Graph Transformation, Atom Tracing, and Isotope Labelling

Peter Severin Rasmussen

University of Southern Denmark (SDU)

June 28, 2018

Outline

Examples from chemistry

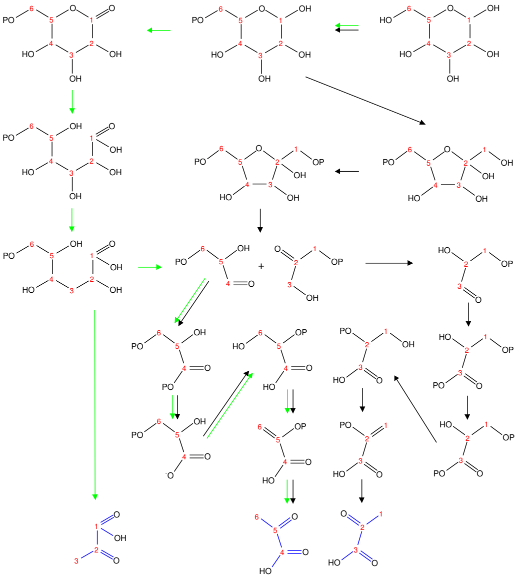
- Isotope labelling experiments
- Mass spectrometry
- Hypothetical (prebiotic) chemistries
- Metabolic engineering
- Synthesis planning
- One-pot synthesis

My master thesis

- Isotope labelling experiments ←
- Mass spectrometry
- Hypothetical (prebiotic) chemistries
- Metabolic engineering
- Synthesis planning
- One-pot synthesis

My master thesis

Isotope labelling experiments & atom tracing

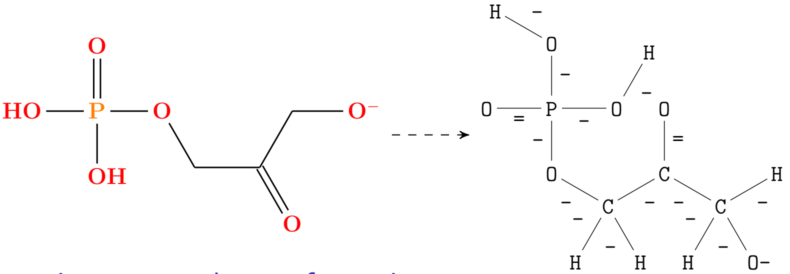


Glycolysis:
ED & EMP
Pathways

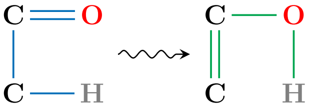
Preliminaries

The Molecular Model

Molecules as graphs

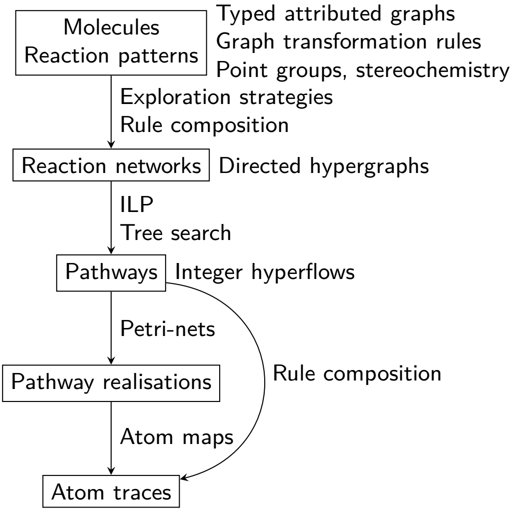


Reactions as graph transformations



MØD Overview

Models, methods, and concepts



Core Graph Algorithms

Monomorphism enum.
Isomorphism
Canonicalization
Automorphism enum.

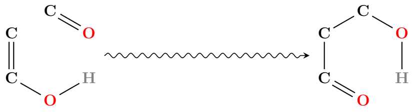
Software

The MØD package:

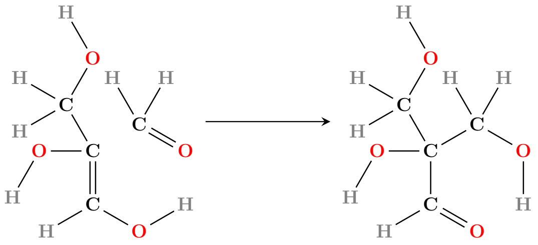
- C++ library
- Python interface
- Figure generation

