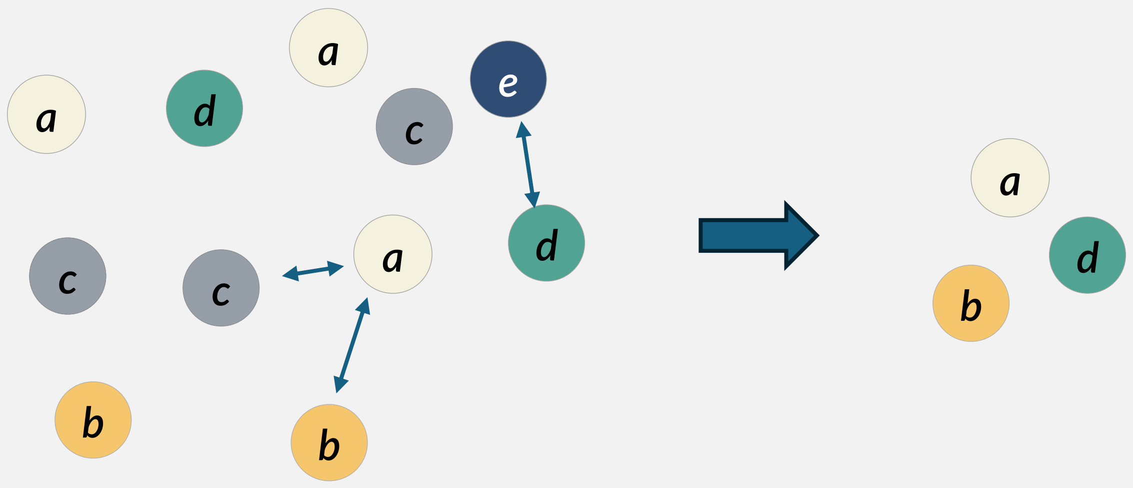
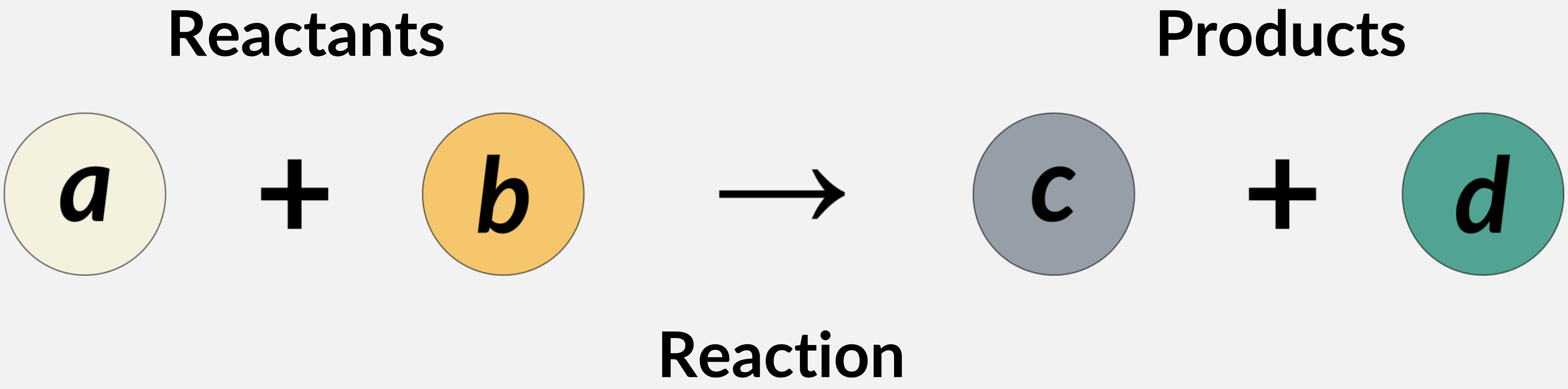
Brief Announcement:
Reachability in Deletion-only Chemical Reaction Networks

The 4th Symposium on Algorithmic Foundations of Dynamic Networks (SAND)

