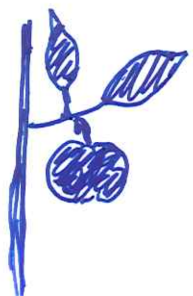
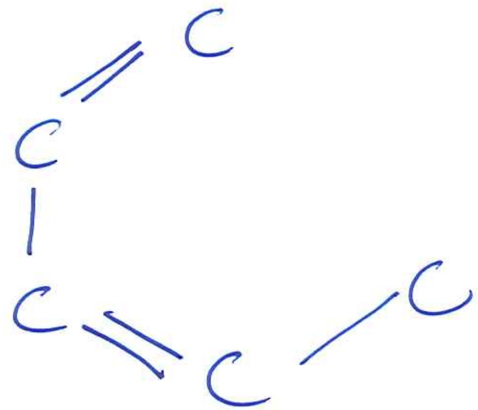
Decomposition of Chemical Reactions

UFFE THORSEN et al

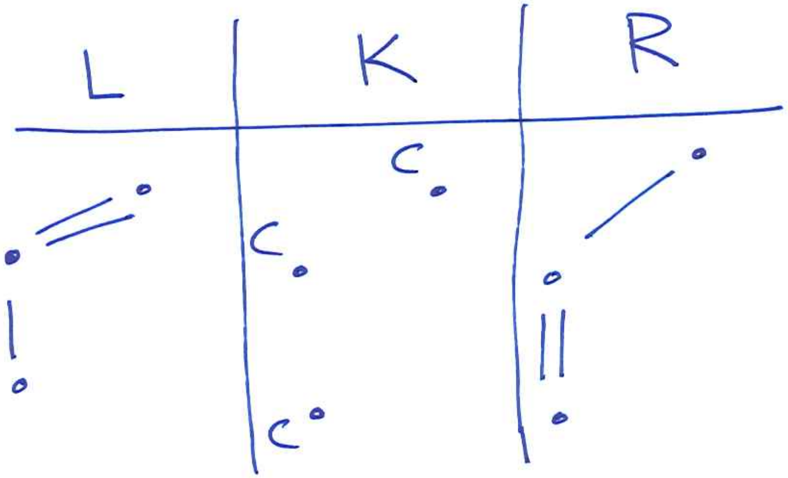


Double Pushout

Molecules as labeled graphs

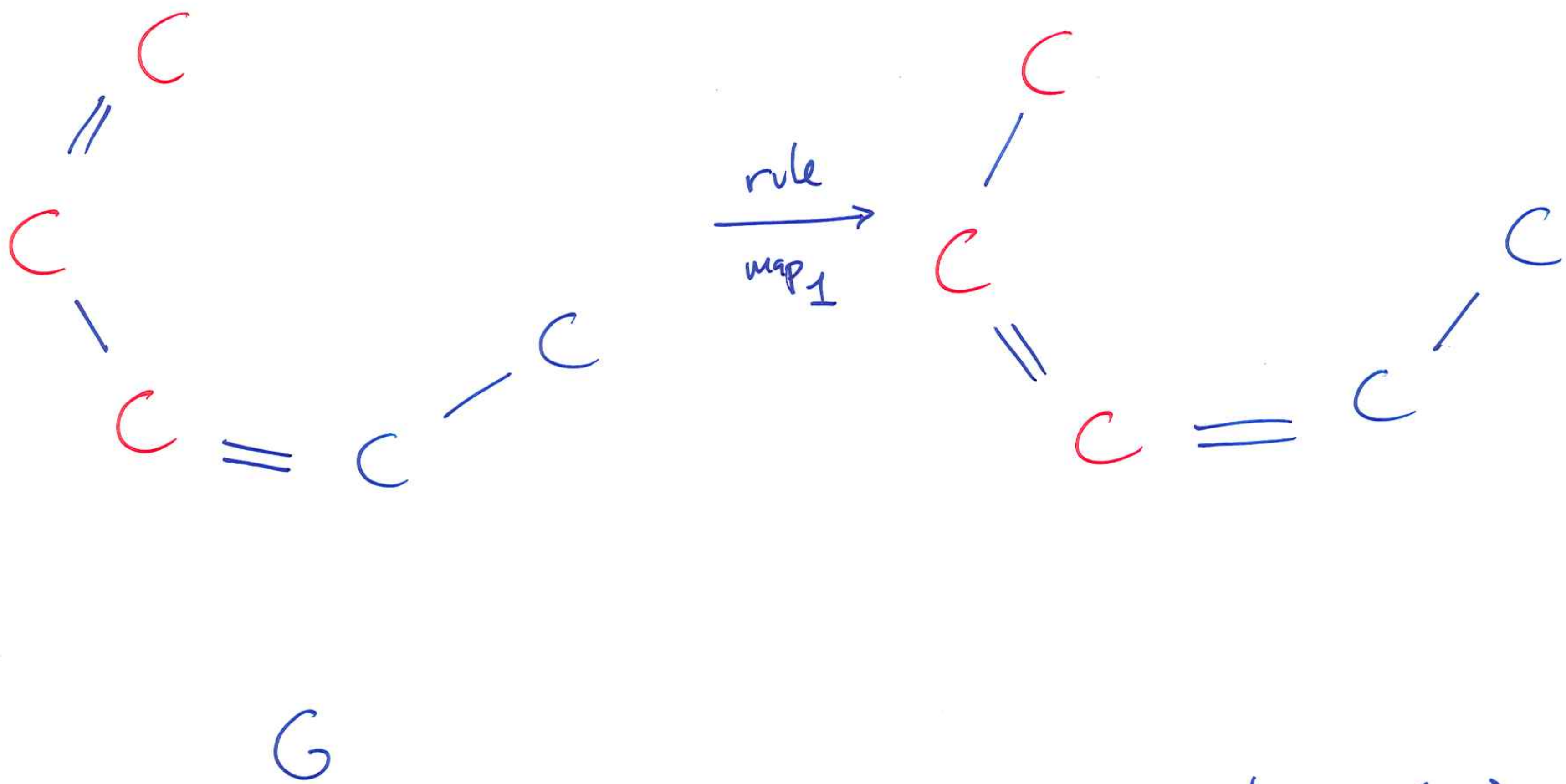


Rules



Double Pushout

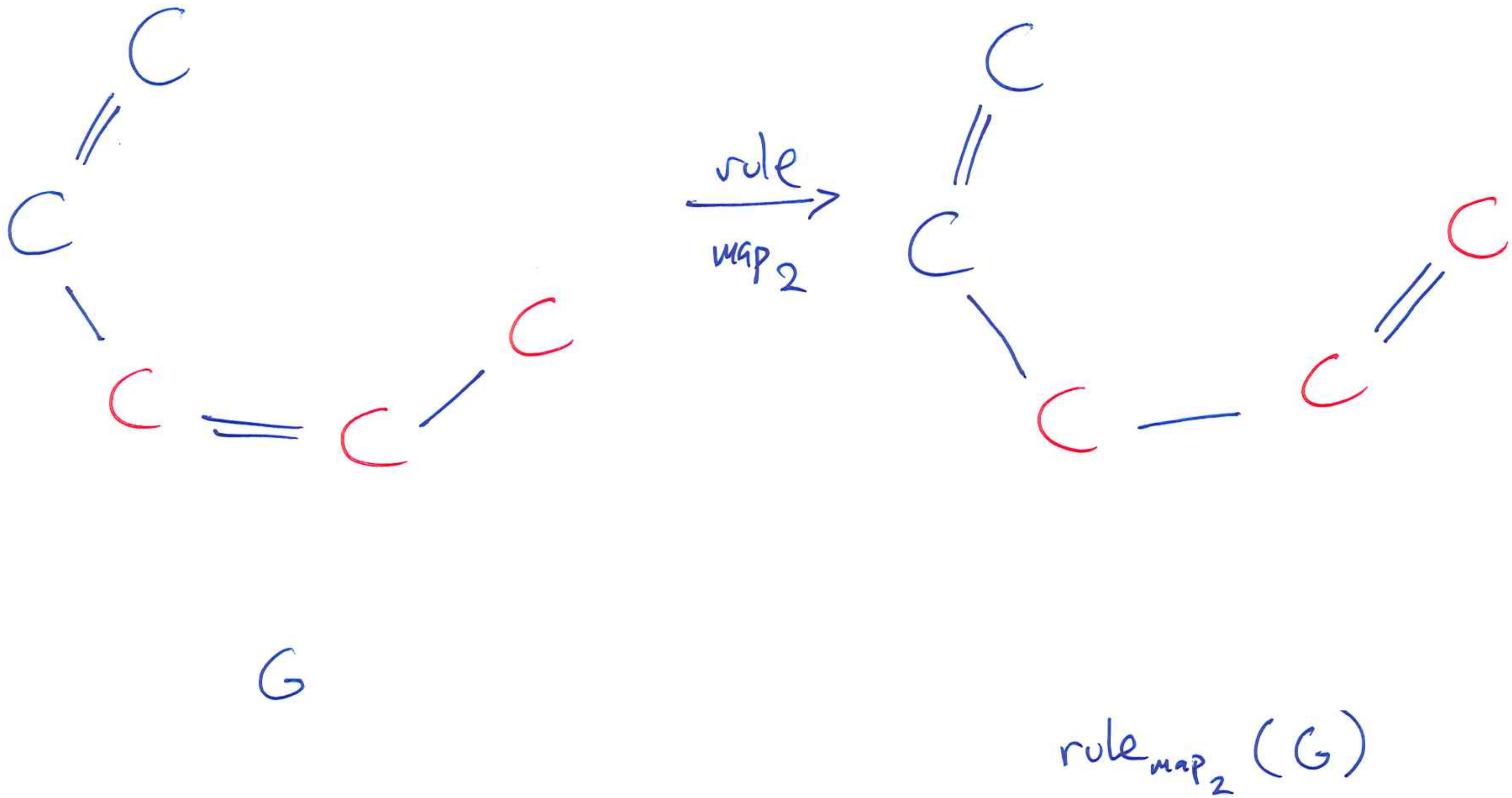
Application of a rule can give several new graphs, depending on mapping.



