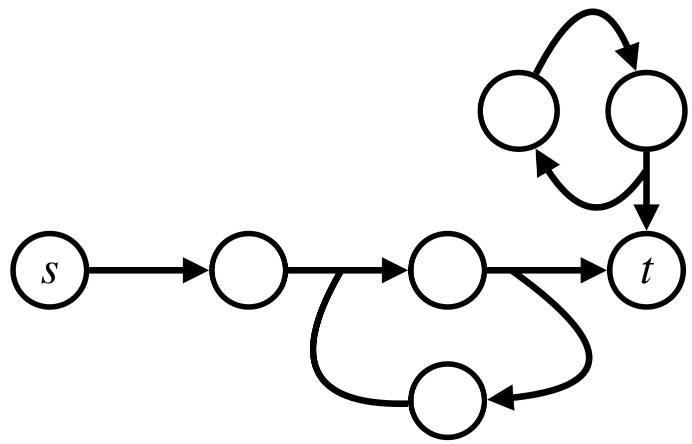


ordinary graph

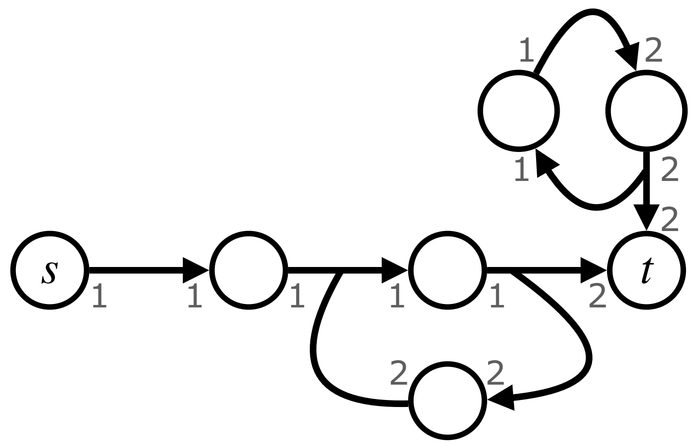


hypergraph

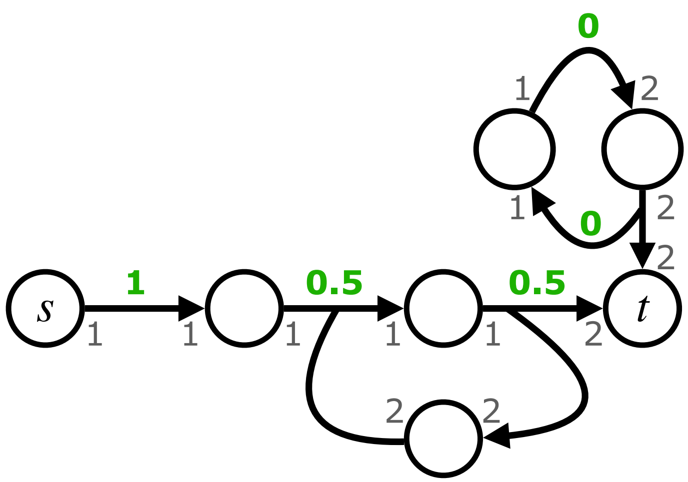
Background: Metabolic factories



Background: Metabolic factories

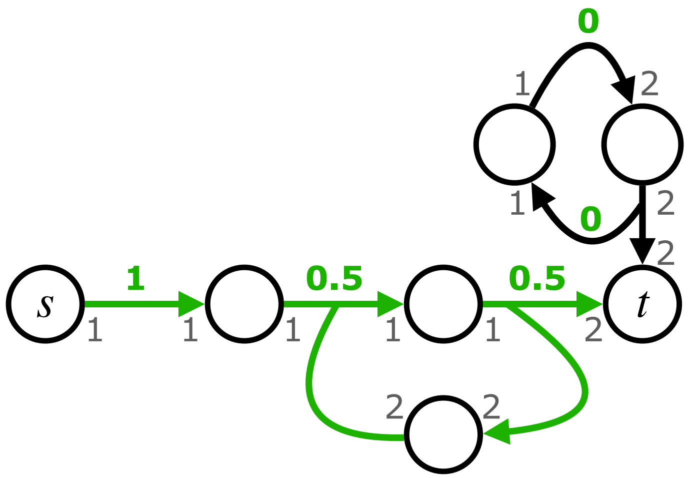


Background: Metabolic factories



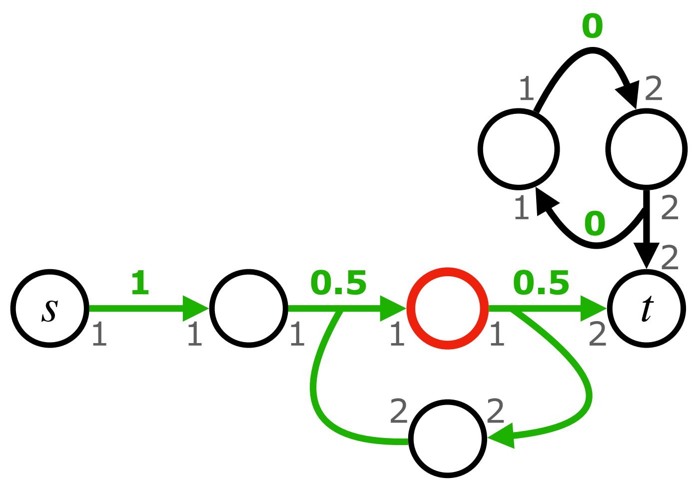
- A factory specifies a flux on all reactions, which produces all targets from sources