June 9-11, 2025 | Liverpool, UK

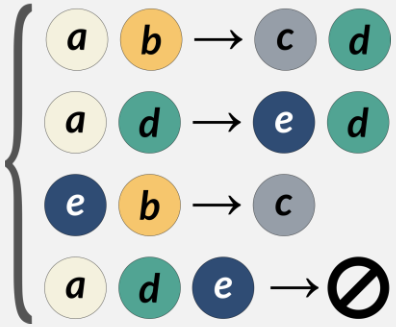
Bin Fu, Timothy Gomez , Ryan Knobel, Austin Luchsinger, Aiden Massie,
Marco Rodriguez, Adrian Salinas, Robert Schweller, Tim Wylie

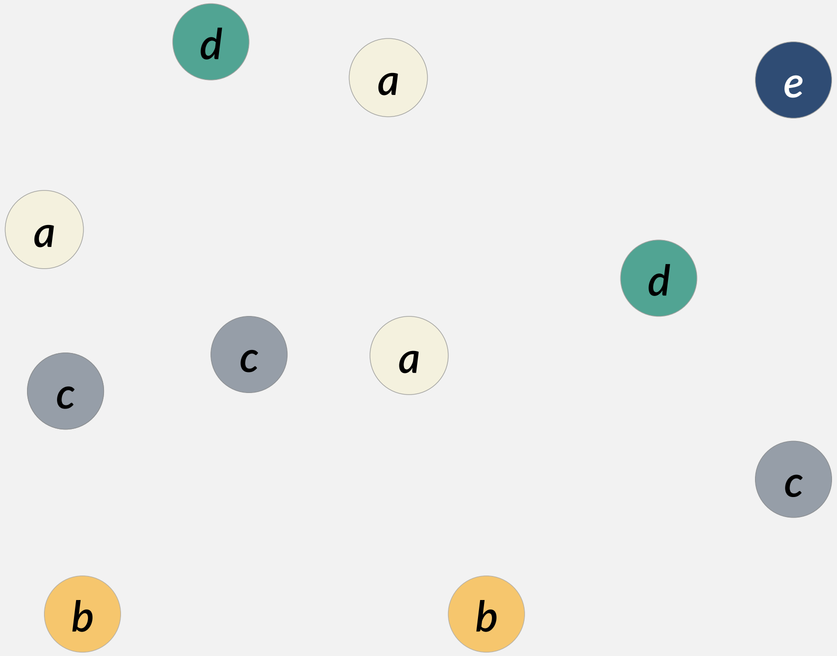




Chemical Reaction Networks

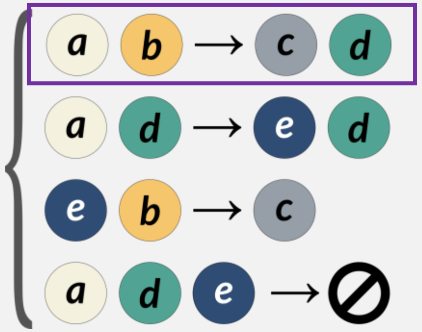
Reaction Rules

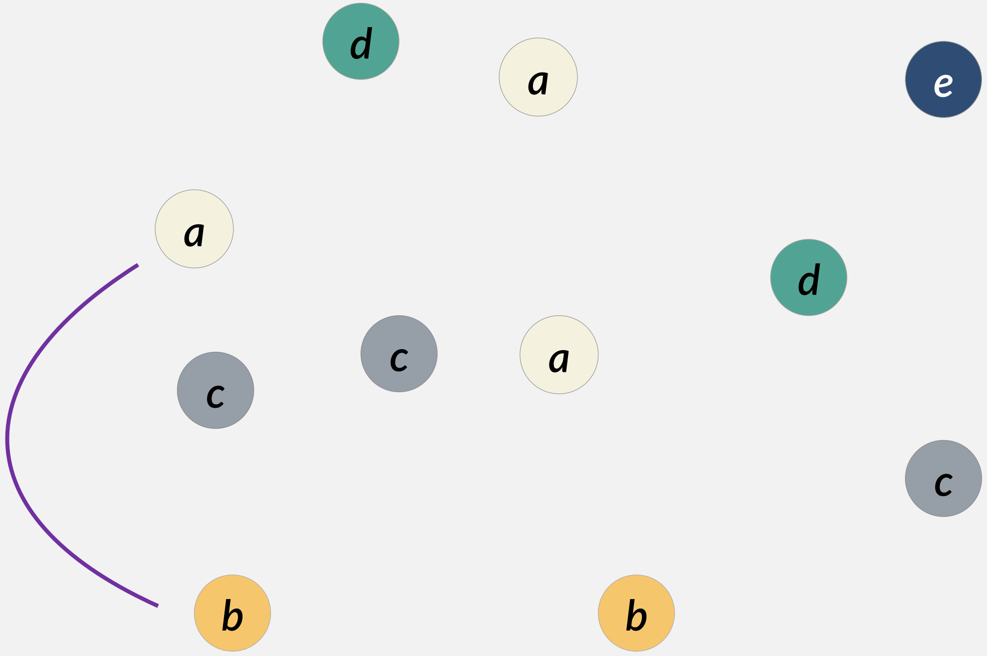