$rule_{map_1}(G)$

Double Pushout

Application of a rule can give several new graphs, depending on mapping.

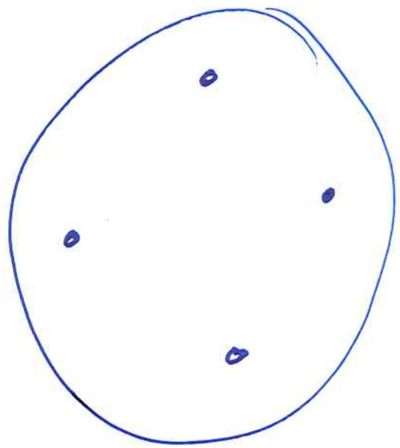


Exploring a Chemical Space

Given: A set of rules
an initial set of molecules

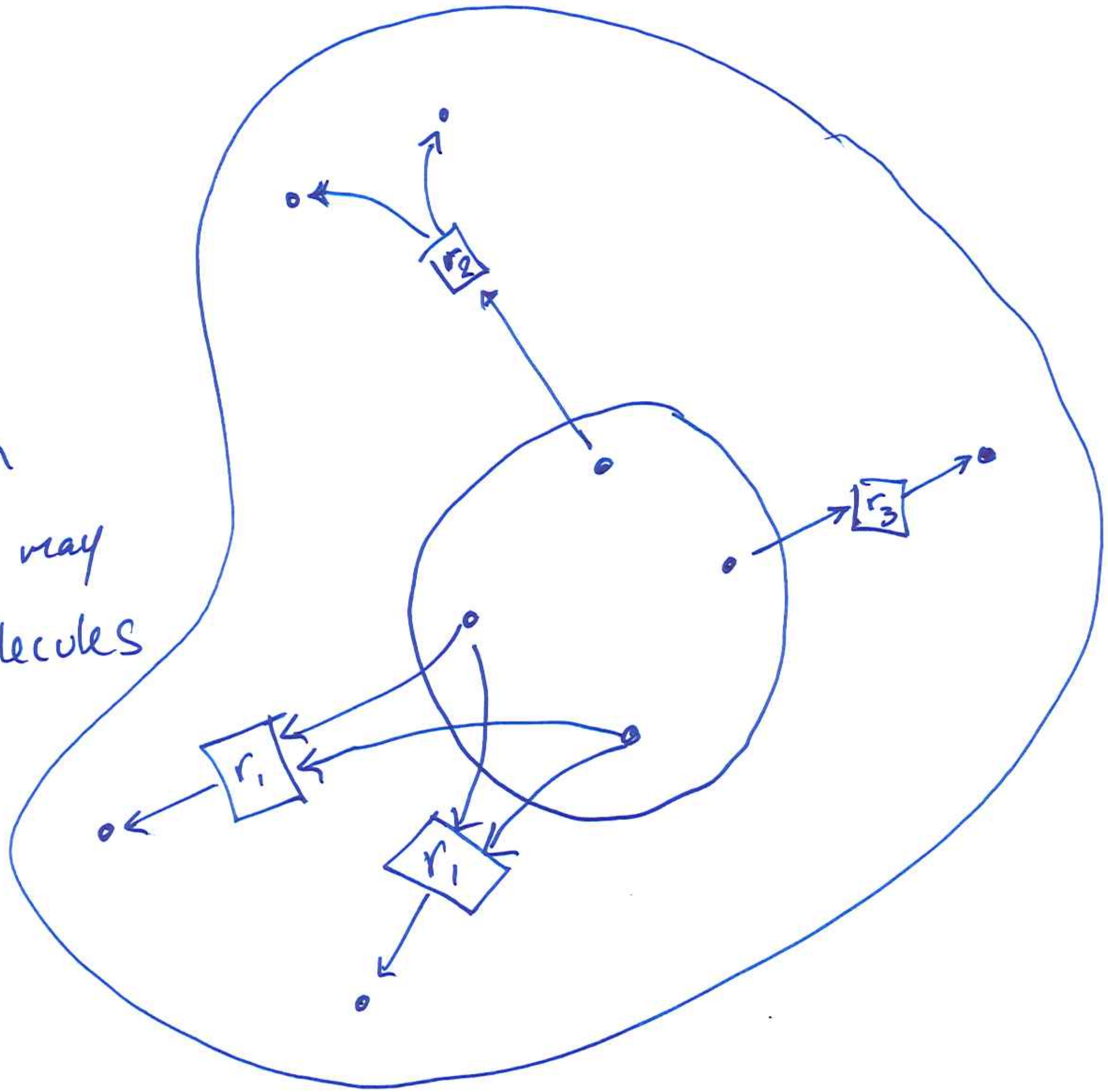
Explore: Possible applications of rules and thereby new molecules] repeat

Note: Each vertex is a single molecule



Exploring a Chemical Space

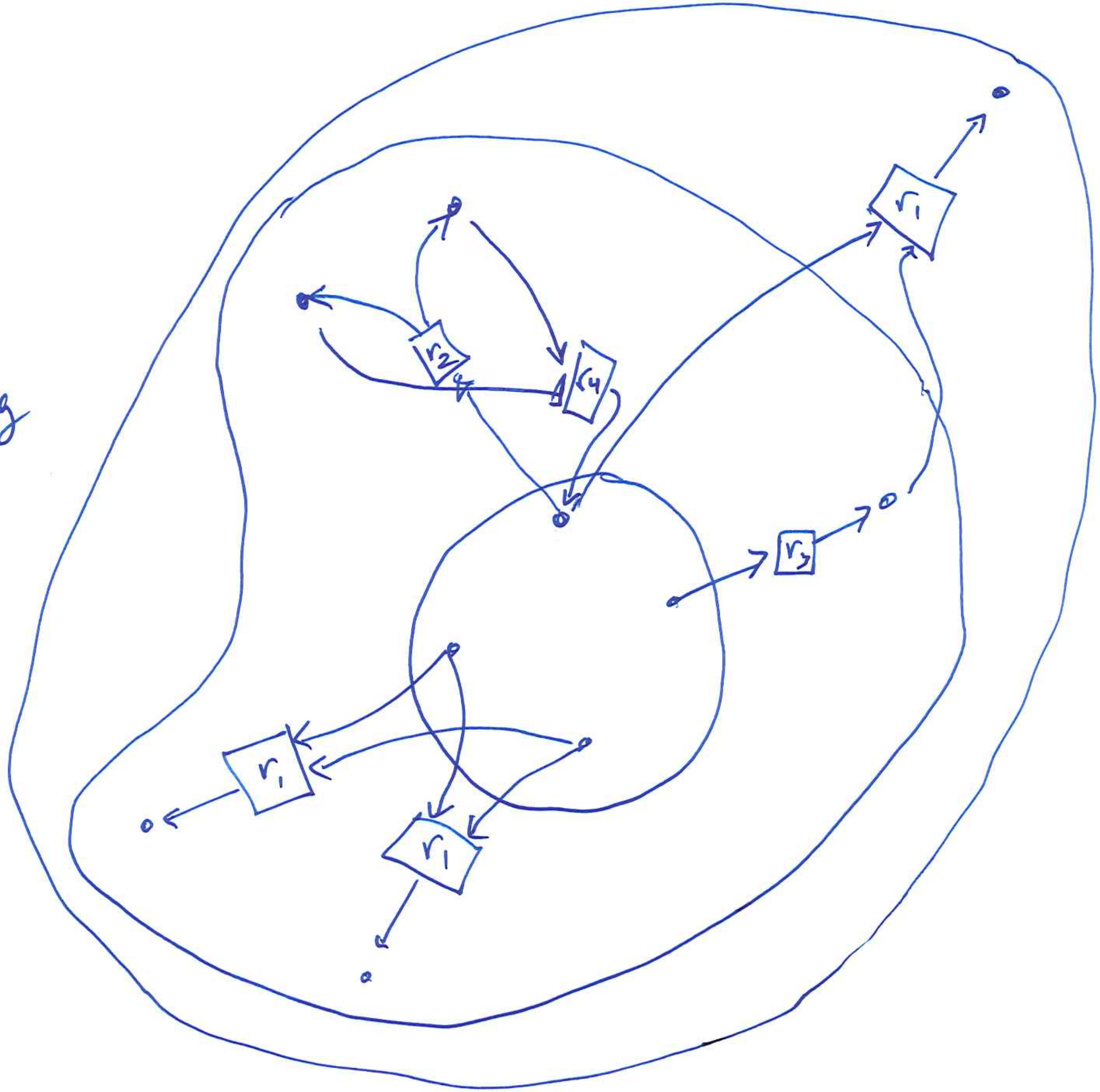
Apply all rules to all current molecules