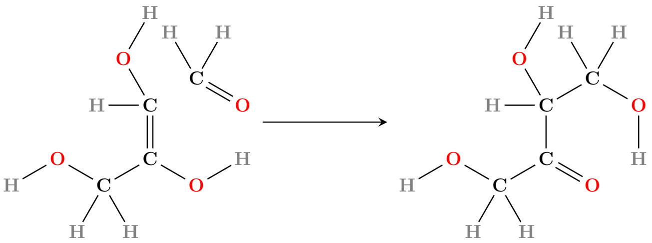
GraphCanon library
PermGroup library

Chemical reaction patterns

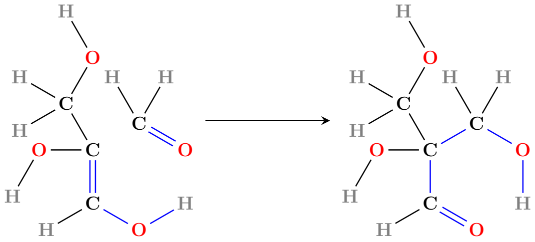


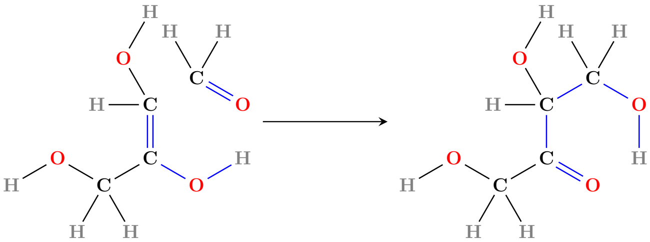
Chemical reaction patterns





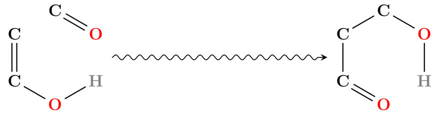
Chemical reaction patterns

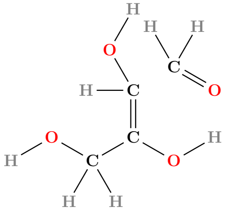




Chemical reaction patterns

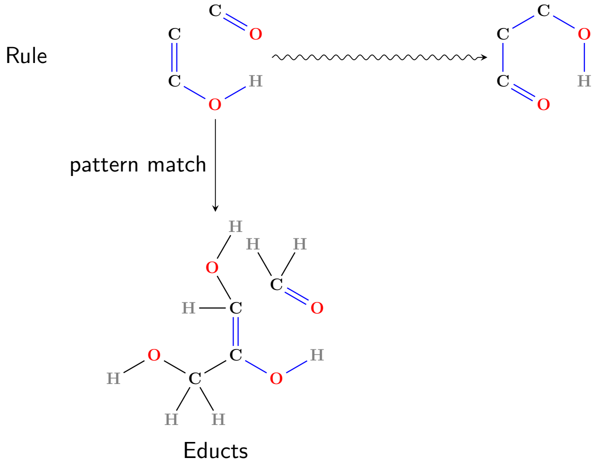
Rule



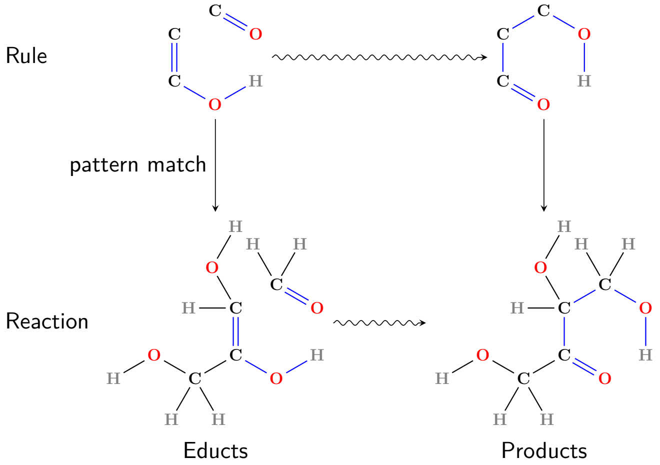


Educts

Chemical reaction patterns



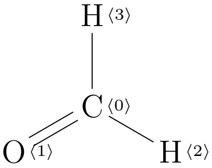
Chemical reaction patterns



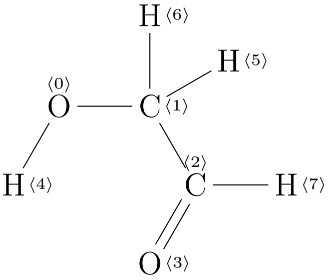
Example: Formose

Molecules

Formaldehyde



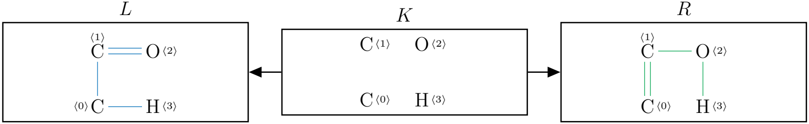
Glycolaldehyde



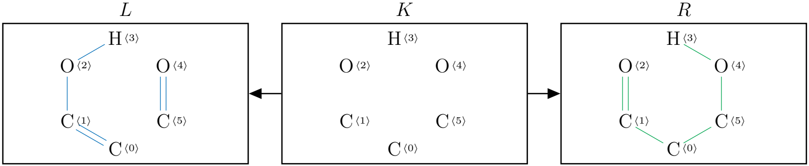
Example: Formose

Rules

Keto-Enol Isomerization

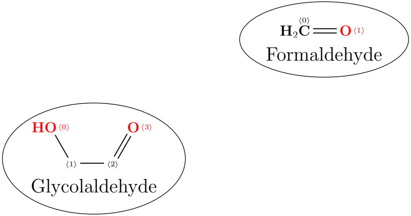


Aldol Addition



Example: Formose

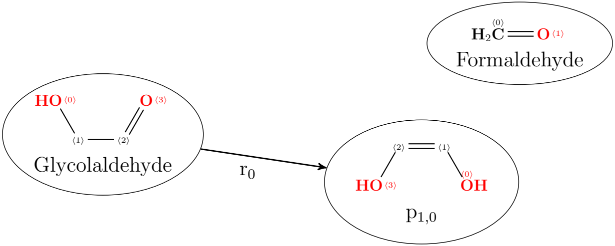
Derivation Graph / Chemical network



Generation 0

Example: Formose

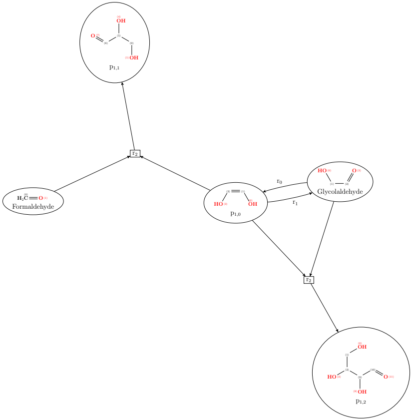
Derivation Graph / Chemical network



Generation 1

Example: Formose

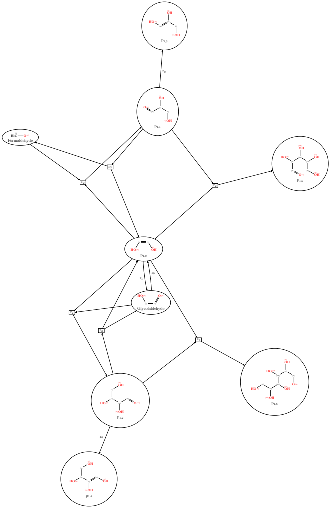
Derivation Graph / Chemical network



Generation 2

Example: Formose

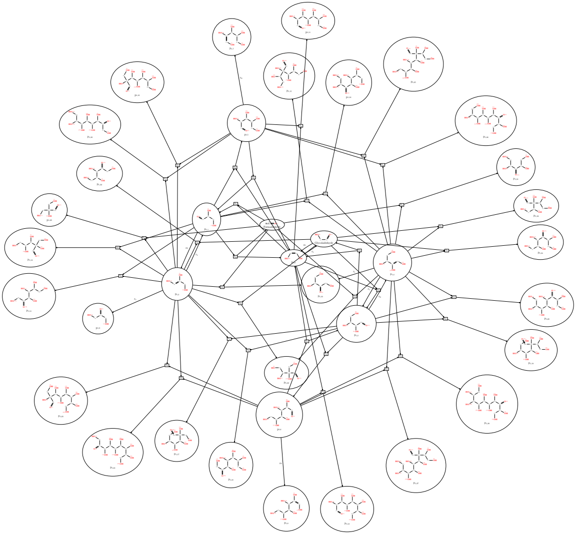
Derivation Graph / Chemical network



Generation 3

Example: Formose

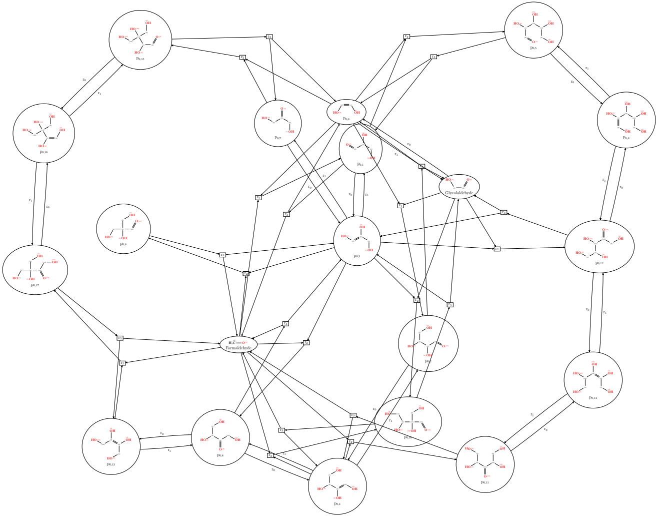
Derivation Graph / Chemical network



Generation 4

Example: Formose

Derivation Graph / Chemical network



Limited to molecules with ≤ 5 carbons

Group Theory

Group Theory

Group: $(G, \bullet)$

Closure If $g, h \in G$, then $g \bullet h \in G$.

Associativity For all $g, h, k \in G$, then $(g \bullet h) \bullet k = g \bullet (h \bullet k)$.

Identity There exists $e \in G$ s.t. for all $g \in G$, then

$$e \bullet g = g = g \bullet e$$

Inverse For all $g \in G$, there exists $g^{-1} \in G$ s.t.

$$g^{-1} \bullet g = e = g \bullet g^{-1}$$

Group Theory: Example

$$G = \{0, 1, 2, 3\}$$
$$\bullet = (\_ + \_) \text{ mod } 4$$

Identity: 0

$$(0 + 2) \text{ mod } 4 = 2$$
$$(2 + 0) \text{ mod } 4 = 2$$

Group Theory: Example

$$G = \{0, 1, 2, 3\}$$
$$\bullet = (\_ + \_) \bmod 4$$

Inverse: $-x$

$$\begin{aligned}(1 + (-1)) \bmod 4 &= (1 + 3) \bmod 4 \\ &= 4 \bmod 4 \\ &= 0\end{aligned}$$

Group Theory: Example

$$G = \{0, 1, 2, 3\}$$
$$\bullet = (\_ + \_) \bmod 4$$

Closure

$$(1 + 1) \bmod = 2$$
$$(2 + 3) \bmod = 5 \bmod 4 = 1$$

Group Theory: Example

$$G = \{0, 1, 2, 3\}$$
$$\bullet = (\_ + \_) \text{ mod } 4$$

Generators

$$G = \langle 1 \rangle = \langle 1, 2 \rangle$$

Permutation Groups

Points

$\Omega = \{0, 1, 2, \ldots, n\}$

Permutation

$\sigma \colon \Omega \to \Omega$

Permutation Groups

Permutation

$\sigma\colon \Omega \to \Omega$

Ex:

$$\begin{aligned} \sigma\colon 5 &\mapsto 7 \\ 7 &\mapsto 5 \\ 11 &\mapsto 42 \\ 42 &\mapsto 10 \\ 10 &\mapsto 11 \end{aligned}$$

(rest unchanged)

Permutation Groups

Permutation

$\sigma\colon \Omega \to \Omega$

Ex:

$$\begin{aligned} \sigma\colon 5 &\mapsto 7 \\ 7 &\mapsto 5 \\ 11 &\mapsto 42 \\ 42 &\mapsto 10 \\ 10 &\mapsto 11 \end{aligned}$$

(rest unchanged)

Cyclic notation

$$\sigma = (5\ 7)(11\ 42\ 10)$$

Tools from Group Theory

- Orbit
- Schreier-Sims algorithm
- ...

Tools from Group Theory

- Orbit

$$\mathrm{Orbit}_G(\omega) = \{g(\omega) \mid g \in G\}$$
$$\mathrm{Orbit}_G(1) = \{1, 2, 5\}$$

  Can be done on pairs too.

- Schreier-Sims algorithm

Tools from Group Theory

- Orbit
- Schreier-Sims algorithm
  - Membership testing in poly time
  - Element decomposition

Orbit Example

$$G = \left\langle \underbrace{(1\ 2)(3\ 4)}_{g_1}, \underbrace{(2\ 5)}_{g_2} \right\rangle, \quad \Omega = \{1, \ldots, 5\}$$

$$\begin{aligned} G = \{\ &(), (1\ 2)(3\ 4), (2\ 5), (3\ 4), (2\ 5)(3\ 4), (1\ 2), (1\ 2\ 5), \\ &(1\ 2\ 5)(3\ 4), (1\ 5\ 2), (1\ 5\ 2)(3\ 4), (1\ 5), (1\ 5)(3\ 4)\} \end{aligned}$$

$$\begin{aligned} \mathrm{Orbit}_G(\omega) &= \{g(\omega) \mid g \in G\} \\ \mathrm{Orbit}_G(1) &= \{1, 2, 5\} \end{aligned}$$

Can be done on pairs too.