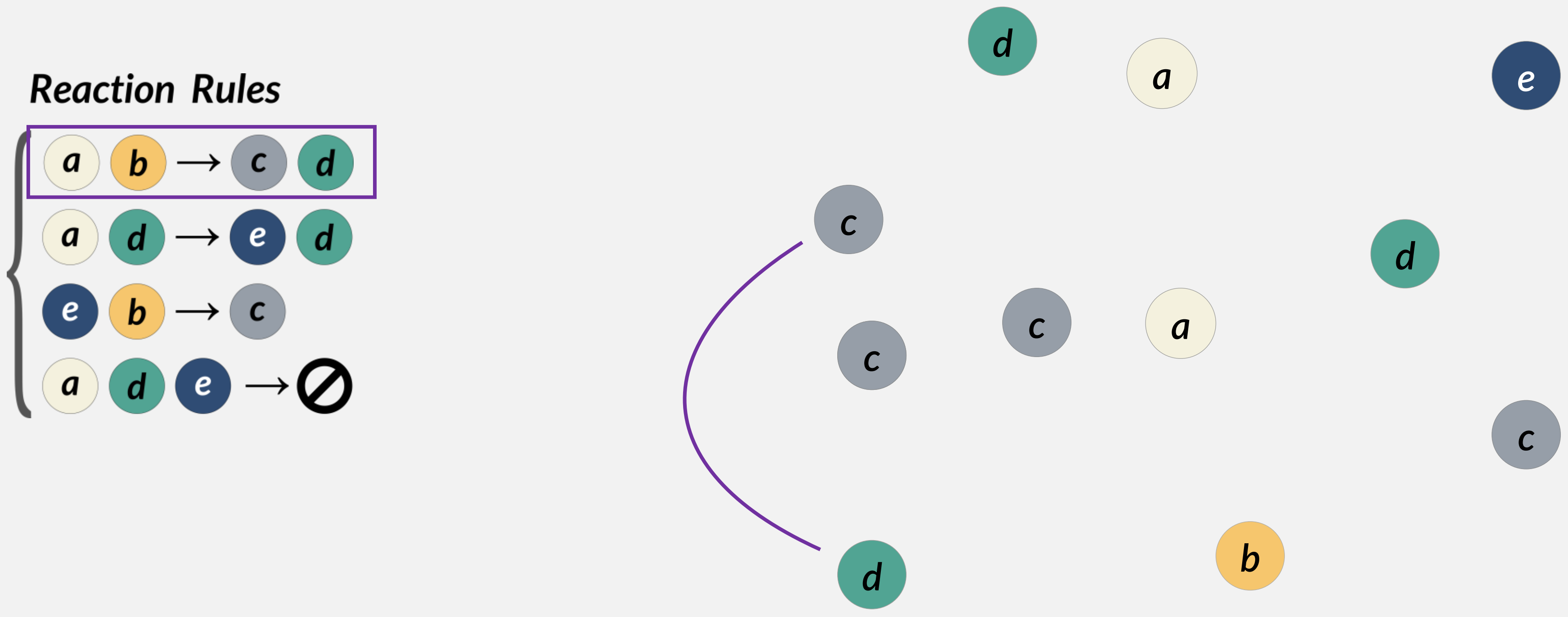
CRNs

Reaction Rules



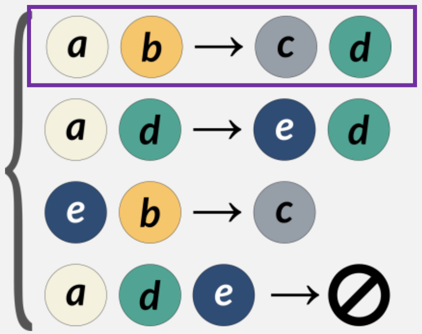


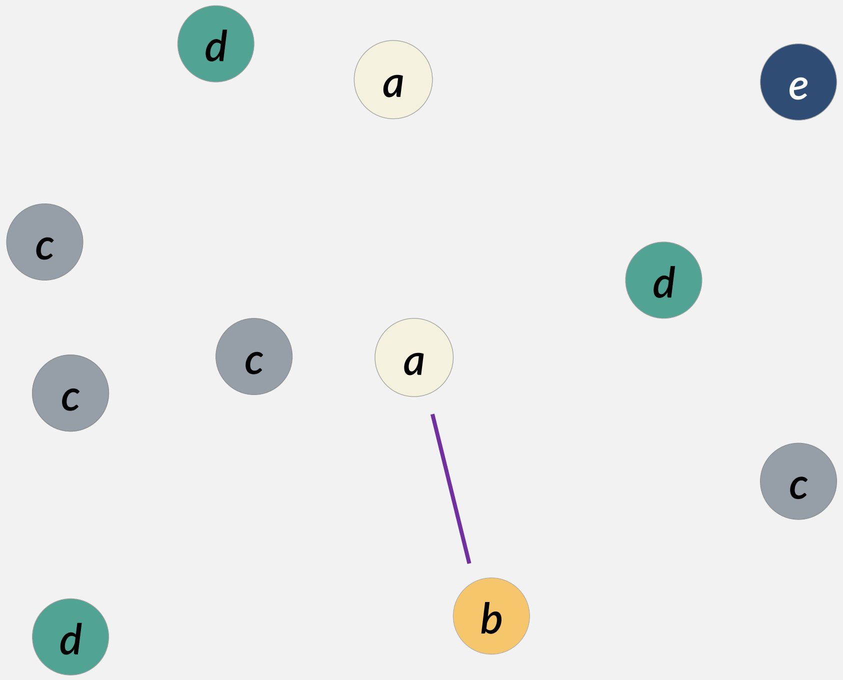
CRNs



CRNs