Note: A dihypergraph as reactions may split/merge molecules

Exploring a Chemical Space

Repeat



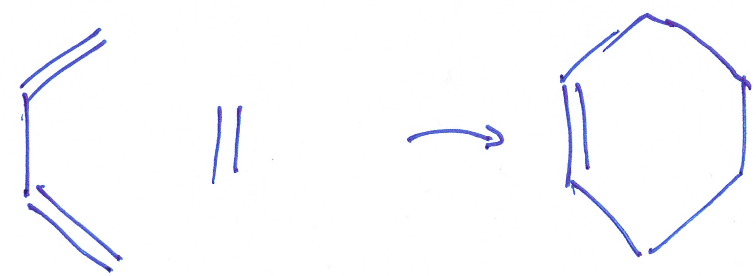
May converge, depending on rules (unlikely ~~depending~~ without additional constraints).

Atom Atom Mapping

Given: chemical reaction $E \to P$

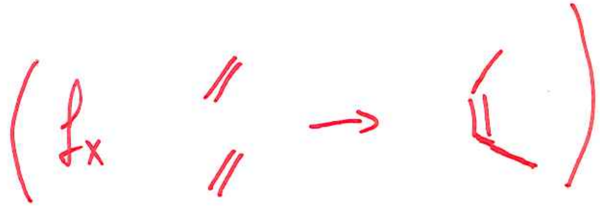
ψ is a map from atoms/vertices of E
to atoms/vertices of P

EX



where ψ is implied
by position

NOW: Rule Decomposition

Given: A rule/reaction $E \rightarrow P$ 

where E and P are (disconnected) graphs.

And a set of elementary rules ER.

(possibly with an atom atom mapping φ)

How to get from E to P?

WHY?

- To find / check atom atom mapping
- To infer context / minimize context of the rule
- To understand chemistry / develop theories

Requires:

- elementary Rules ER are chemically valid

Rule Decomposition

DEF: Given a rule $E \to P$
and a set of rules ER
Find a sequence of rules and maps

$$\langle (r_1, m_1), (r_2, m_2), \ldots, (r_k, m_k) \rangle$$

s.t.

$$r_{k_{m_k}}(r_{k-1_{m_{k-1}}}(\ldots (r_{1_{m_1}}(E)) \ldots)) = P$$

OR equivalently a sequence of graphs

$$\langle E, G_1, G_2, \ldots, G_{k-1}, P \rangle$$

s.t. $G_i \to G_{i+1}$ corresponds to a rule in ER

Rule Decomposition

Can think of it as a path in a digraph.

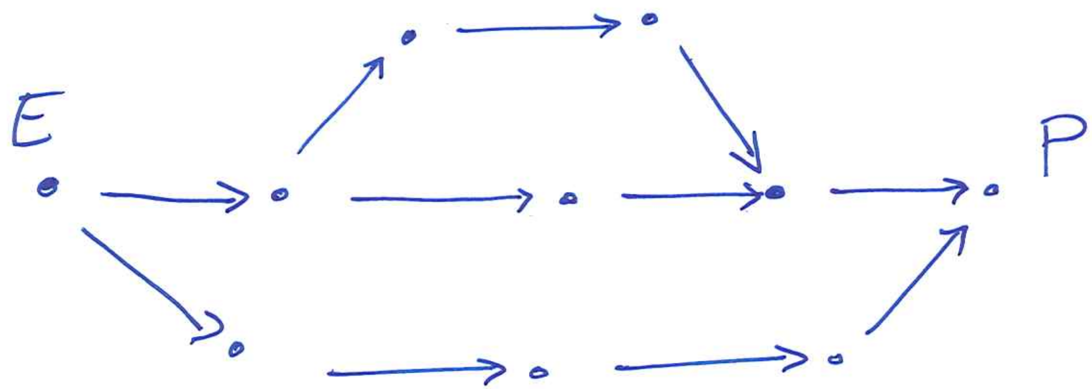
$$E \xrightarrow{r_1, m_1} G_1 \xrightarrow{r_2, m_2} G_2 \xrightarrow{r_3, m_3} G_3 \xrightarrow{r_4, m_4} P$$

Note: vertices are collection of molecules.

Note: Not a hypergraph, mass is preserved.

Rule Decomposition

... or many/all such paths

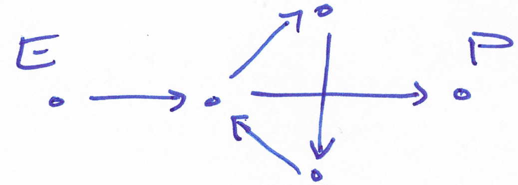


Decomposition graph: Contains several/all decompositions of $E \to P$

Still Underspecified

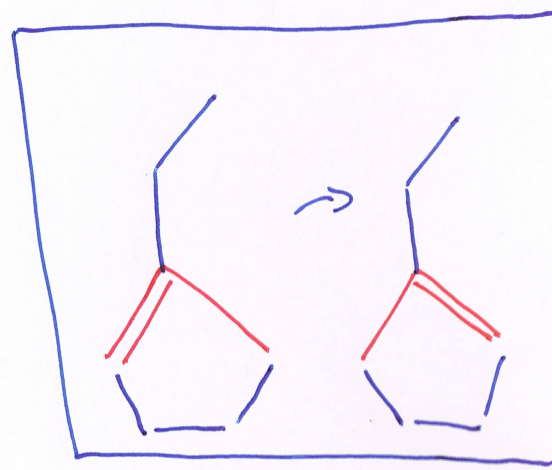
- Do we allow non-simple paths?

  No



- When are two graphs identical?
  - → Isomorphism
  - Equality under atom atom map φ

- What rule applications are allowed?
  - Catalysts not in $E \to P$
  - → All of $E \to P$
  - All vertices changed under some φ
  - All edges changed under some φ

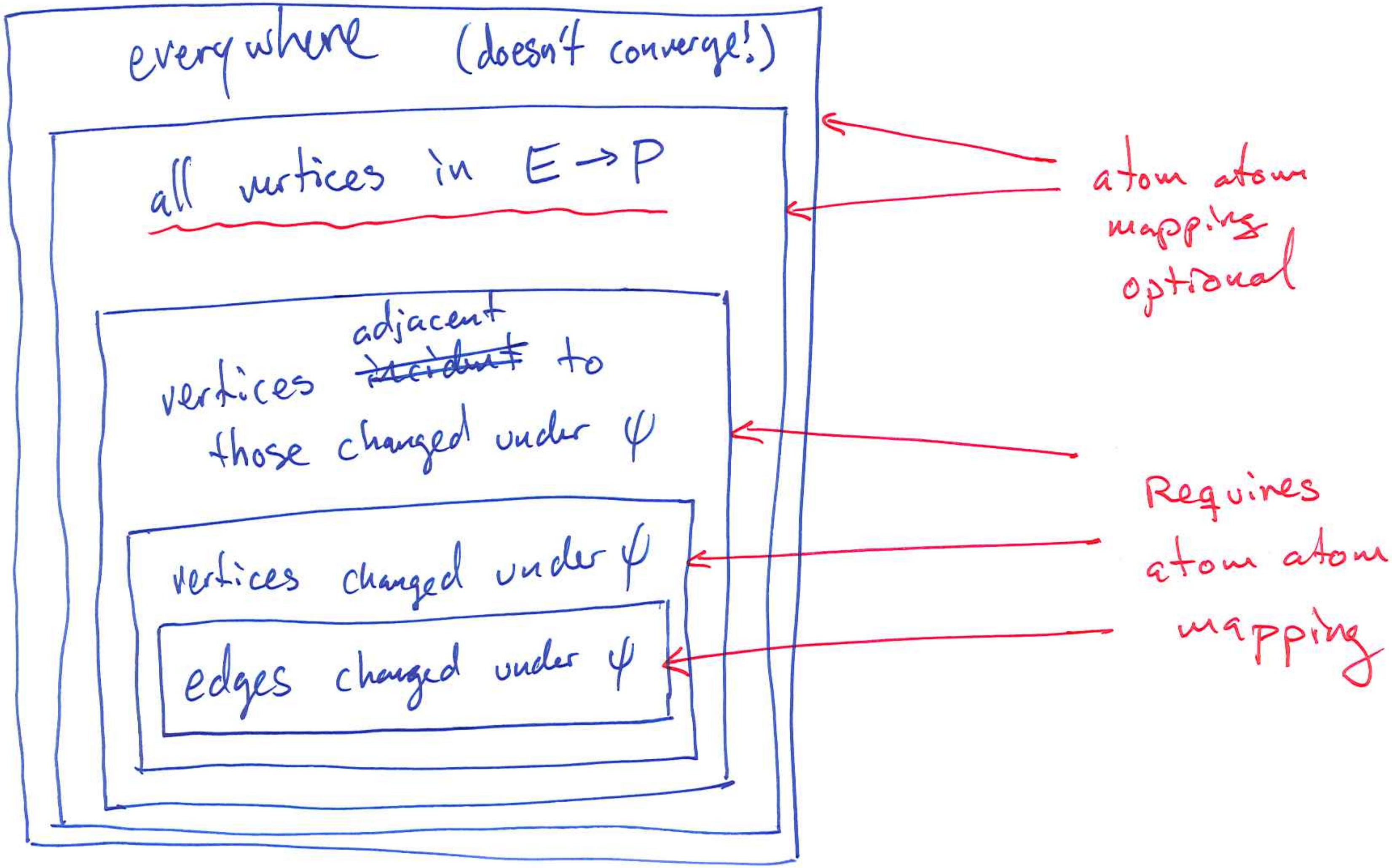


(And technical details ...)