Background: Metabolic factories



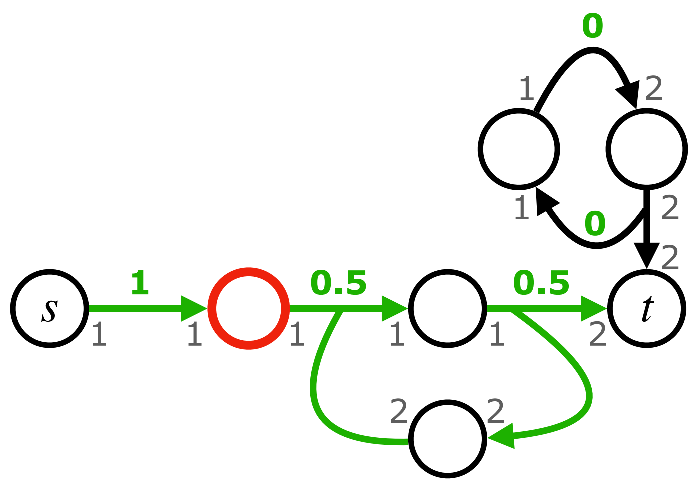
- A factory specifies a flux on all reactions, which produces all targets from sources

Background: Metabolic factories



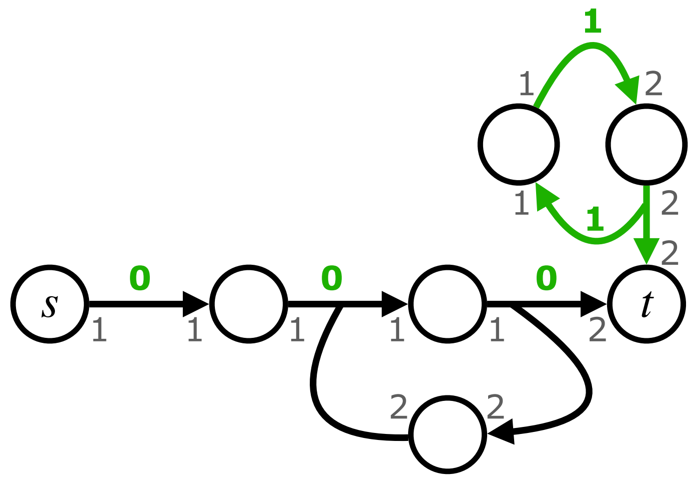
- A factory specifies a flux on all reactions, which produces all targets from sources, and satisfies conservation or accumulation

Background: Metabolic factories



- A factory specifies a flux on all reactions, which produces all targets from sources, and satisfies conservation or accumulation

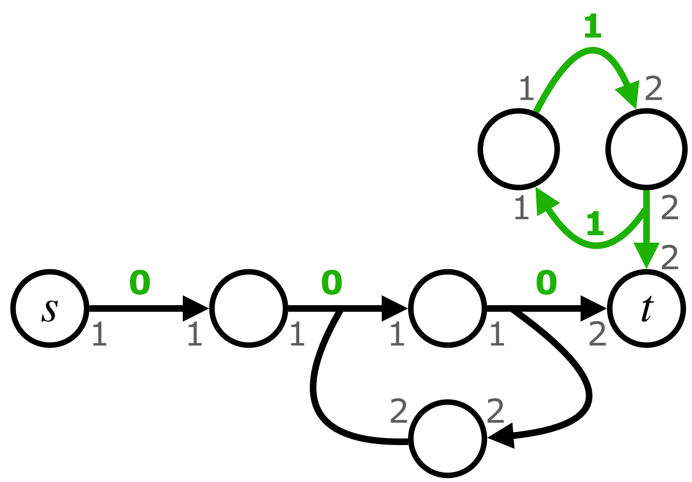
Background: Metabolic factories



???

- A factory specifies a flux on all reactions, which produces all targets from sources, and satisfies conservation or accumulation

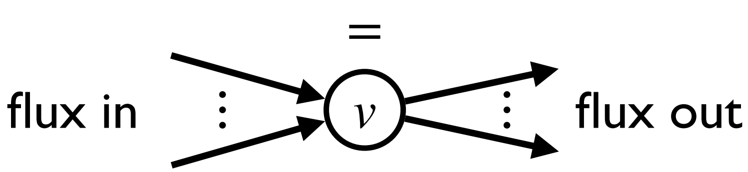
Background: Metabolic factories

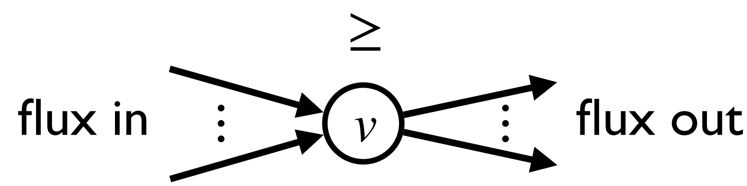


- A factory specifies a flux on all reactions, which produces all targets from sources, and satisfies conservation or accumulation
- Shortest factories is NP-complete