Semigroups

Closure If $g, h \in G$, then $g \bullet h \in G$.

Associativity For all $g, h, k \in G$, then $(g \bullet h) \bullet k = g \bullet (h \bullet k)$.

~~**Identity** There exists $e \in G$ s.t. for all $g \in G$, then~~

$$\cancel{e \bullet g = g = g \bullet e}$$

~~**Inverse** For all $g \in G$, there exists $g^{-1} \in G$ s.t.~~

$$\cancel{g^{-1} \bullet g = e = g \bullet g^{-1}}$$

Semigroups

Closure If $g, h \in G$, then $g \bullet h \in G$.

Associativity For all $g, h, k \in G$, then $(g \bullet h) \bullet k = g \bullet (h \bullet k)$.

~~**Identity** There exists $e \in G$ s.t. for all $g \in G$, then~~

$$\cancel{e \bullet g = g = g \bullet e}$$

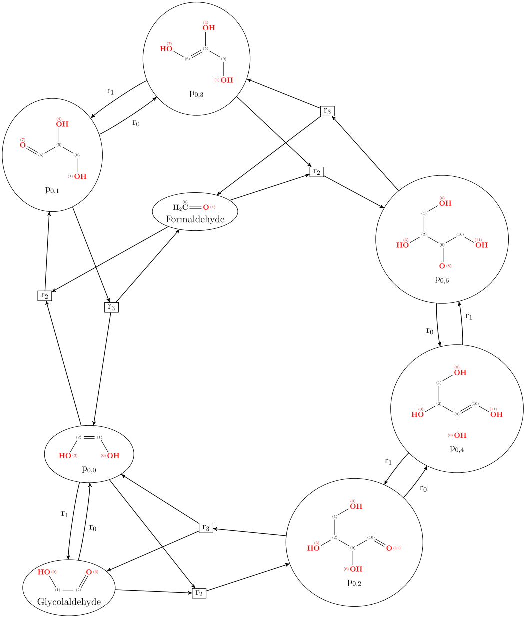
~~**Inverse** For all $g \in G$, there exists $g^{-1} \in G$ s.t.~~

$$\cancel{g^{-1} \bullet g = e = g \bullet g^{-1}}$$

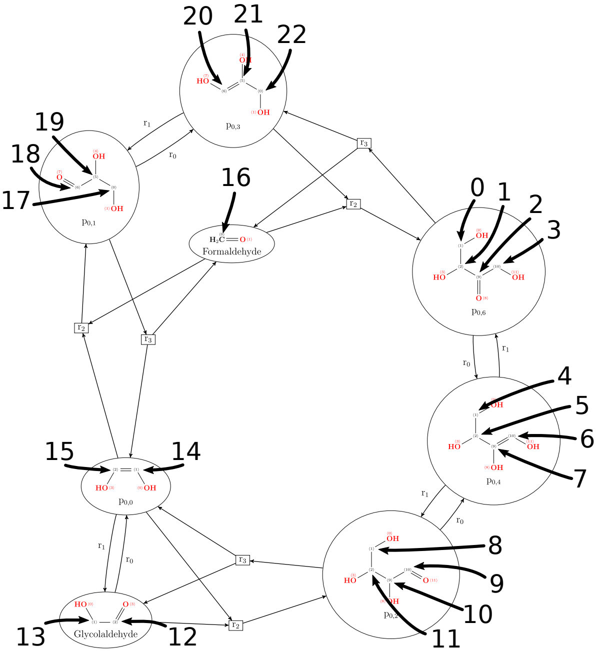
Semigroups of Transformations.

The Hypergraph-Semigroup Approach

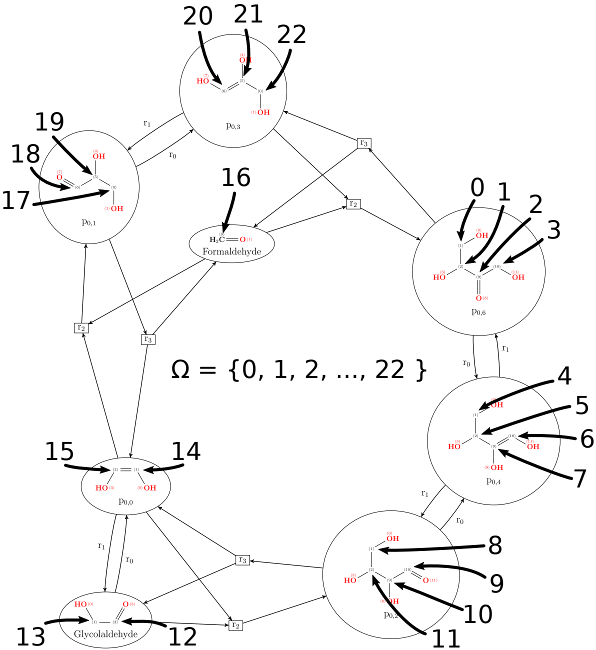
The Hypergraph-Semigroup Approach



The Hypergraph-Semigroup Approach

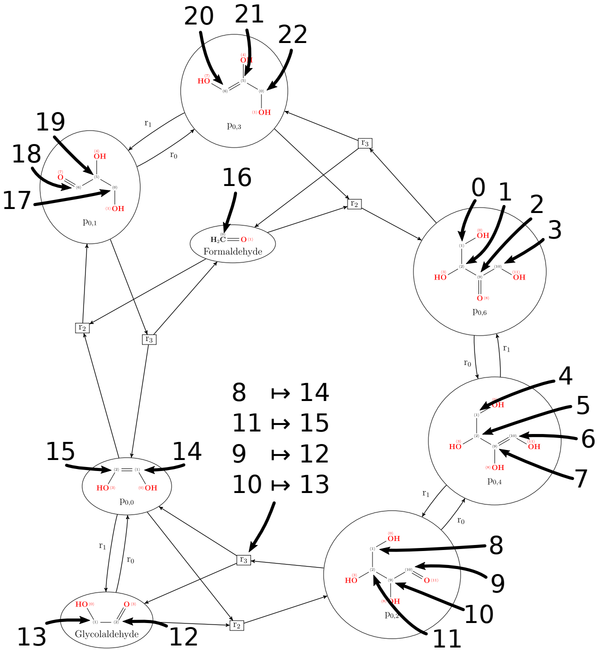


The Hypergraph-Semigroup Approach

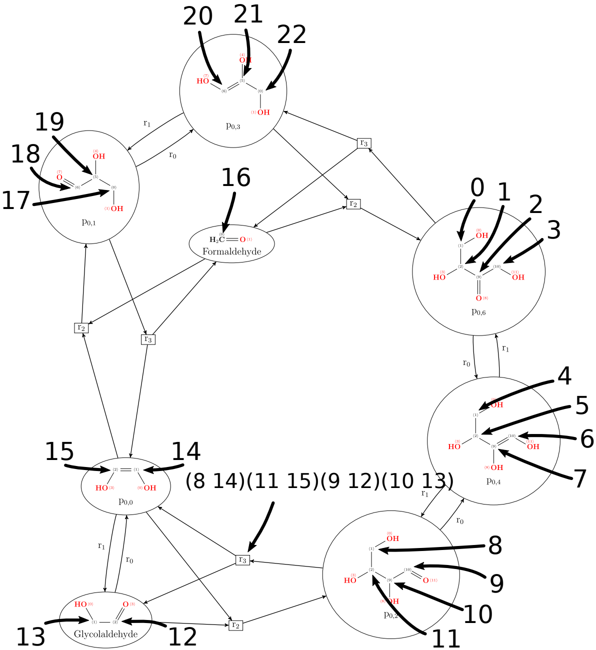


$\Omega = \{0, 1, 2, \ldots, 22\}$

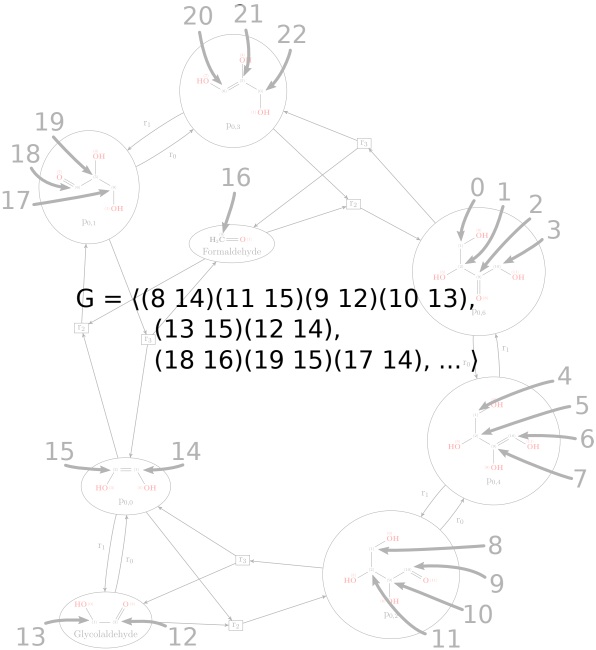
The Hypergraph-Semigroup Approach



The Hypergraph-Semigroup Approach



The Hypergraph-Semigroup Approach



$G = \langle (8\ 14)(11\ 15)(9\ 12)(10\ 13),$
$(13\ 15)(12\ 14),$
$(18\ 16)(19\ 15)(17\ 14), \ldots \rangle$