Variant Problems

Where can non-context parts of rules be mapped?

Similar choice for context



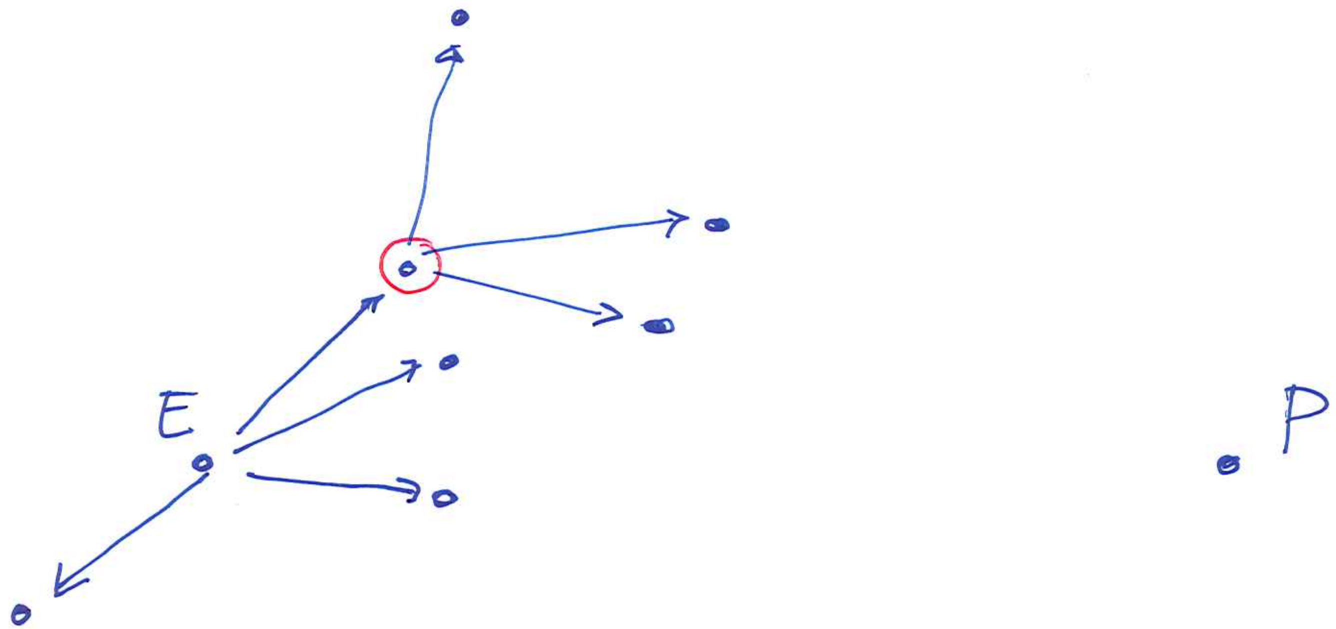
How to Solve It?

Start with a digraph containing only E and P

How to Solve It?

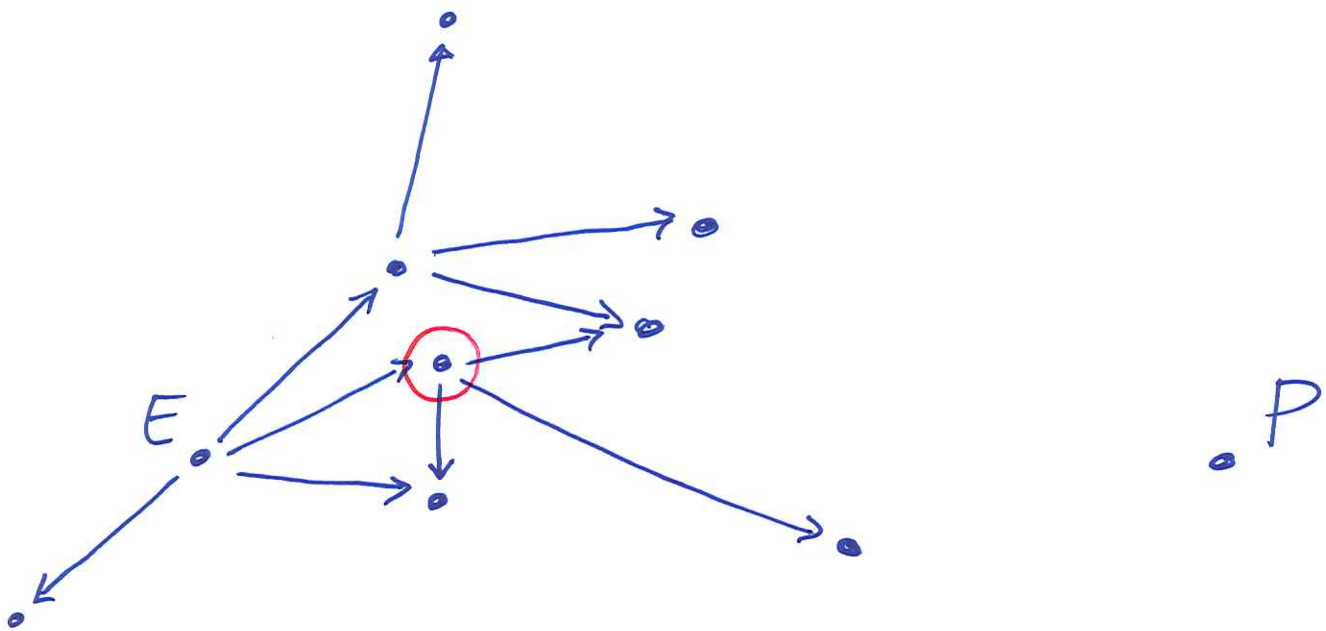
Explore out edges of E
Explore each rule in ER
Remember: each rule may give several new graphs (vertices)

How to Solve It?



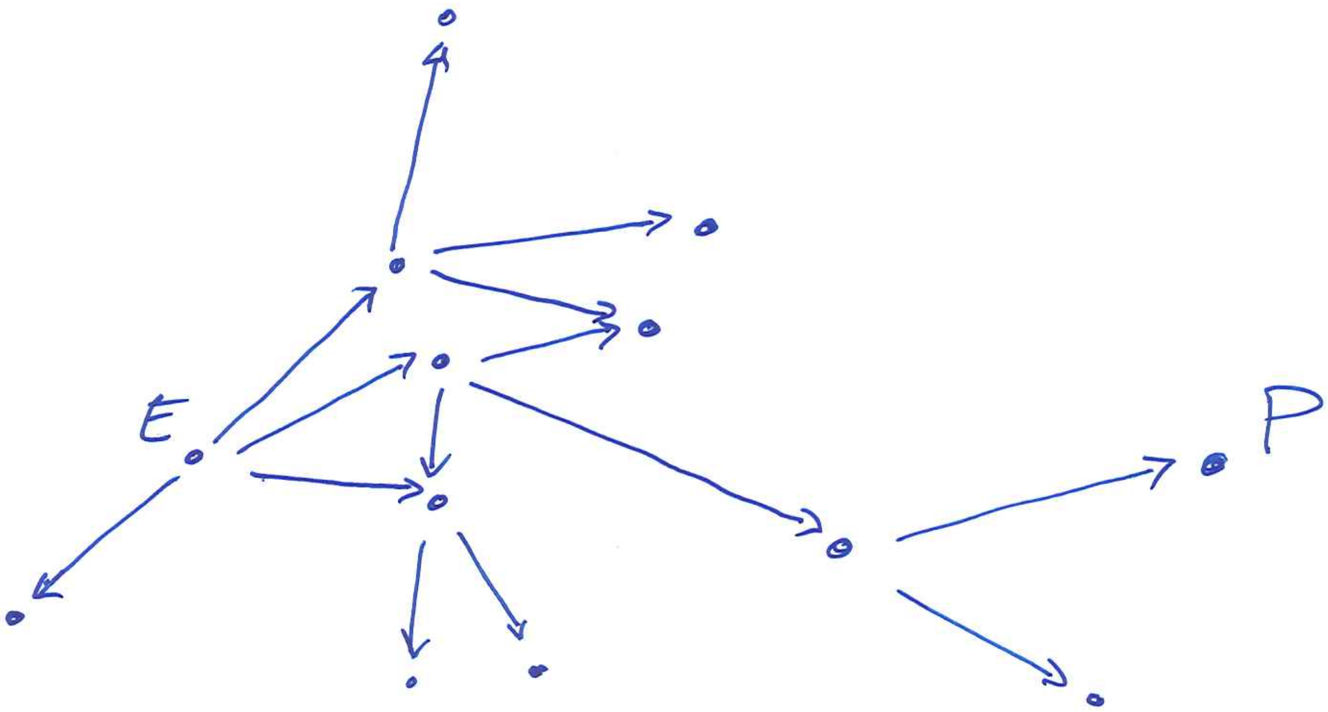
Choose another vertex (not P)
find all out edges

How to Solve It?