Background: Metabolic factories

- Metabolic flux f is a nonnegative real value on each reaction specifying relative usage
- An active hyperedge has nonzero flux on its reaction
- Stoichiometry matrix M gives stoichiometries for each metabolite in each reaction
- Metabolic flux must satisfy, on all intermediate metabolites $v \in I$:





Conservation $M|_I \cdot f = 0$ or Accumulation $M|_I \cdot f \geq 0$

- A factory specifies a flux on all reactions, which produces all targets from sources, and satisfies conservation or accumulation for intermediate metabolites

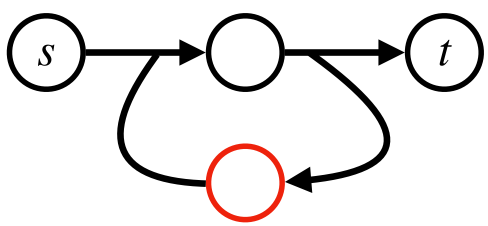
Factories vs hyperpaths

Factory

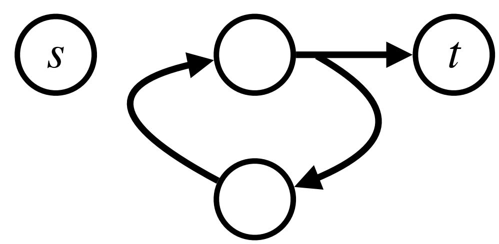
- Can consume reactants before they are produced
- Stoichiometry constraints must be satisfied
- Has degenerate solutions

Hyperpath

- Reactants are produced before consumed
- No stoichiometry constraints
- No degenerate solutions