The Hypergraph-Semigroup Approach

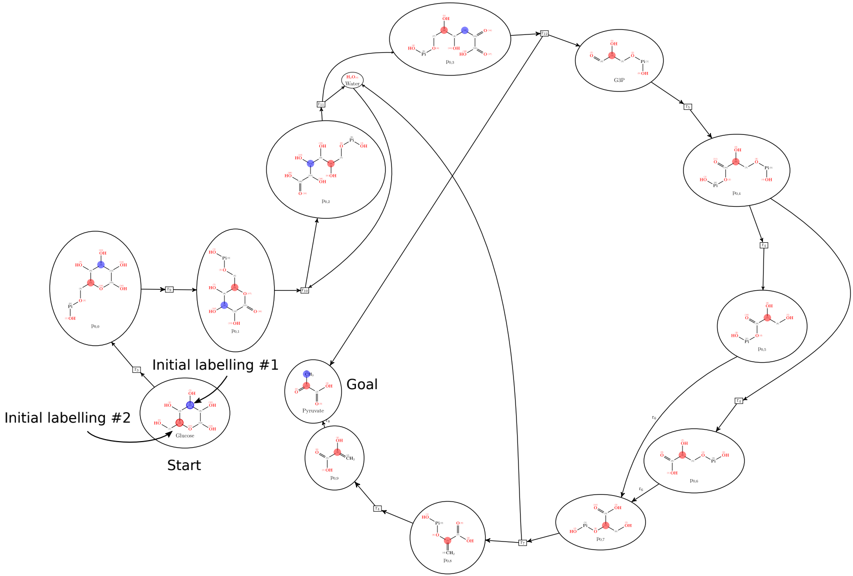
Definition

The *Hypergraph-Semigroup* of a Derivation Graph $H = (V, E)$ is a semigroup $G = \langle S \rangle$ acting on Ω, where

$$S = \bigcup_{e \in E} \mathrm{VertexMaps}(e)$$

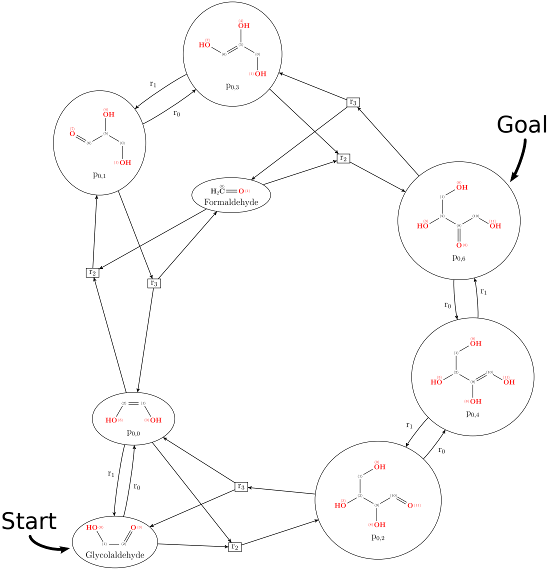
Orbits

The Hypergraph-Semigroup Approach



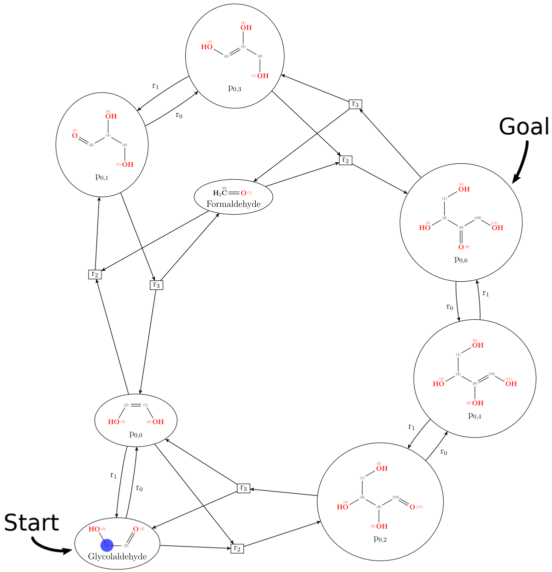
Orbits

The Hypergraph-Semigroup Approach



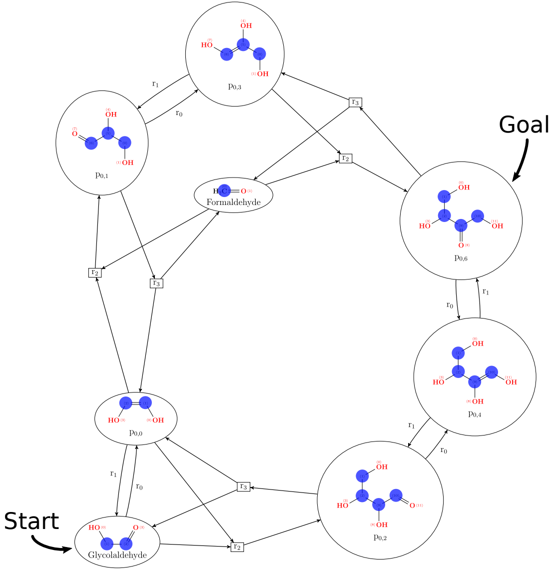
Orbits

The Hypergraph-Semigroup Approach



Orbits

The Hypergraph-Semigroup Approach

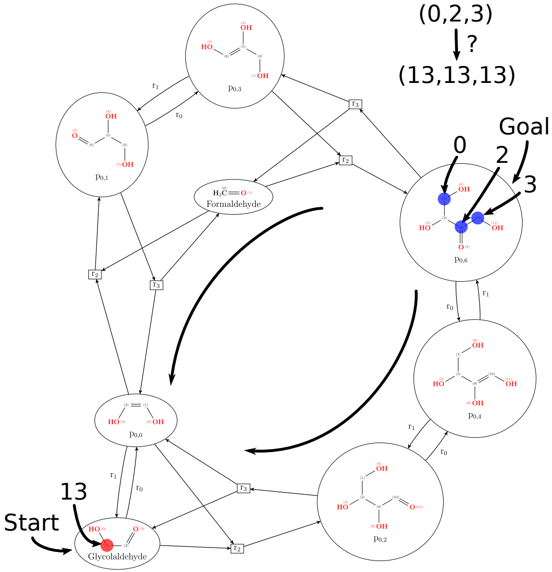


Hypothesis

Let G be a Hypergraph-Semigroup of some DG and let a single atom k be labelled. Suppose a molecule with a label at id i is observed in the laboratory. If $i \notin \mathrm{Orbit}_G(k)$, then the DG does not correctly describe the events happening in the laboratory.

Inverted Orbits

The Hypergraph-Semigroup Approach



Inverted Orbits

The Hypergraph-Semigroup Approach

Definition

Let $G = \langle T \rangle$ be a Hypergraph-Semigroup.
The *inverted Hypergraph-Semigroup* G^{-1} of G is a semigroup $G^{-1} = \langle X \rangle$ where

$$X = \{t^{-1} \mid t \in T\}$$

Inverted Orbits

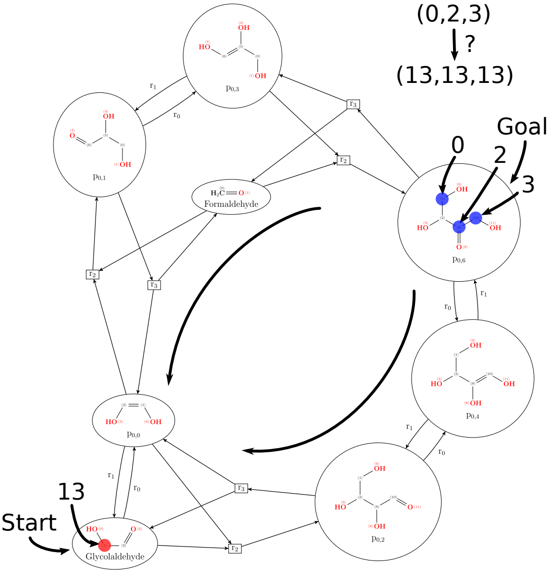
The Hypergraph-Semigroup Approach

Hypothesis

Let G be a Hypergraph-Semigroup of some DG and let a single atom k be labelled. Suppose the labelling $(i_1, \ldots, i_n)$ is observed in the laboratory. If $(k, \ldots, k) \notin \mathrm{Orbit}_{G^{-1}}((i_1, \ldots, i_n))$, then the DG does not correctly describe the events happening in the laboratory.