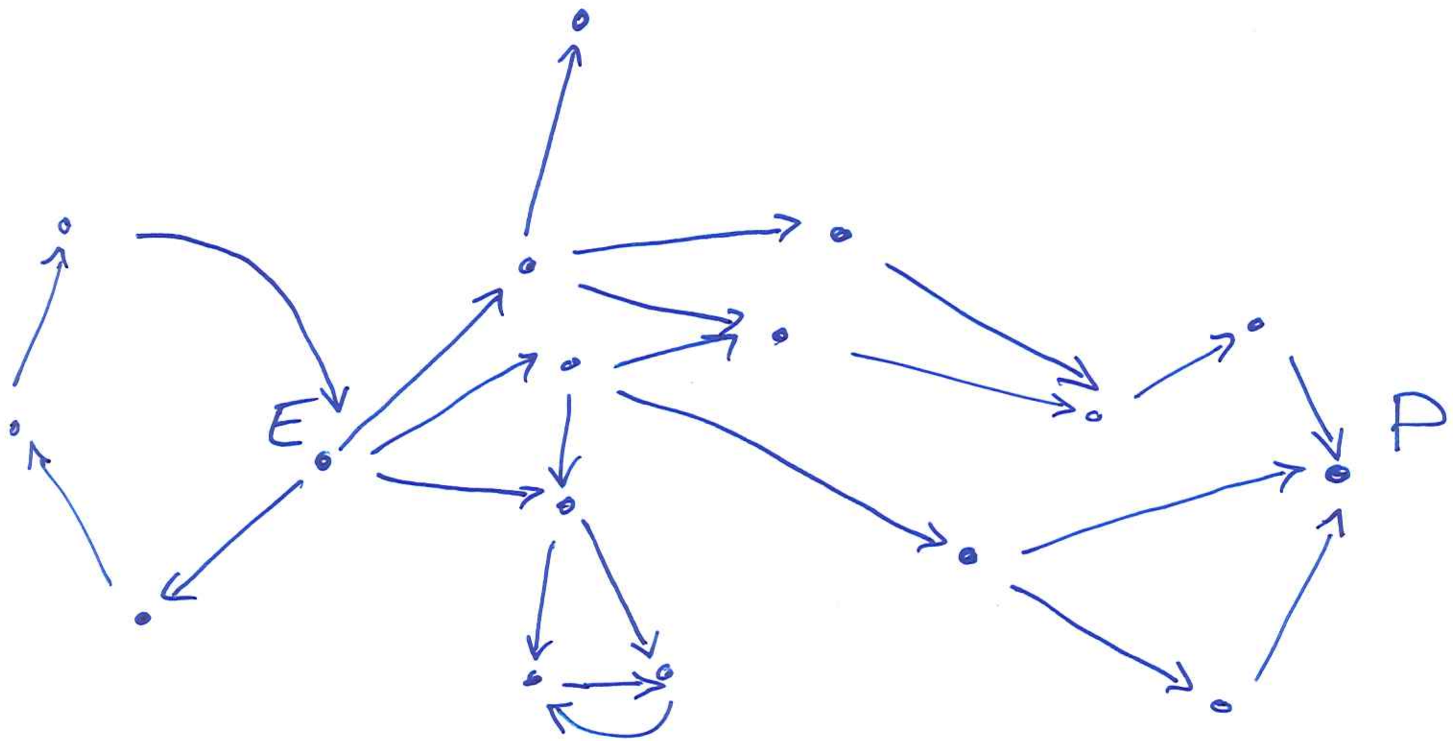
If we get a graph identical to one already found, only add the edge

How to Solve It?



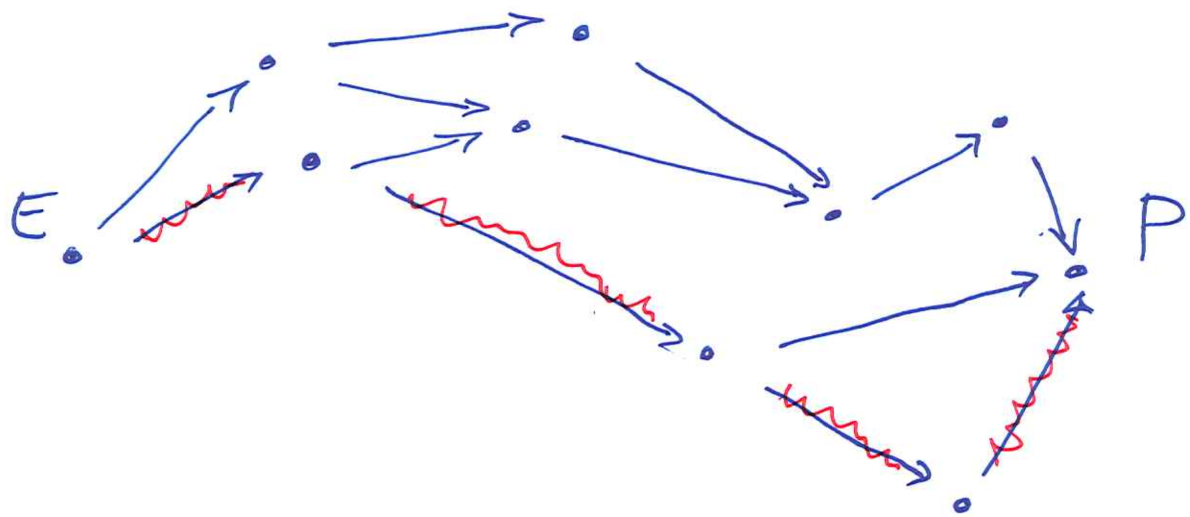
Repeat until we find P

How to Solve It?



... or until it converges.

How to Solve It?



the solution is then all (E, P) - paths in the decomposition graph

~~~ is an example
~~~

CODE

Algorithm 1 ElementaryDecomposition($r = (E, P)$)

Require: ER is a set of all elementary reactions
Require: r is a reaction, specified by educt E and product P
Ensure: D contains all elementary decompositions of r

$D \leftarrow$ an arc-labeled digraph $(V = \{E, P\}, A = \emptyset, l)$
$\quad$ $current \leftarrow$ a queue containing element E
while $current \neq \emptyset$ **do**
$\quad$ $G \leftarrow current.dequeue()$
$\quad$ **for** $e \in ER$ **do**
$\quad\quad$ $G'_{set} \leftarrow \{e_m(G) \mid m$ is valid mapping modulo problem restrictions$\}$ ← Most of the work!
$\quad\quad$ **for** $G' \in G'_{set}$ **do**
$\quad\quad\quad$ **if** $G' \notin V$ with equality modulo problem restrictions **then**
$\quad\quad\quad\quad$ $V \leftarrow V \cup \{G'\}$
$\quad\quad\quad\quad$ $current.enqueue(G')$
$\quad\quad\quad$ **end if**
$\quad\quad\quad$ $A \leftarrow A \cup \{(G, G')\}$ with $l((G, G')) = (e, m)$
$\quad\quad$ **end for**
$\quad$ **end for**
end while
return all simple (E, P)-paths in D with arc-label information