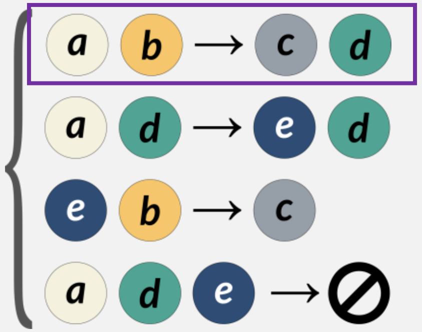
Reaction Rules

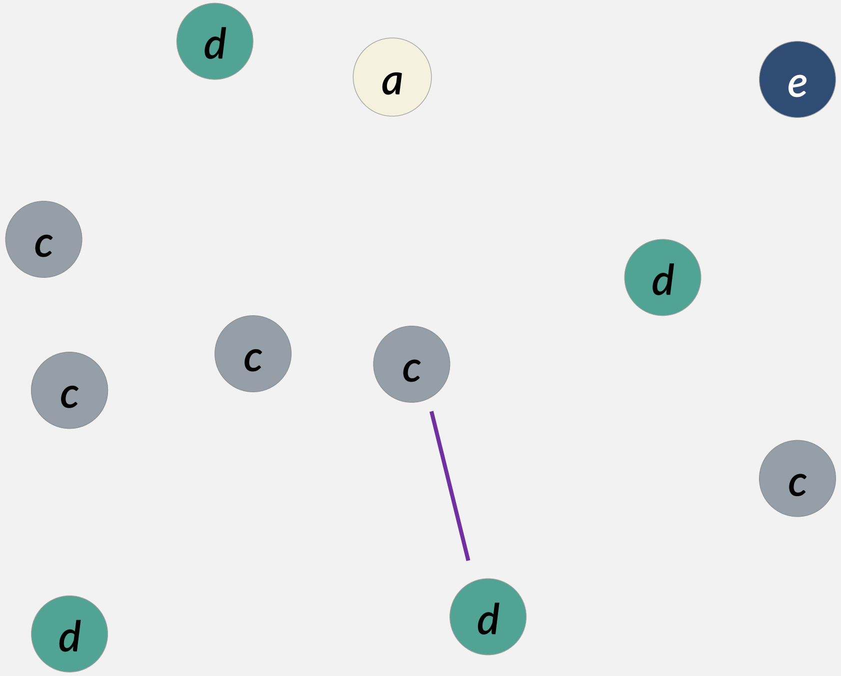




CRNs

Reaction Rules

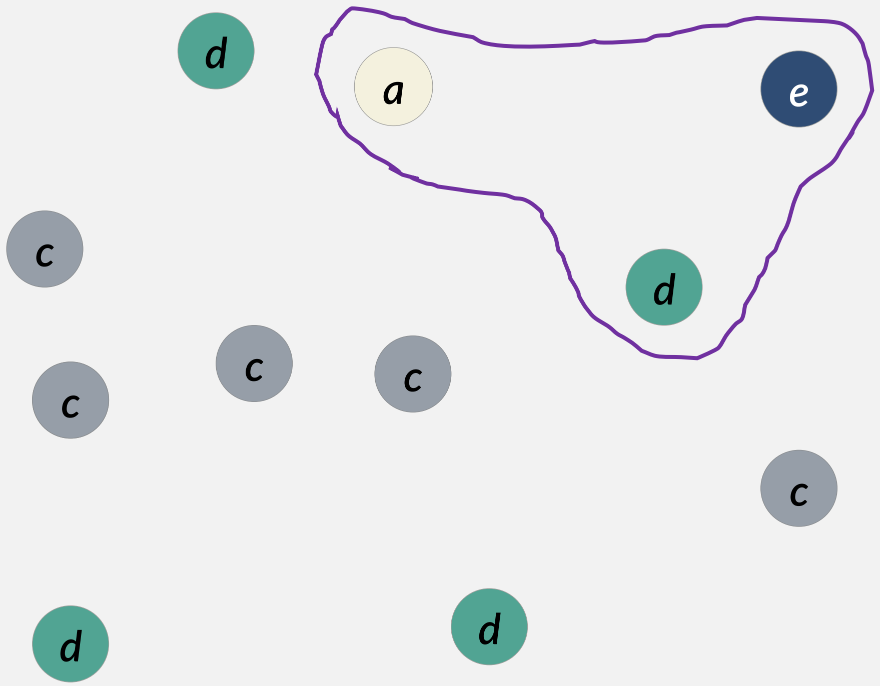




CRNs

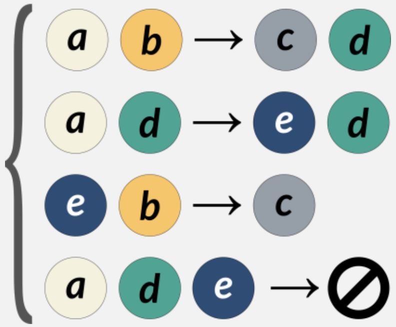