Inverted Orbits

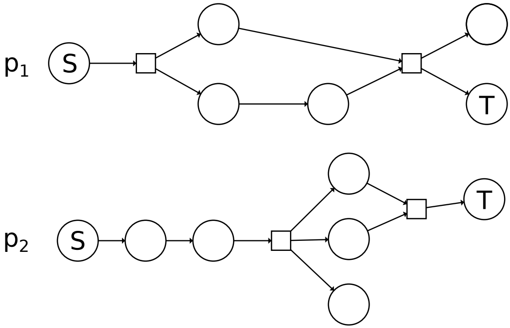
The Hypergraph-Semigroup Approach



Pathway Table

The Hypergraph-Semigroup Approach

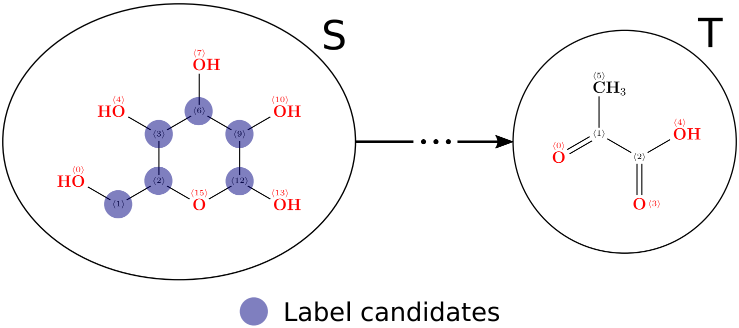
- Pathways P from start-molecule S to goal-molecule T.



Pathway Table

The Hypergraph-Semigroup Approach

- Pathways P from start-molecule S to goal-molecule T.
- Label candidates A.

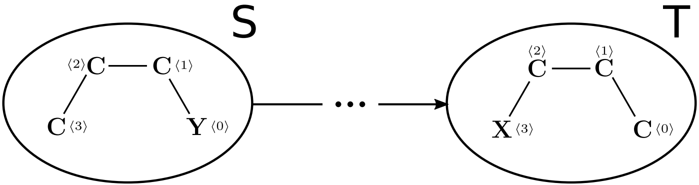


Pathway Table

The Hypergraph-Semigroup Approach

- Pathways P from start-molecule S to goal-molecule T.
- Label candidates A.
- Example:

Pathway \ Atom label	1, C	2, C	3, C
p_1	0, 2	1	0, 2
p_2			0, 1, 2

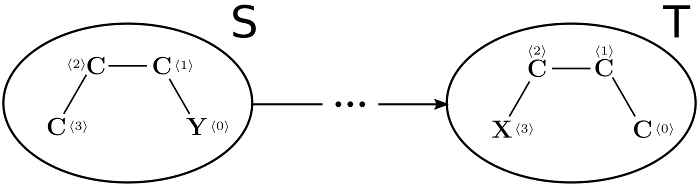


Pathway Table

The Hypergraph-Semigroup Approach

- Pathways P from start-molecule S to goal-molecule T.
- Label candidates A.
- Example:

Pathway \ Atom label	1, C	2, C	3, C
p_1	0, 2	1 (*)	0, 2 (*)
p_2			0, 1, 2 (*)



Pathway Table

The Hypergraph-Semigroup Approach

Pathway \ Atom label	0, C	1, C	2, C
p_1	0, 1, 3, 4	2	1, 3
p_2	0, 1, 2, 4	3	2, 4
p_3	1, 2, 3, 4	0	1, 2, 3, 4
p_4	0, 1, 2, 3, 4		

Pathway Table

The Hypergraph-Semigroup Approach

Pathway \ Atom label	0, C	1, C	2, C
p_1	0, 1, 3, 4	2	1, 3
p_2	0, 1, 2, 4	3	2, 4
p_3	1, 2, 3, 4	0	1, 2, 3, 4
p_4	0, 1, 2, 3, 4		

Pathway Table

The Hypergraph-Semigroup Approach

Pathway \ Atom label	0, C	1, C	2, C
p_1	0, 1, 3, 4	2	1, 3
p_2	0, 1, 2, 4	3	2, 4
p_3	1, 2, 3, 4	0	1, 2, 3, 4
p_4	0, 1, 2, 3, 4		

$$\mathrm{Orbit}_{G_{p_1}^{-1}}((1,3)) = \{\ldots,(0,2),(2,0)\}$$

$$\mathrm{Orbit}_{G_{p_3}^{-1}}((1,3)) = \{\ldots,(0,2),(2,2),(0,0),(2,0)\}$$

Pathway Comparison Table

The Hypergraph-Semigroup Approach

For each entry (p_i, p_j):

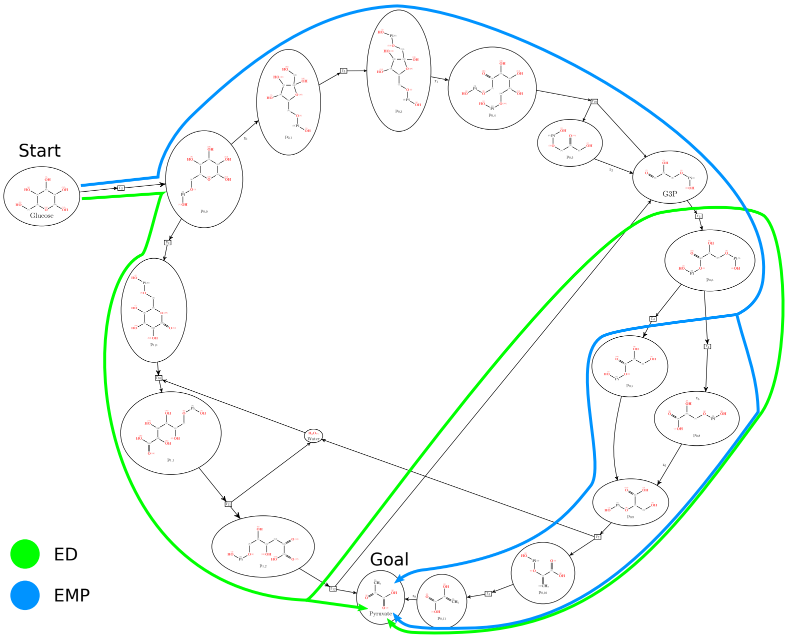
- Initialize entry (p_i, p_j) to be an empty list.
- Let $O_i = T[p_i, a]$ and $O_j = T[p_j, a]$. Similarly for O_j.
- Let $O = O_i \cap O_j$.
- Let $[O]^k =$ set of all subsets of O of size k.
- For each $t \in [O]^k$:
  - $d_i = \mathrm{Orbit}_{G_{p_i}^{-1}}(t)$.
  - $d_j = \mathrm{Orbit}_{G_{p_j}^{-1}}(t)$.
  - If d_i contains $(\underbrace{a, \ldots, a}_{k})$ and d_j does not, add t to (p_i, p_j).

Pathway Comparison Table

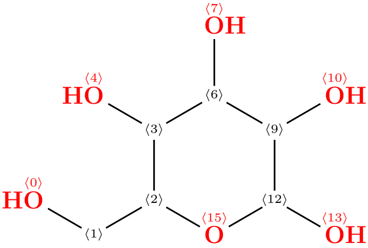
The Hypergraph-Semigroup Approach

Contains \ Not contains	1	2	3	4
1				
2				
3	$(1, 3)$	$(2, 4)$		
4	$(1, 3)$	$(2, 4)$		

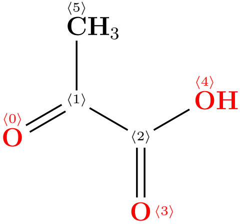
Example: Glycolysis



Example: Glycolysis



(a) Glucose



(b) Pyruvate

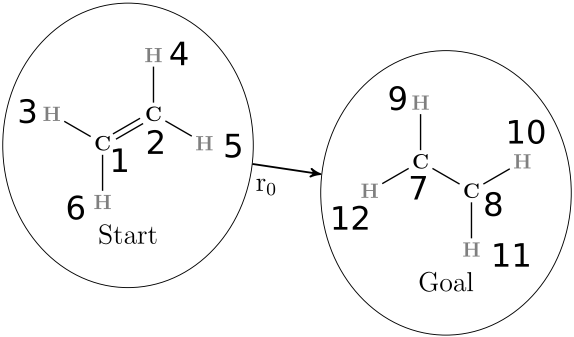
Example: Glycolysis

Pathway \ Atom label	0, O	6, C	7, O	12, C	13, O	15, O
EMP		2	3	5		0
ED	0	5	0	2	3	4
Both	0	2, 5	0, 3	2, 5	0, 3	0, 4

Demo

Vertex Map Optimization

Vertex Map Optimization



Rule Composition

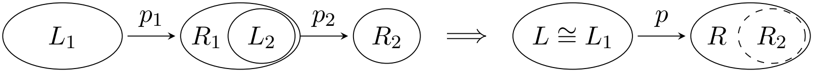


Figure 3: Abstract depiction of full rule composition $p_1 \bullet_{\supseteq} p_2$.