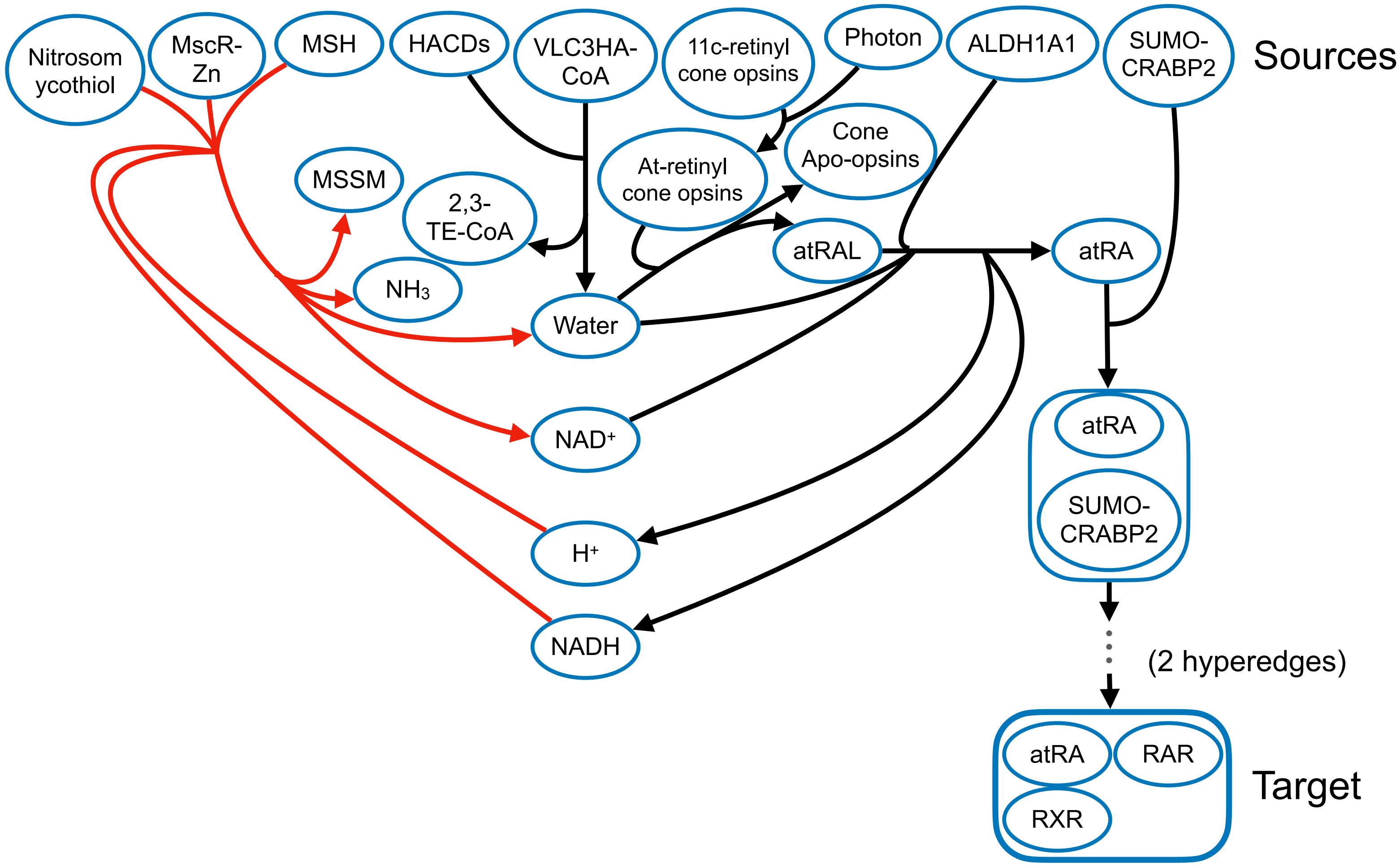
Factory



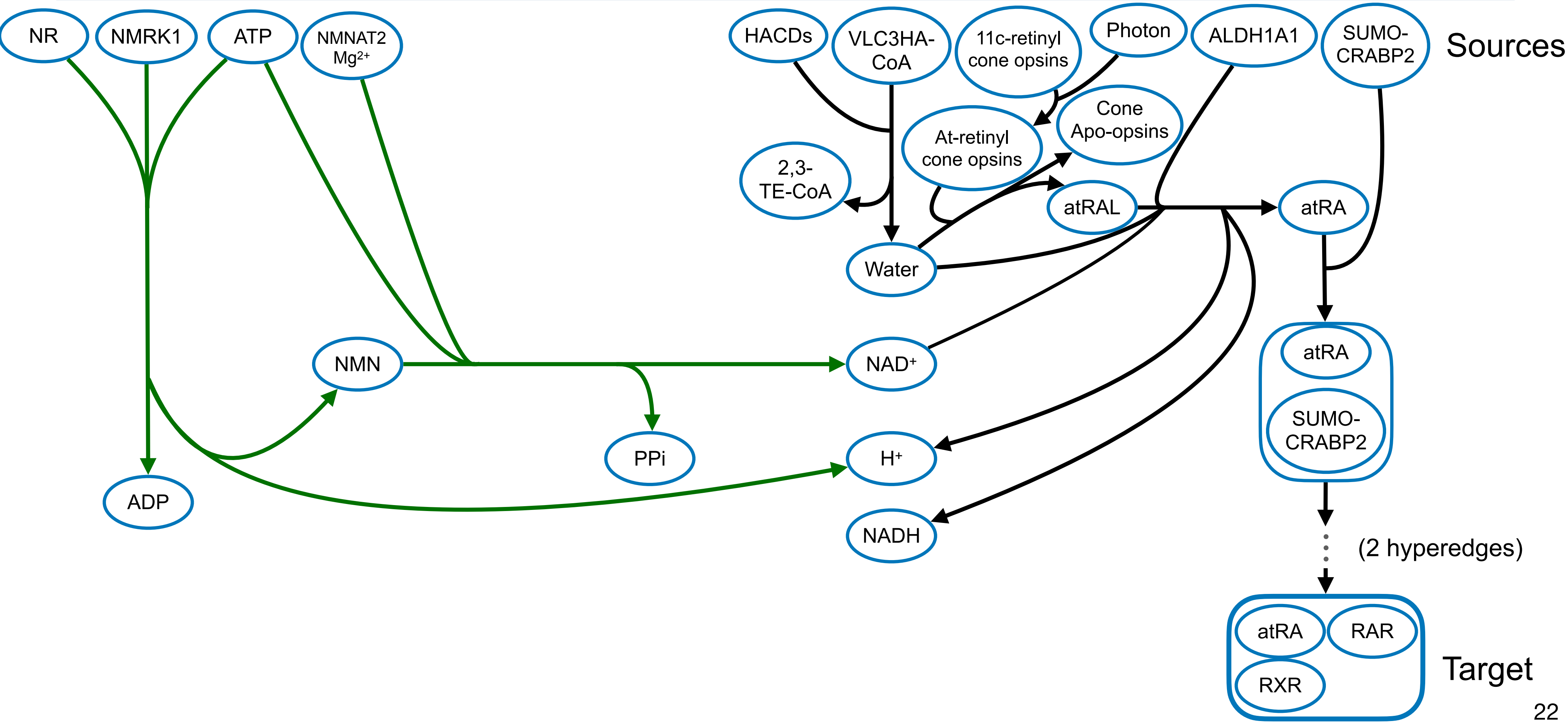
Degenerate factory

- Shortest factory and shortest hyperpath are both NP-complete

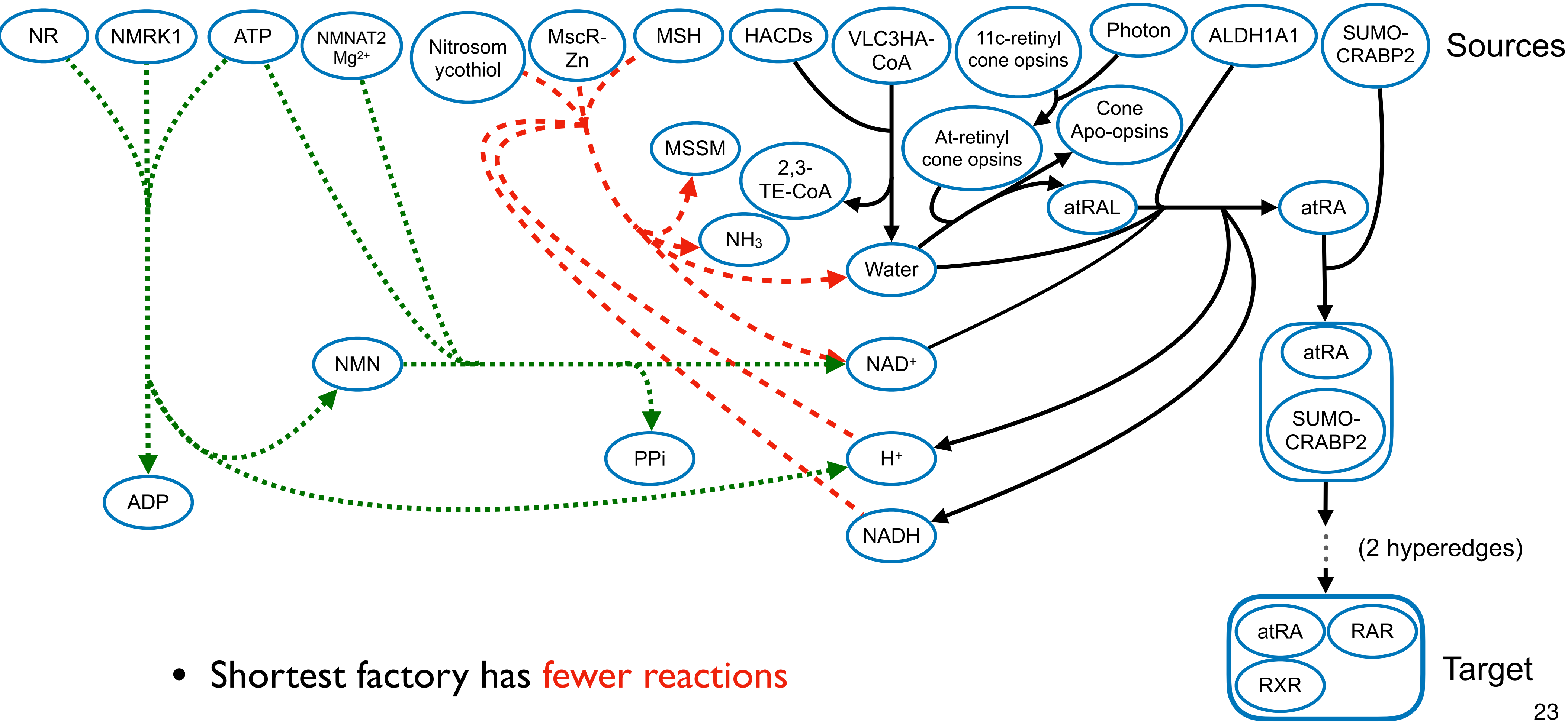
Example: Min-edge factory