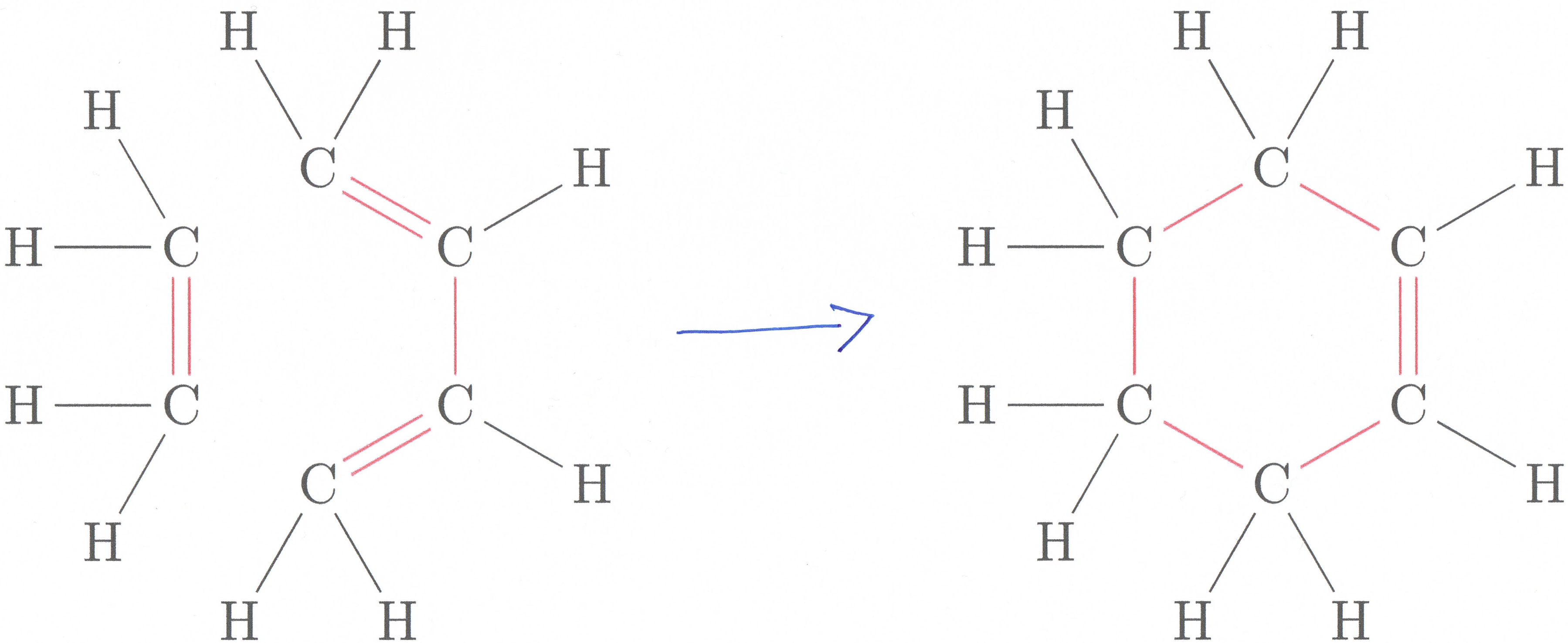
Search Strategy

Problem Version

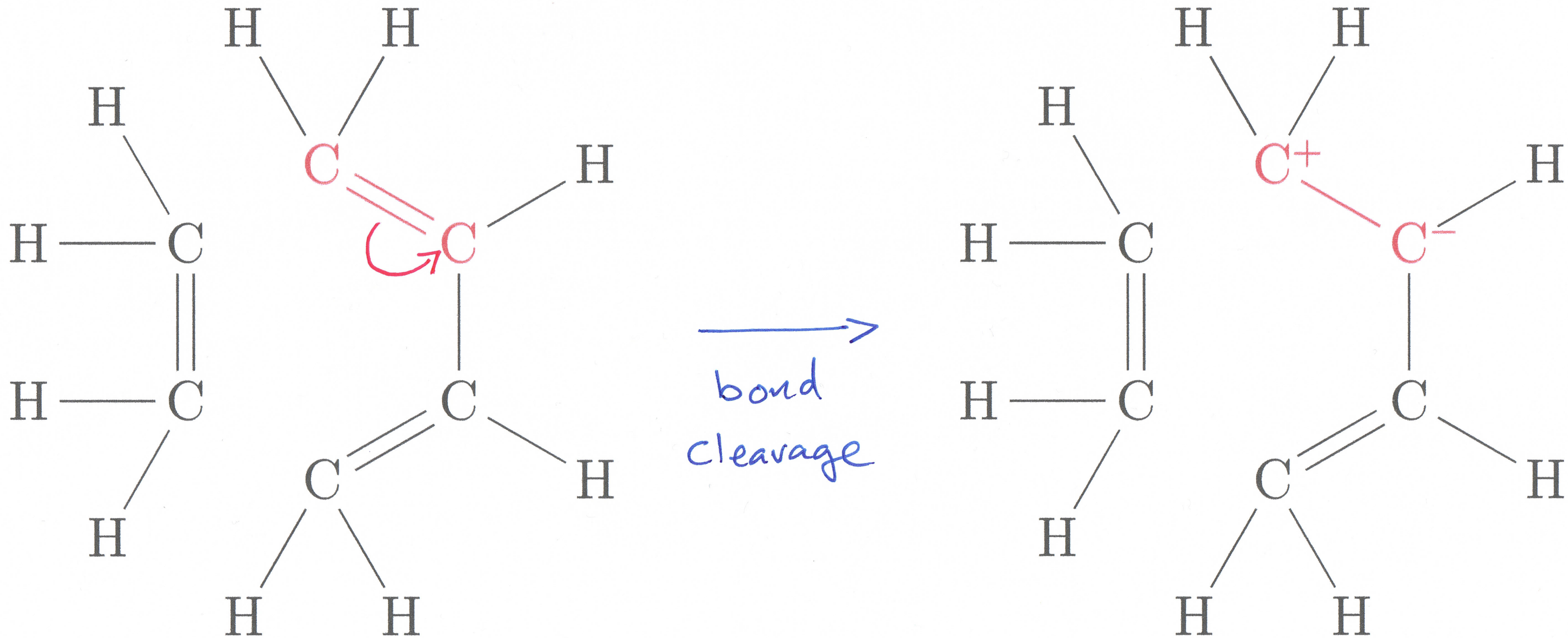
$\cong$ or $=$

EXAMPLE

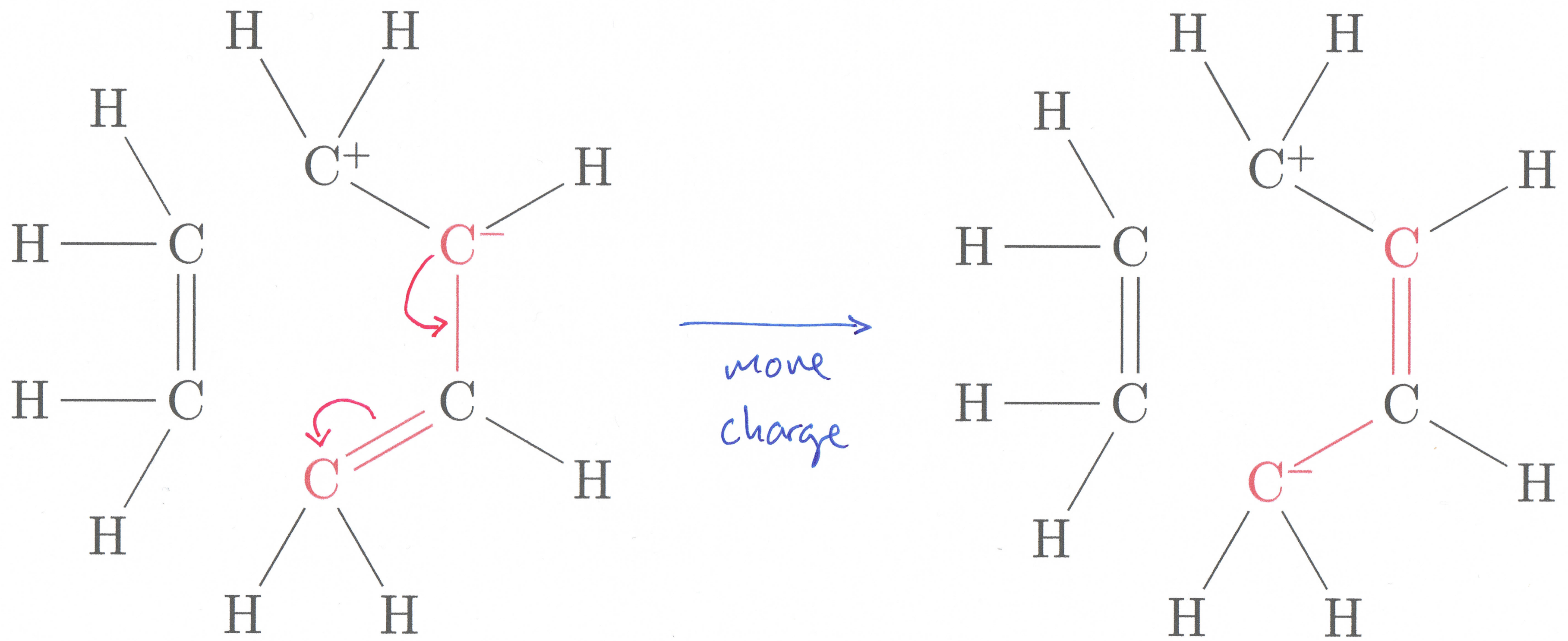
Diels Alder



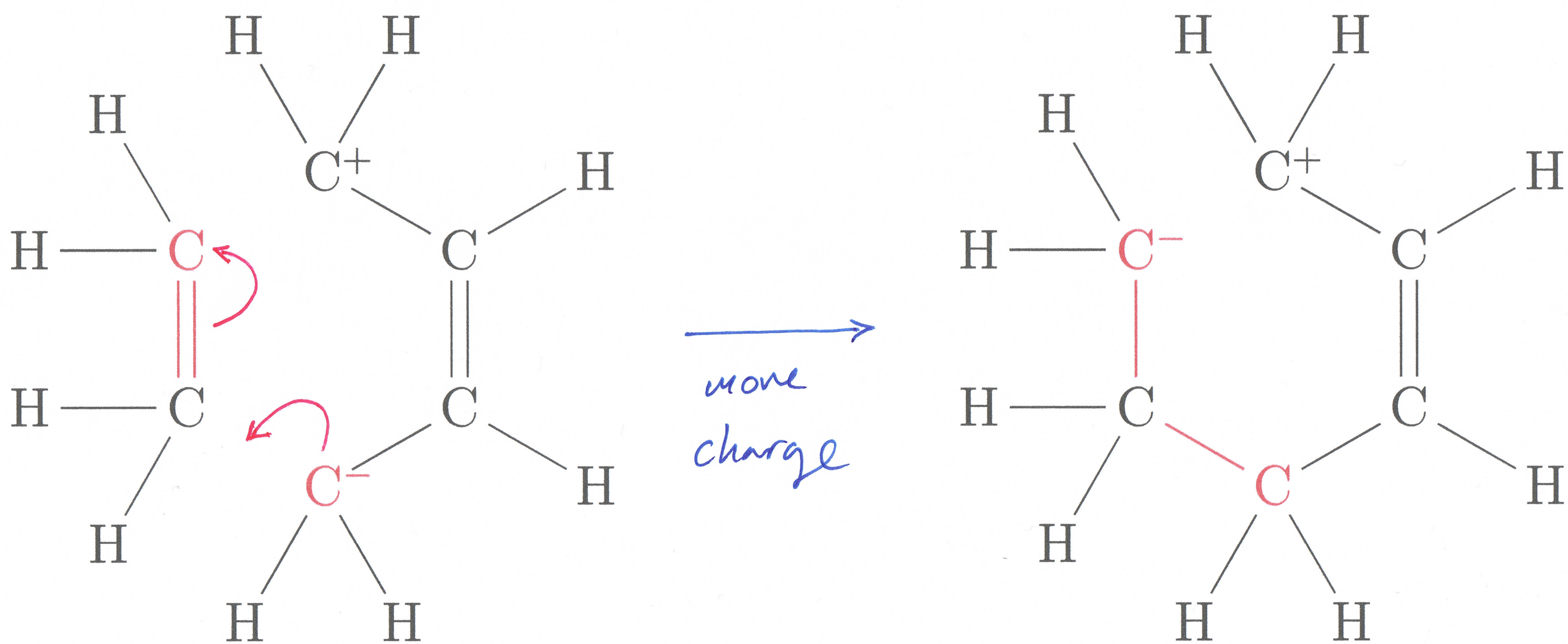
EXAMPLE



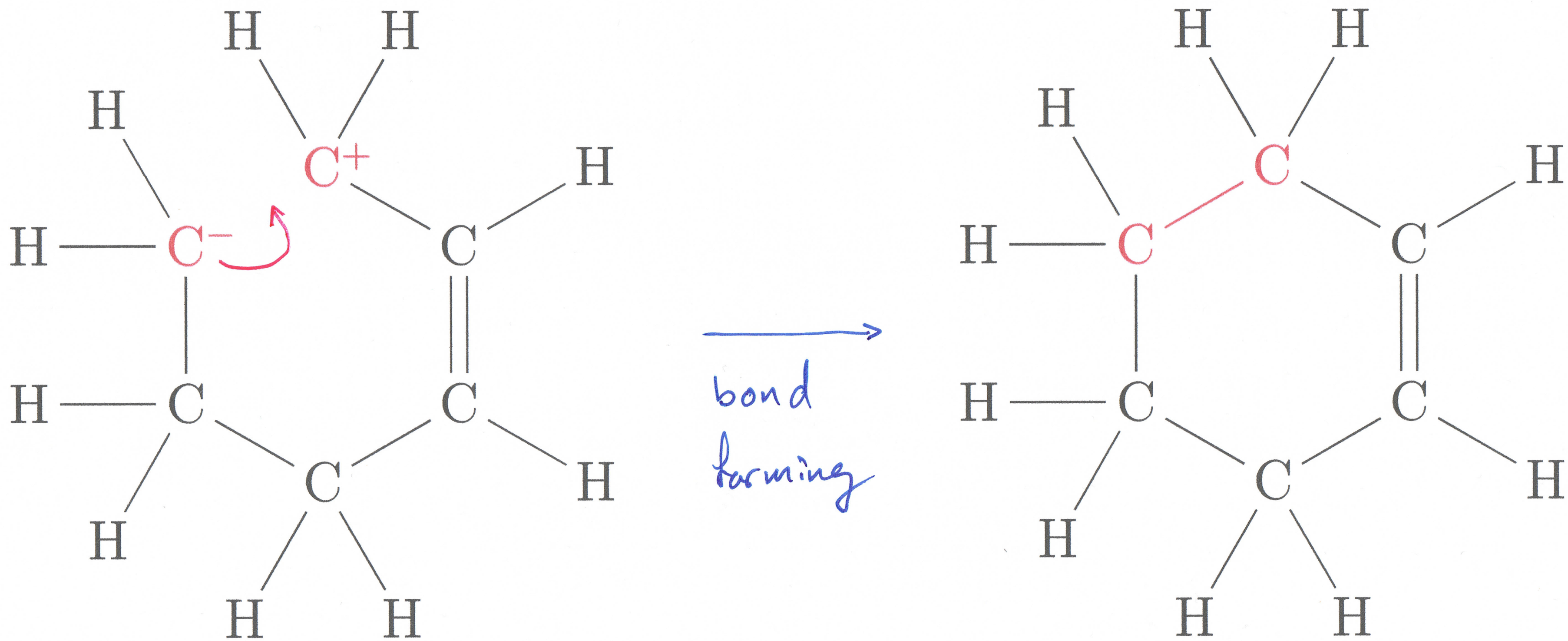
EXAMPLE



EXAMPLE

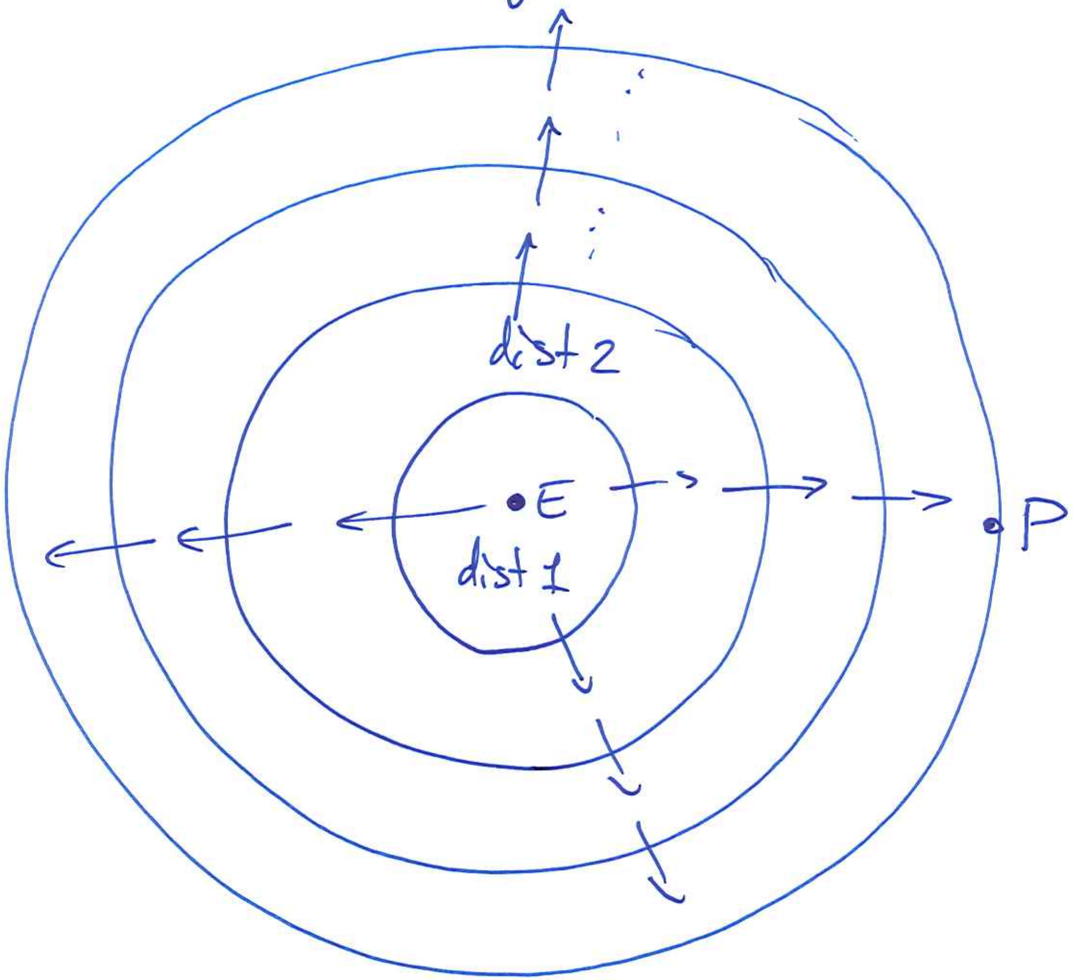


EXAMPLE



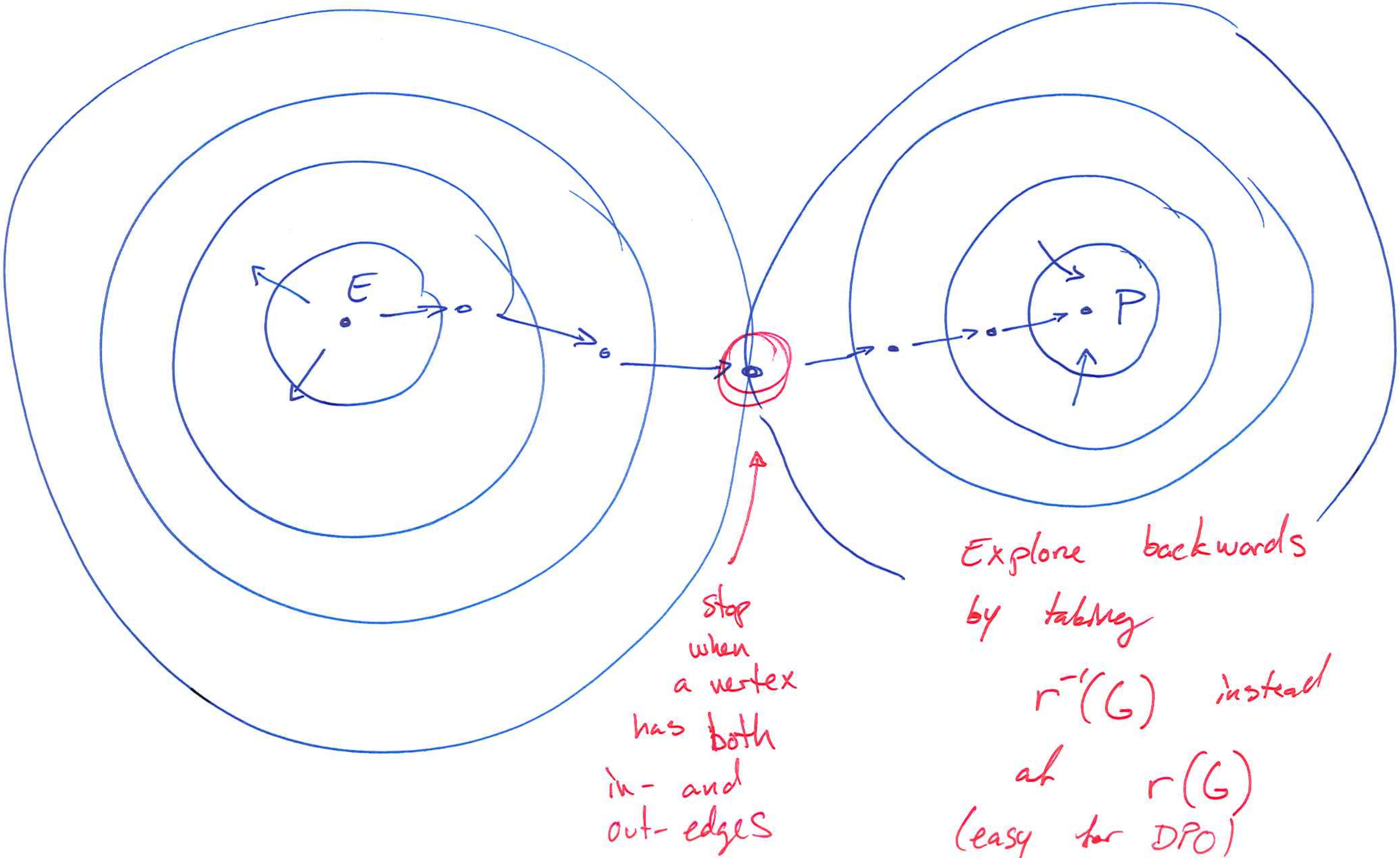