Double Pushout Diagram

Hyperedge with rule $p = (L \stackrel{l}{\leftarrow} K \stackrel{r}{\rightarrow} R)$.

$$\begin{array}{ccccc} L & \stackrel{l}{\longleftarrow} & K & \stackrel{r}{\longrightarrow} & R \\ \downarrow & & \downarrow & & \downarrow \\ G & \longleftarrow & D & \longrightarrow & H \end{array}$$

Rule Comp. Approach

Goal: Given $e = (e^+, e^-)$, determine $\mathrm{VertexMaps}(e)$.
Also given $p = (L \xleftarrow{l} K \xrightarrow{r} R)$, G and H.

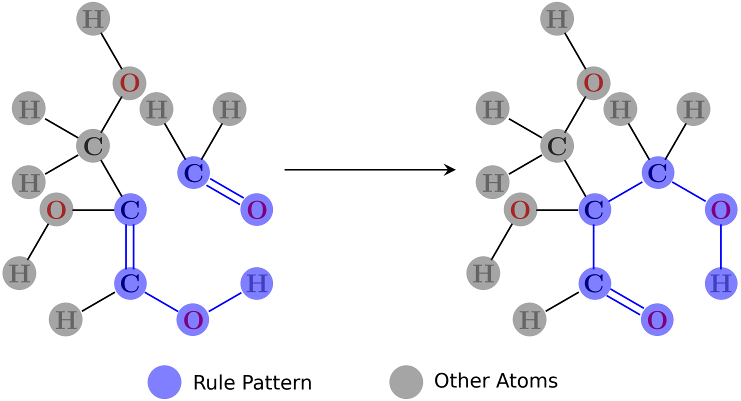
- Let $p_G = (G \leftarrow G \rightarrow G)$ and $p_H = (H \leftarrow H \rightarrow H)$.
- Compute $P' = p_G \bullet_{\supseteq} p$.
- For each $p' \in P'$, compute $P'' = p' \bullet_{\subseteq} p_H$.
- Compute:

$$\begin{aligned}\mathrm{VertexMaps}(e) = \{m_{GL'} \circ l'^{-1} \circ r' \circ m_{R'H} \mid\ & (L' \xleftarrow{l'} K' \xrightarrow{r'} R') \in X, \\ & m_{GL'} \in M_{GL'}, \\ & m_{R'H} \in M_{R'H}\}\end{aligned}$$

where $M_{GL'}$ is the set of isomorphisms from G to L' and $M_{R'H}$ is the set of isomorphisms from R' to H.

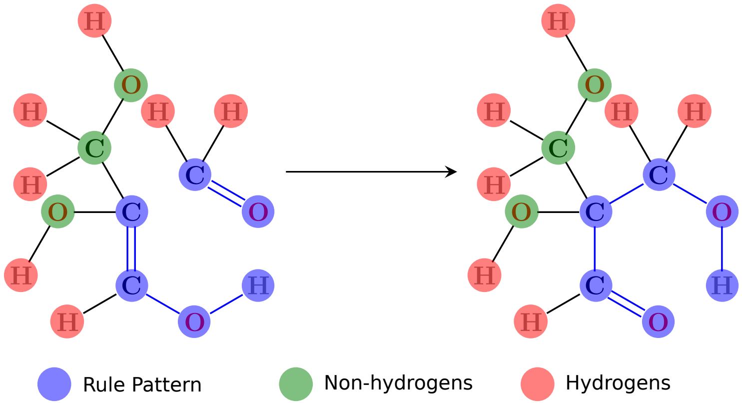
The New Approach

1 stage



The New Approach

2 stage



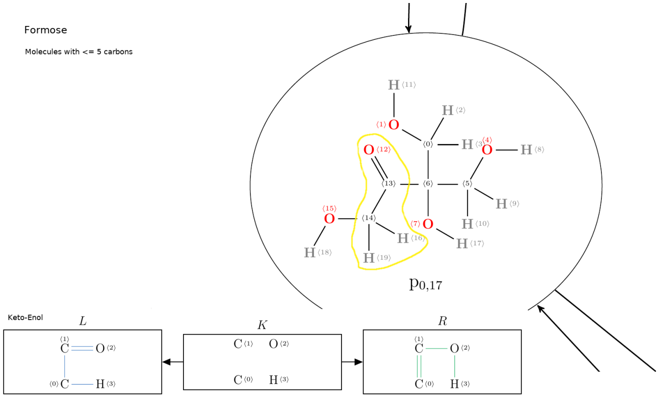
The New Approach

Advantages

- More direct
- Flexible
- Easy to extend

The New Approach

Minor Issue



Results

- Formose
- Glycolysis
- PPP

Results

- Formose
- Glycolysis
- PPP

Test (in Python):

1. Compute DG.

Results

- Formose
- Glycolysis
- PPP

Test (in Python):

1. Compute DG.
2. Compute Hypergraph-Semigroup via rule comp. approach.

Results

- Formose
- Glycolysis
- PPP

Test (in Python):

1. Compute DG.
2. Compute Hypergraph-Semigroup via rule comp. approach.
3. Compute Hypergraph-Semigroup via new approach.

Results

- Formose
- Glycolysis
- PPP

Test (in Python):

1. Compute DG.
2. Compute Hypergraph-Semigroup via rule comp. approach.
3. Compute Hypergraph-Semigroup via new approach.
4. Repeat 5 times.

Formose

Results

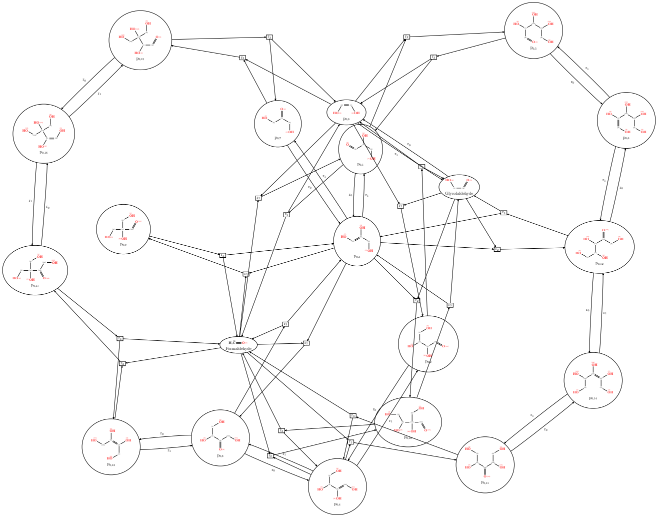


Figure 4: 46 hyperedges

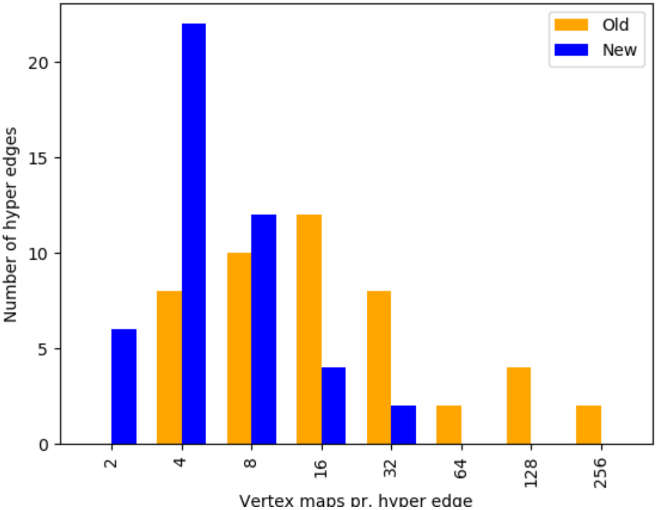
Timing (average):