Example: Shortest hyperpath

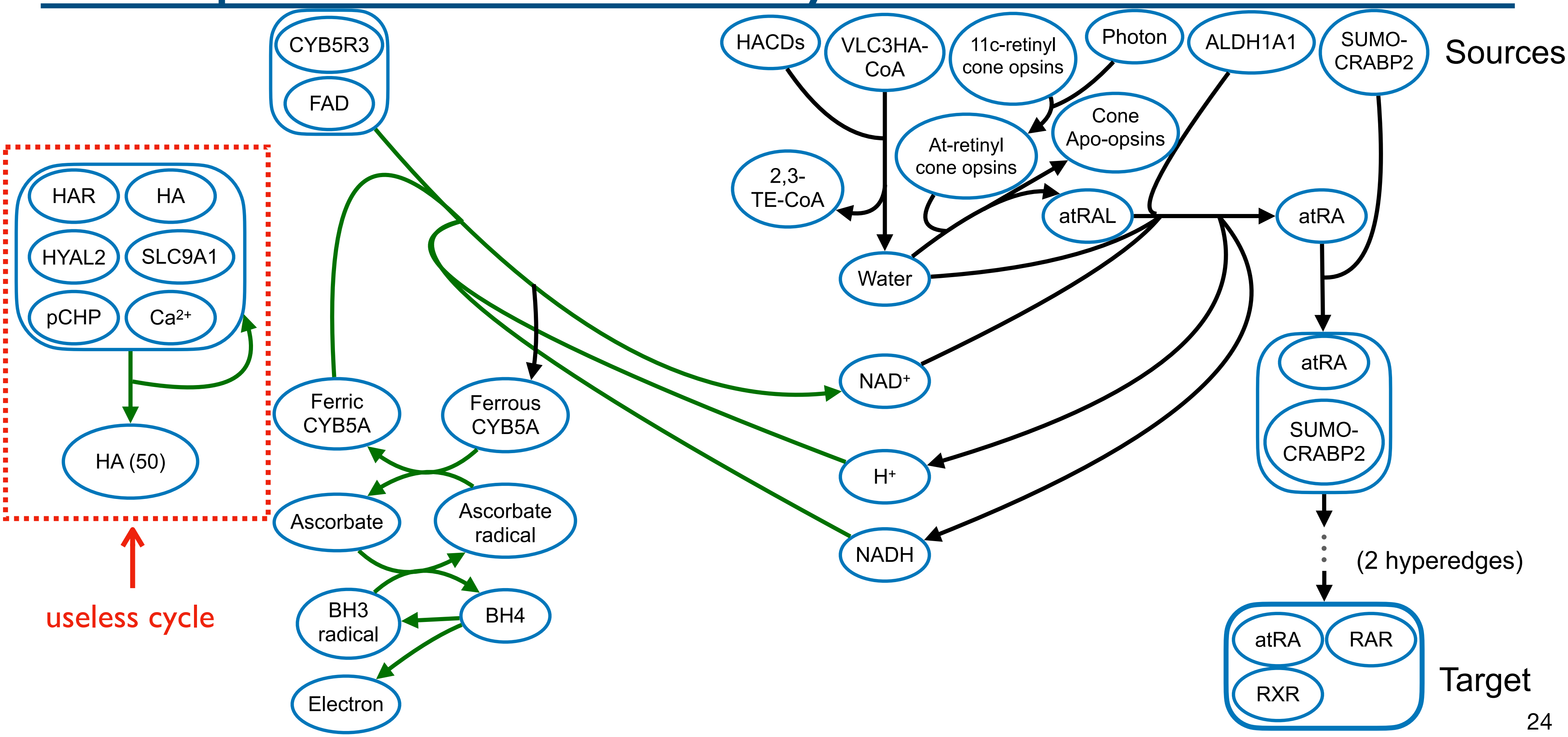


Example: Shortest factory vs shortest hyperpath

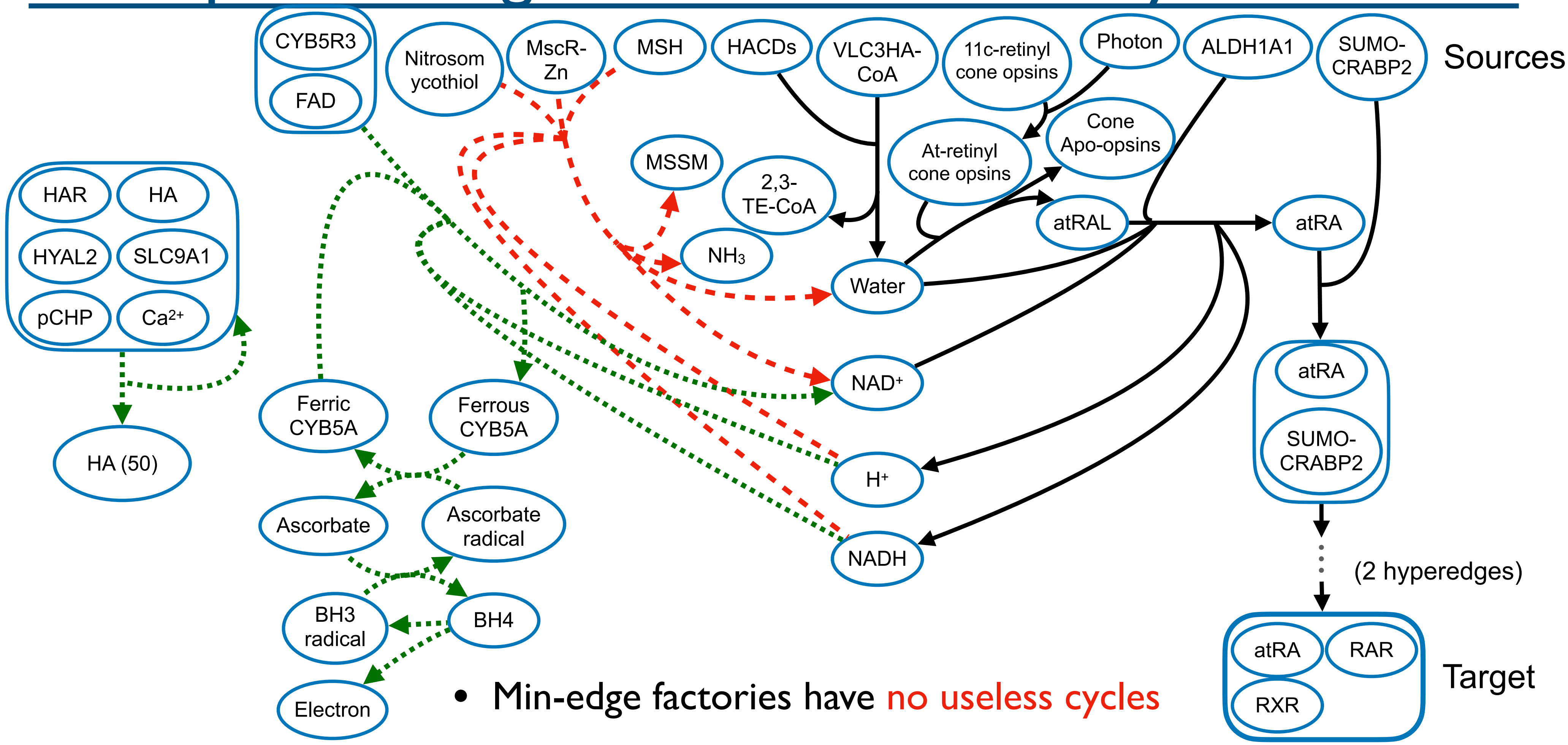


- Shortest factory has fewer reactions

Example: Min-source factory



Example: Min-edge vs min-source factory



- Min-edge factories have no useless cycles

Related work

Factories

- Andrade et al.: Min-source factory, three user-set parameters (Andrade et al., 2016)
- `Odinn`: Min-edge factory, one user-set parameter (Krieger and Kececioglu, 2022)