Reaction Rules

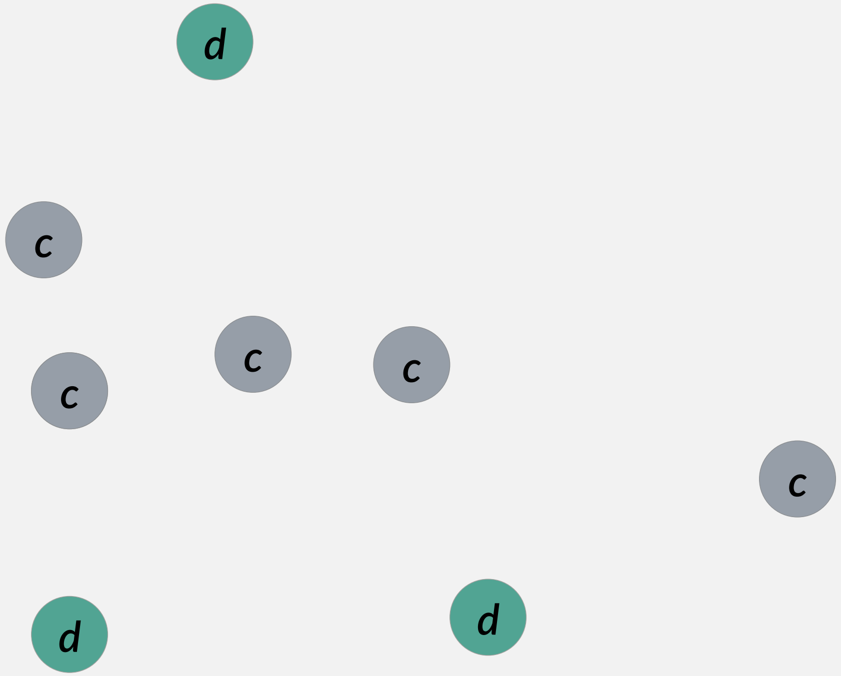
$$a + b \rightarrow c + d$$
$$a + d \rightarrow e + d$$
$$e + b \rightarrow c$$
$$a + d + e \rightarrow \emptyset$$