Rule Comp.	0.69 s
New	0.19 s

Speed-up: ≈ 3.5

Formose

Results



Glycolysis
Results

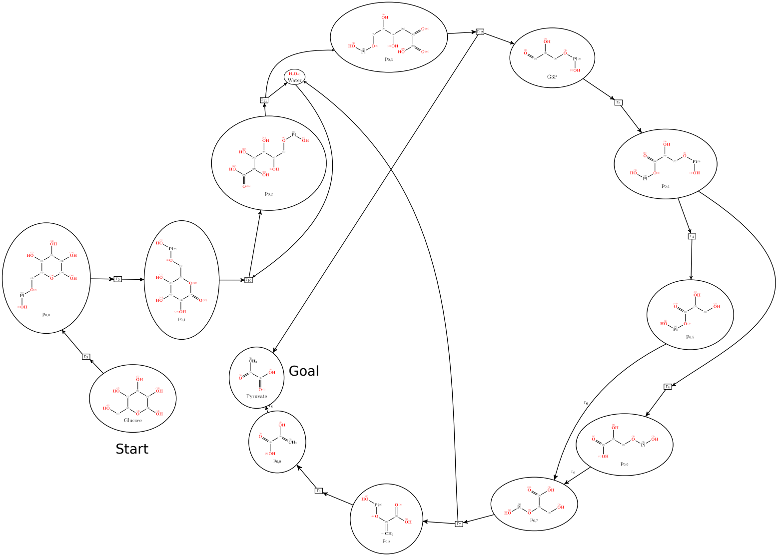


Figure 4: EMP – 19 hyperedges

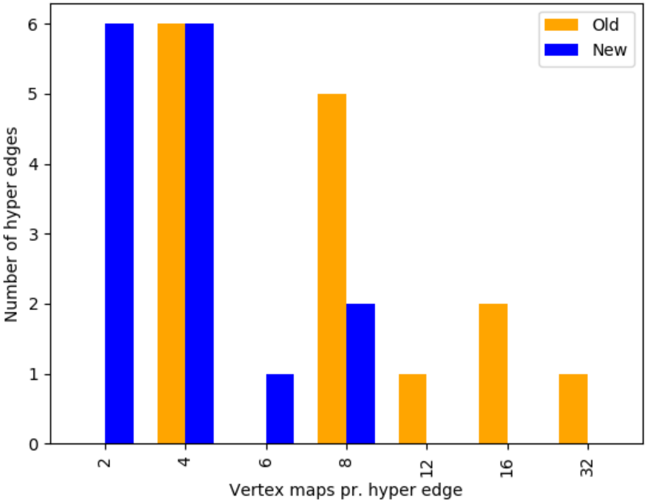
Glycolysis
Results

Timing (average):

Rule Comp.	0.10 s
New	0.05 s

Speed-up: ≈ 2

Glycolysis
Results



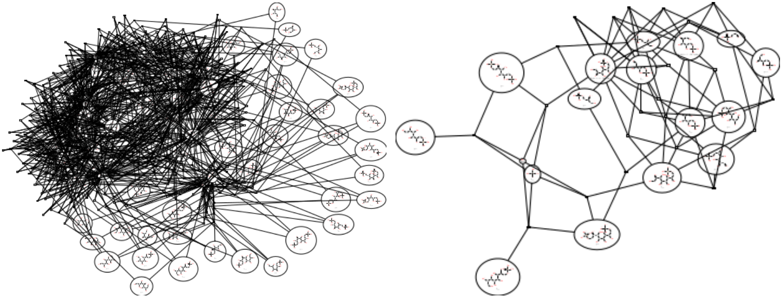


Figure 4: Normal: 333 hyperedges. Strict: 24 hyperedges.

Timing (average):

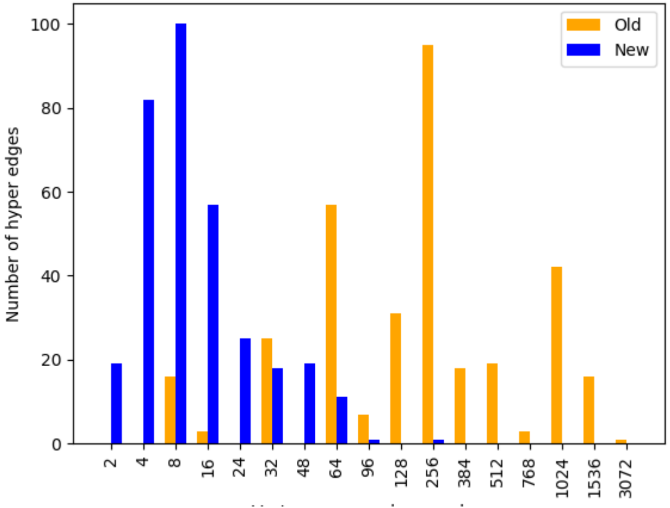
	Normal [s]	Strict [s]
Rule Comp.	135.34	30.14
New	8.08	1.01

Speed-up:

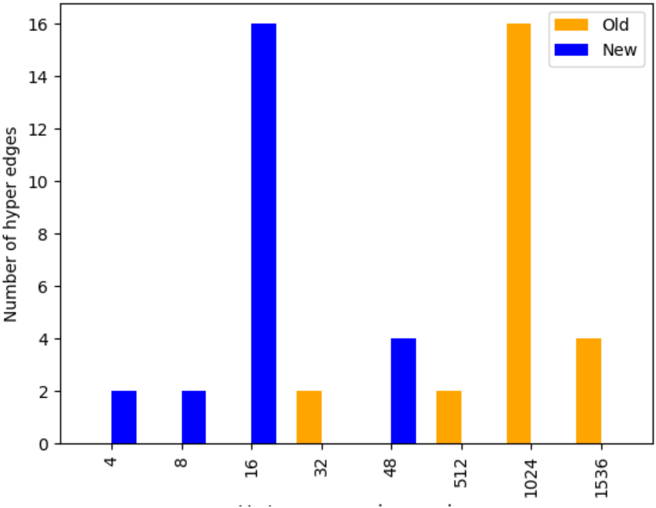
- Normal: ≈ 16
- Strict: ≈ 30

PPP

Results



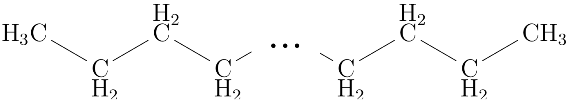
Number of hyper edges
100
80
60
40
20
0
Old
New
2
4
8
16
24
32
48
64
96
128
256
384
512
768
1024
1536
3072


Number of hyper edges
16
14
12
10
8
6
4
2
0
Old
New
4
8
16
32
48
512
1024
1536

Linear Molecules

Results

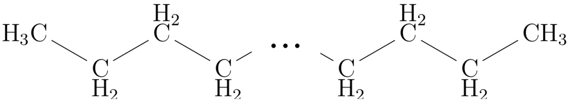
N carbons



Linear Molecules

Results

N carbons



$$\underbrace{C - C - \cdots - C}_{k} \rightarrow \underbrace{C = C = \cdots = C}_{k}$$

Linear Molecules

Results

- SMALL: $k = 2$.
- MEDIUM: $k = \lfloor N/2 \rfloor$.
- LARGE: $k = N - 1$.
- FULL: $k = N$.

Linear Molecules

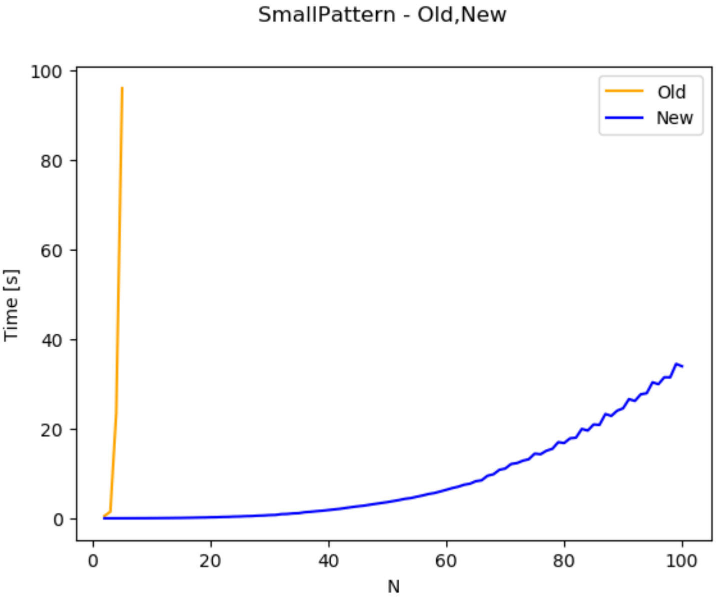
Results

- Small: $k = 2$.
- Medium: $k = \lfloor N/2 \rfloor$.
- Large: $k = N - 1$.
- Full: $k = N$.

Run rule comp. for $N \in \{2, \ldots, 5\}$.
Run new for $N \in \{2, \ldots, 100\}$.