Hyperpaths

- Alternative model without stoichiometries, with ordered reactions where reactants are produced before consumed
- Ritz et al.: Exact algorithm for shortest acyclic hyperpaths (Ritz et al., 2017)
- `Hhugin`: Fast heuristic for general shortest hyperpaths (Krieger and Kececioglu, 2021)
- `Mmunin`: Exact algorithm for general shortest hyperpaths (Krieger and Kececioglu, 2023)

Shortcomings of user-set parameters

- All prior factory methods solve an MILP that has a target production parameter ϵ
- When ϵ is too high:
    - Excludes factories that produce a smaller amount of the target
- When ϵ is too low:
    - Introduces numerical errors in the MILP solver

Our contributions

Contributions

- We develop the first robust algorithm for shortest factories
  - Never misses a valid factory
  - No parameters that must be set
  - Solution is guaranteed to be nondegenerate
- We characterize the graph-theoretic structure of reactions in shortest factories
- We unify hypergraph models by showing that hyperpaths are a subclass of factories
- Our approach is fast in practice, with a median runtime of a few seconds
- Our approach is implemented in the tool `Freeia`

Methods

Methods: Shortest Factory

- Input is
  - hypergraph $G = (V, E)$
  - stoichiometry matrix M
  - candidate sources $S \subseteq V$
  - target molecules $T \subseteq V - S$

- Output is flux f such that
  - for intermediate metabolites $I = V - (S \cup T)$, either conservation or accumulation holds
  - every target $t \in T$ is produced in a nonzero amount
  - the weight of active hyperedges is minimized

Methods: Parameter-based MILP

- Variables: for each hyperedge $e \in E$
  - real-valued flux f_e with $0 \leq f_e \leq 1$
  - integer-valued active-edge indicator x_e with $x_e \in \{0,1\}$
- Constraints:
  - active edge constraints: $x_e \geq f_e$
  - conservation or accumulation constraints
  - target production constraint: $\Sigma_{e \in in(t)} f_e \geq \epsilon$ for each $t \in T$ and parameter ϵ
- Objective function: minimize weight of active hyperedges

Methods: Parameter-free MILP

- Variables: for each hyperedge $e \in E$
  - real-valued flux f_e with $0 \leq f_e \leq 1$
  - integer-valued active-edge indicator x_e with $x_e \in \{0,1\}$
- Constraints:
  - active edge constraints: $x_e \geq f_e$
  - conservation or accumulation constraints
- Objective function: minimize weight of active hyperedges

Methods: Parameter-free MILP

- Variables: for each hyperedge $e \in E$
    - real-valued flux f_e with $0 \leq f_e \leq 1$
    - integer-valued active-edge indicator x_e with $x_e \in \{0,1\}$
- Constraints:
    - active edge constraints: $x_e \geq f_e$
    - conservation or accumulation constraints
    - active hyperedge weight limit: $\Sigma_{e \in E} \; w_e x_e \leq \ell$
- Objective function: minimize weight of active hyperedges

Methods: Parameter-free MILP

- Variables: for each hyperedge $e \in E$
    - real-valued flux f_e with $0 \leq f_e \leq 1$
    - integer-valued active-edge indicator x_e with $x_e \in \{0,1\}$
- Constraints:
    - active edge constraints: $x_e \geq f_e$
    - conservation or accumulation constraints
    - active hyperedge weight limit: $\Sigma_{e \in E}\; w_e x_e \leq \ell$
- Objective function:

Methods: Parameter-free MILP

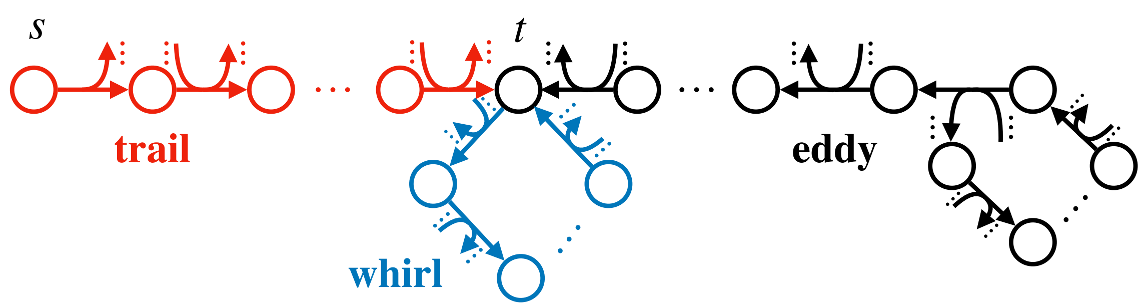
- Variables: for each hyperedge $e \in E$
  - real-valued flux f_e with $0 \leq f_e \leq 1$
  - integer-valued active-edge indicator x_e with $x_e \in \{0,1\}$
- Constraints:
  - active edge constraints: $x_e \geq f_e$
  - conservation or accumulation constraints
  - active hyperedge weight limit: $\Sigma_{e \in E}\ w_e x_e \leq \ell$
- Objective function: maximize target production

Methods: Parameter-free MILP

- Variables: for each hyperedge $e \in E$
  - real-valued flux f_e with $0 \leq f_e \leq 1$
  - integer-valued active-edge indicator x_e with $x_e \in \{0,1\}$
- Constraints:
  - active edge constraints: $x_e \geq f_e$
  - conservation or accumulation constraints
  - active hyperedge weight limit: $\Sigma_{e \in E} \ w_e x_e \leq \ell$
- Objective function: maximize target production

binary search
for minimum ℓ

Characterization theorem



Theorem (Characterizing factories)

F *is a shortest* s,t-*factory* *if and only if* F *is an* s,t-*trail*, -*whirl*, *or* -*eddy*.