Speed

Time used depends heavily on length of the decomposition (for ONE solution)



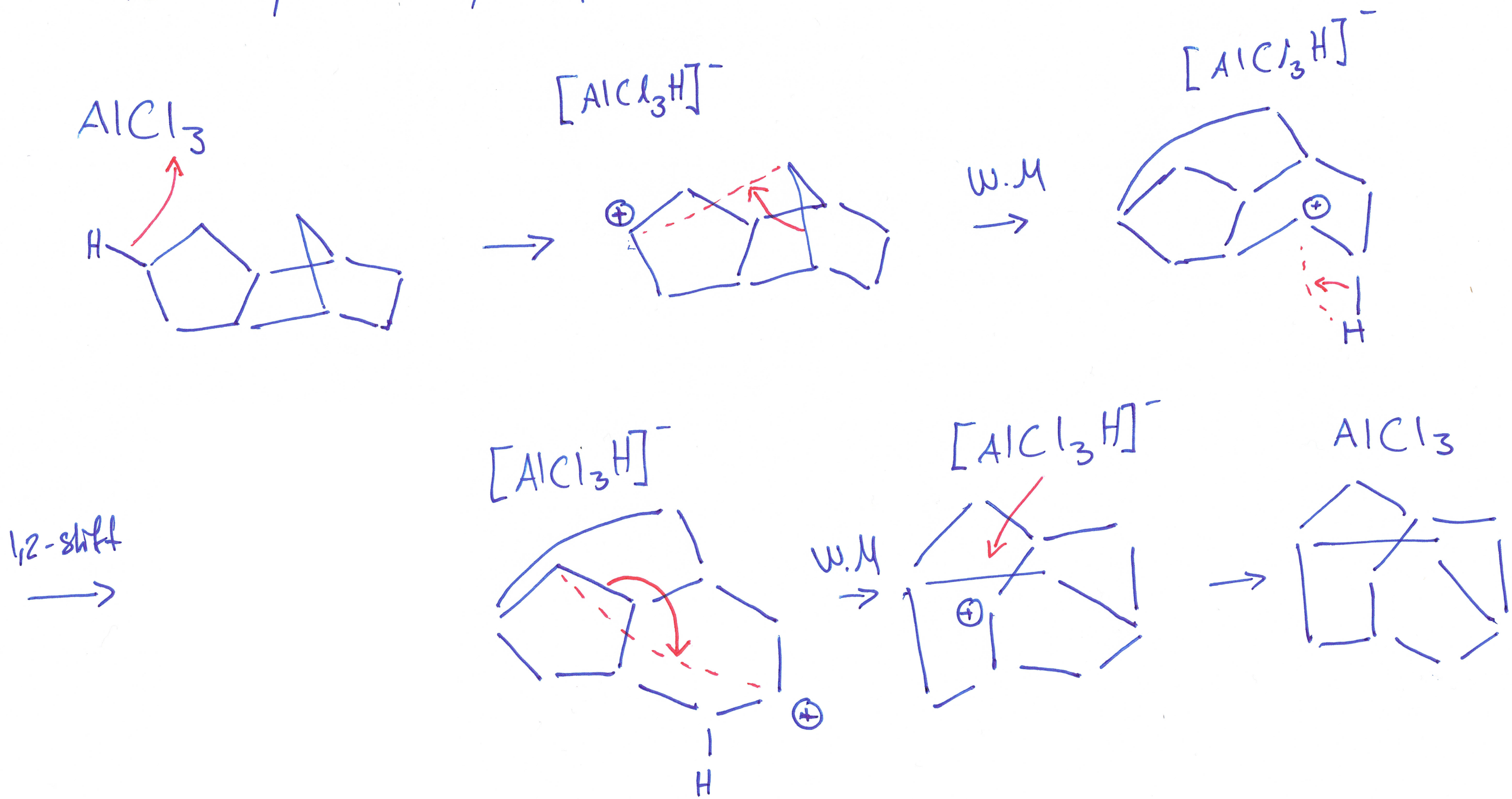
Speed

- Use the fact that we know P!



Example

tetrahydrodicyclopentadiene to adamantane



Other Potential Speedups

- More restricted versions of the problem are easier to solve, but will ~~~~ miss solutions.
- Possibility to use φ to guide search.
- By ordering ~~~~ rules applications some symmetries can be broken.

Free Lunch ?

- Versions not using an atom atom mapping will yield an atom atom mapping as part of solution.

- For most descriptions of context of rules in ER we can infer relevant context of E → P by union of context of decomposition

$\Longrightarrow$ generalization of E → P

The Big Challenge

Chemistry

vs.

Wrong Chemistry (now)

FIN