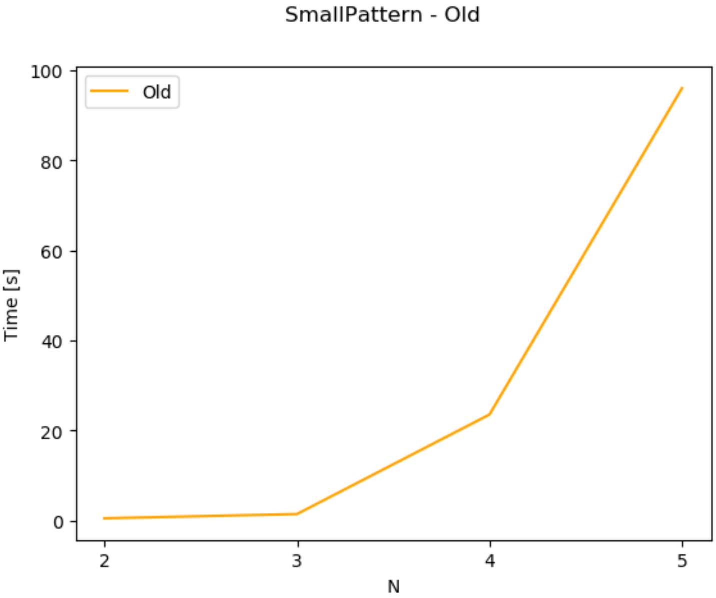
Linear Molecules

Results



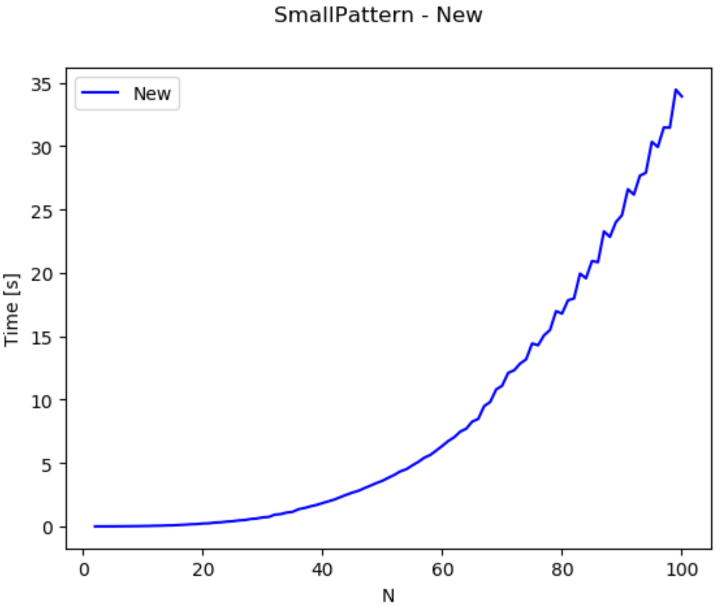
Linear Molecules

Results



Linear Molecules

Results



For $N = 5$:

Pattern	Speed-up factor
Small	18439
Medium	18062
Large	18690
Full	39417

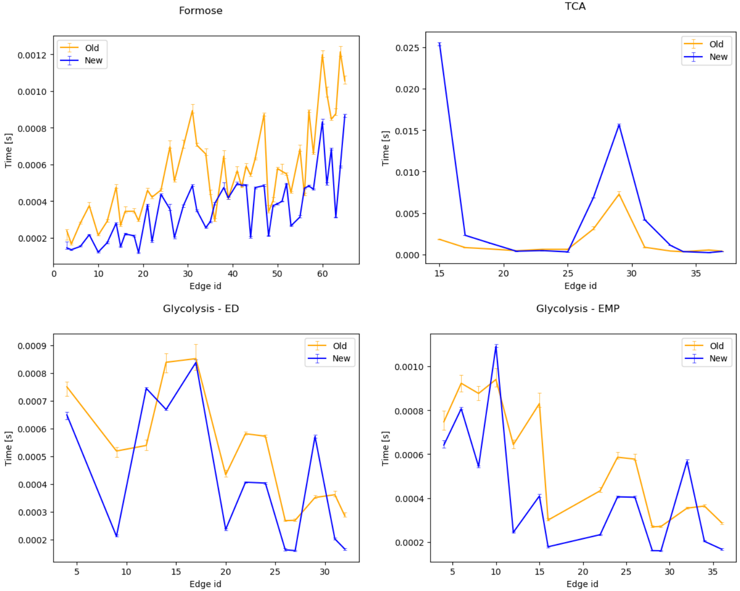
Direct Comparison

Results

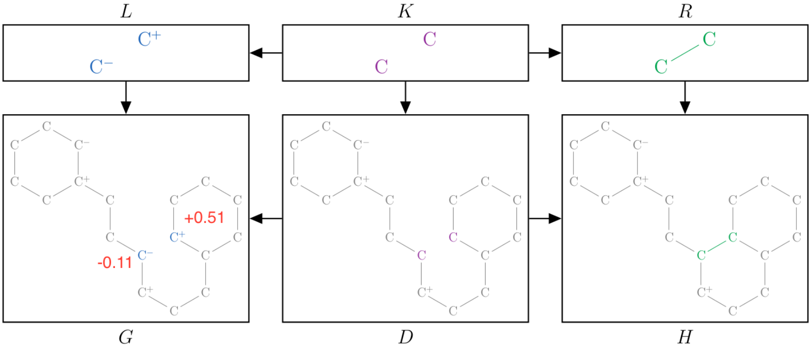
- Python $\leftrightarrow$ C++ overhead.
- Remove filtering of hydrogens.
- Do timing in C++.
- Only time vertex map construction.
- Repeat 20 times.

Direct Comparison

Results



Other Uses



Demo

Concluding Remarks

Concluding Remarks

- Hypergraph-Semigroup $\rightarrow$ Orbits $\rightarrow$ Pathway Tables.

Concluding Remarks

- Hypergraph-Semigroup $\rightarrow$ Orbits $\rightarrow$ Pathway Tables.
- A working Python framework.

Concluding Remarks

- Hypergraph-Semigroup $\rightarrow$ Orbits $\rightarrow$ Pathway Tables.
- A working Python framework.
- New approach to vertex maps.

Concluding Remarks

- Hypergraph-Semigroup $\rightarrow$ Orbits $\rightarrow$ Pathway Tables.
- A working Python framework.
- New approach to vertex maps.
- Flexible and fast.

Concluding Remarks

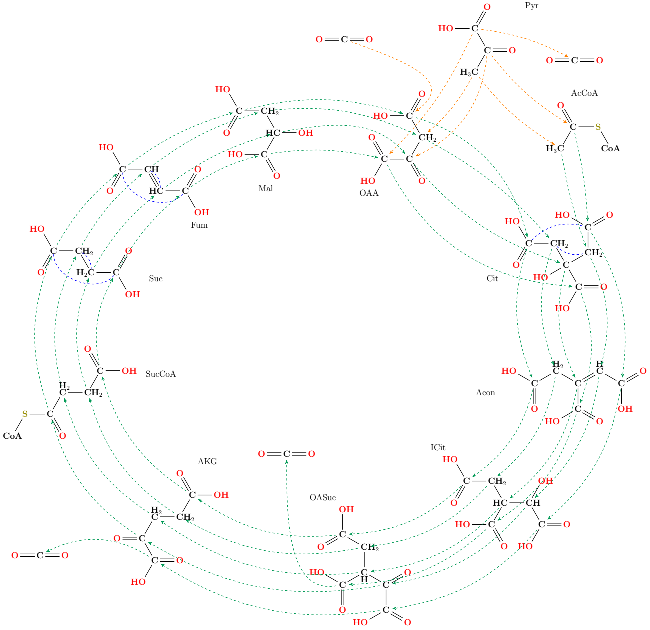
- Hypergraph-Semigroup $\rightarrow$ Orbits $\rightarrow$ Pathway Tables.
- A working Python framework.
- New approach to vertex maps.
- Flexible and fast.
- Modifications to core MØD parts.

Concluding Remarks

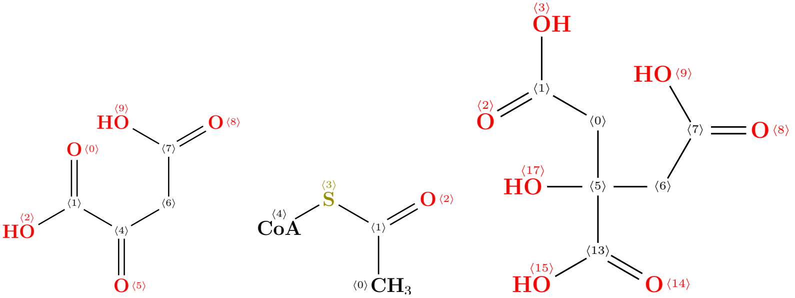
- Hypergraph-Semigroup $\rightarrow$ Orbits $\rightarrow$ Pathway Tables.
- A working Python framework.
- New approach to vertex maps.
- Flexible and fast.
- Modifications to core MØD parts.
- Many future applications.

Thank You!

Example: TCA



Example: TCA



(a) OAA (b) AcCoA (c) Cit

Example: TCA

Pathway \ Atom label	0, C	1, C	5, C
TCA	0, 1, 5, 6, 7, 13	1, 7, 13	0, 1, 5, 6, 7, 13

Pathway \ Atom label	6, C	7, C	13, C
TCA	0, 1, 5, 6, 7, 13	1, 7, 13	1, 7, 13