- Trails correspond to nondegenerate solutions

Hyperpaths are factories

Theorem (Hyperpaths are Factories)

Every s, t-hyperpath has a flux that makes it a factory.

- Holds for any stoichiometries
- Proof yields an algorithm that constructs a valid flux for a hyperpath
- Implies shortest factories are not longer than shortest hyperpaths (and may be shorter)

Results

Results: Datasets

- We build hypergraphs over standard metabolic networks:
  - Reactome (Joshi et al., 2005)
  - Metabolic networks of organisms from MetExplore (Cottret et al., 2018)
    - *B. Aphidicola, B. Cicadellinicola, C. Rudii, E. Coli, H. Sapiens, S. Cerevisiae, S. Muelleri*

- Hypergraphs have up to:
  - 20,000 vertices
  - 12,000 hyperedges
  - 8,000 sources
  - 5,000 targets

- Reactome has 5,000 target instances:
  - 4,000 with factories under accumulation
  - 1,600 with factories under conservation
  - 2,400 with hyperpaths

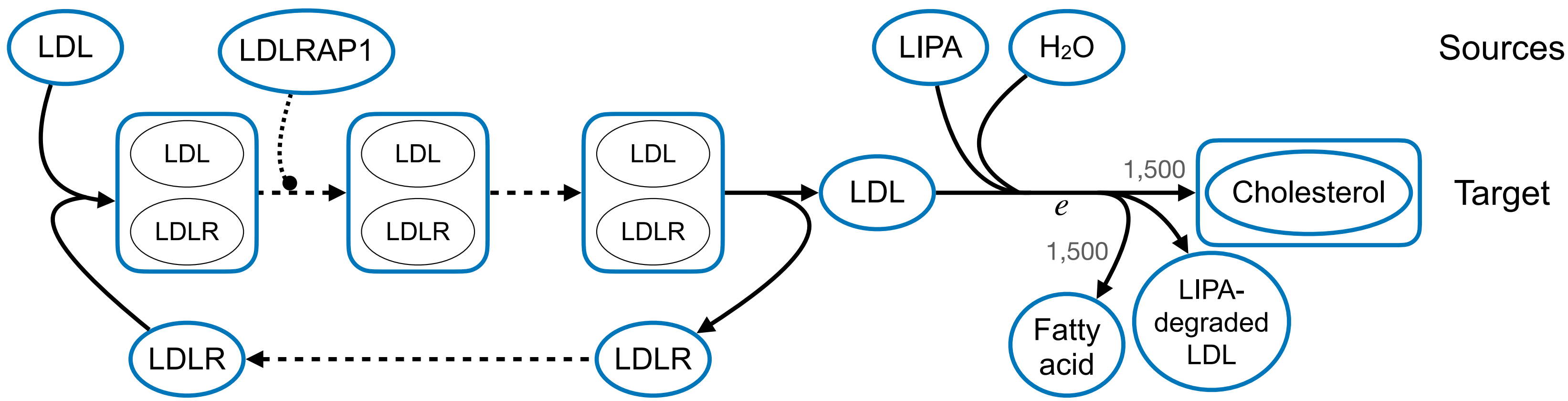
Results: `Freeia` outperforms state-of-the-art

- Prior state-of-the-art `Odinn` fails when parameter ϵ is too high or too low
- Our new tool `Freeia` can determine the largest valid ϵ for any instance
- For every ϵ value, `Odinn` fails to find an optimal factory for some Reactome instance

Target production parameter ϵ	10^{-5}	10^{-4}	0.01	0.1
`Odinn` failures	5,000	1	2	143

- `Freeia` finds an optimal factory for all instances
- Median runtime of `Freeia` is 5 seconds, maximum is just over an hour

Results: `Freeia` finds previously-missed factories



- Default value of ϵ in `Odinn` introduces numerical errors in the MILP solver

Conclusions

- First robust algorithm for optimal factories
- Complete characterization of the structure of reactions in optimal factories
- Unified the pathway models of hyperpaths and factories
- Finding optimal factories is fast in practice
- Implementation and all datasets are at

`http://freeia.cs.arizona.edu`

Further research

- Parameter-free algorithm for shortest factories under general edge weights
- Characterization of shortest factories under positive edge weights
- Fast heuristic for shortest factories
- Noninterference from negative regulators
- Extension to factories under conservation

Acknowledgements

People

TM Murali and Anna Ritz

Funding

NSF Grants IIS-2041613 and CCF-1617192

Tool

`freeia.cs.arizona.edu`

Contact

kece@cs.arizona.edu
skrieger@andrew.cmu.edu

Citation

Spencer Krieger and John Kececioglu,
"Robust optimal metabolic factories,"
Journal of Computational Biology 2024 (RECOMB 2024 special issue).

Thank you!

Spencer Krieger

John Kececioglu