CRNs

Reaction Rules

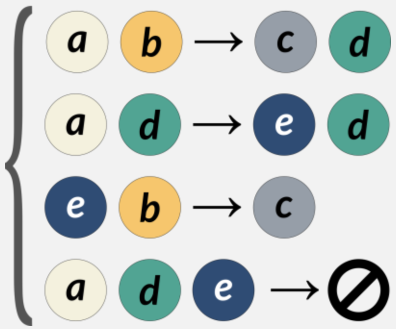




CRNs

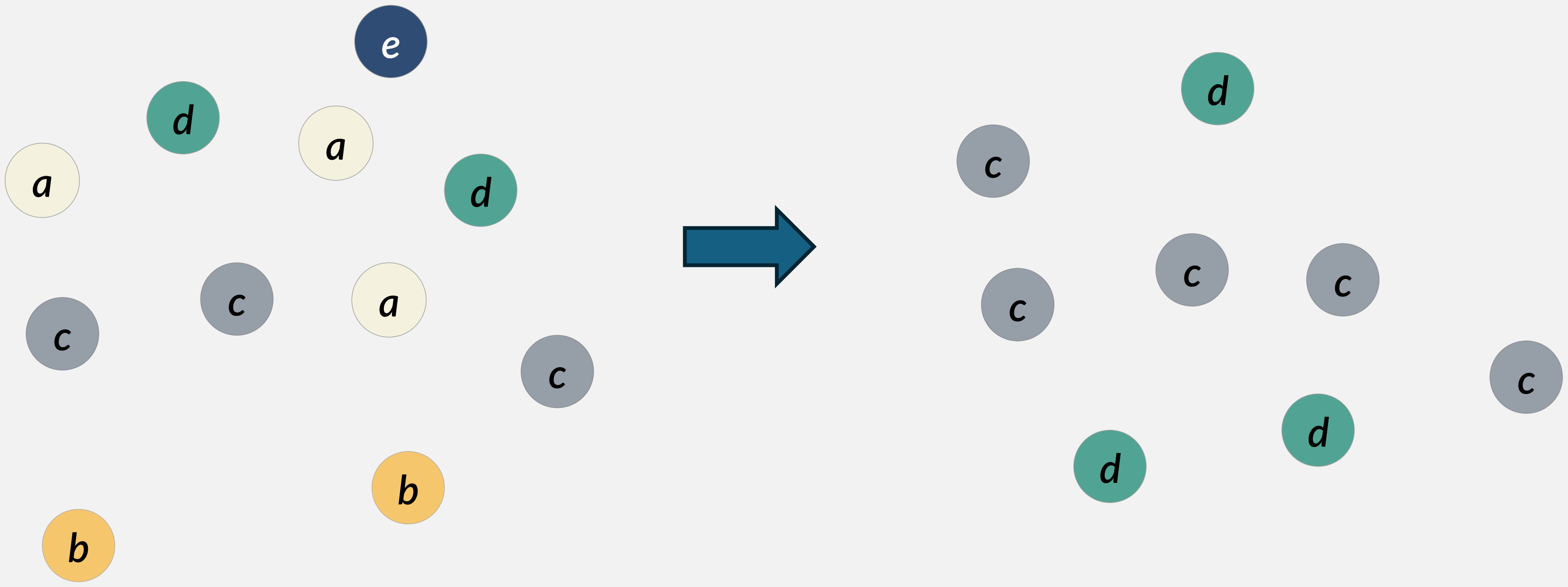
CRN:

Reaction Rules



Configuration I:

Configuration D:

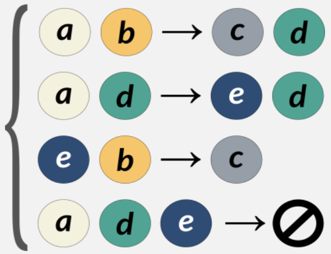


D is **Reachable** from I

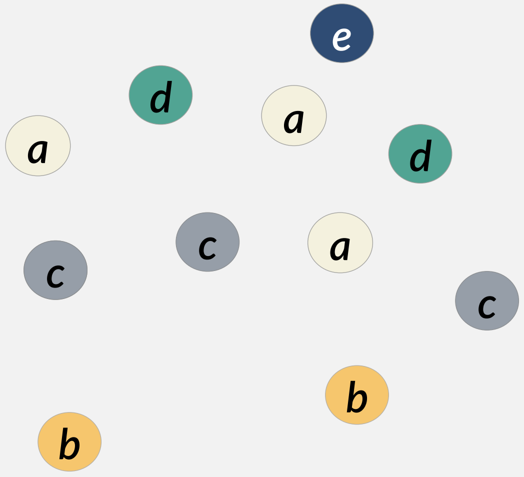
CRNs

CRN:

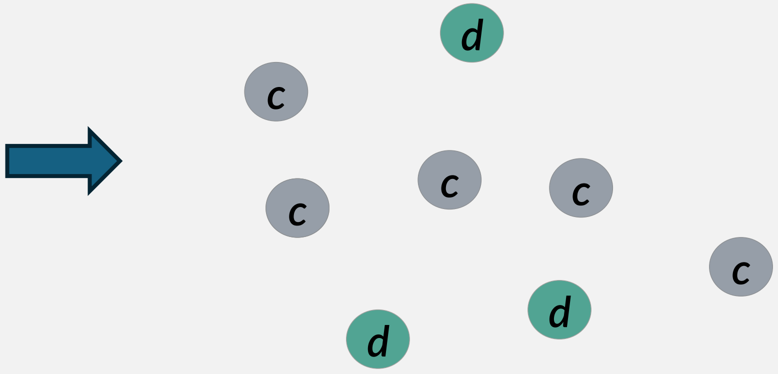
Reaction Rules



Configuration I:



Configuration D:



D is **Reachable** from I

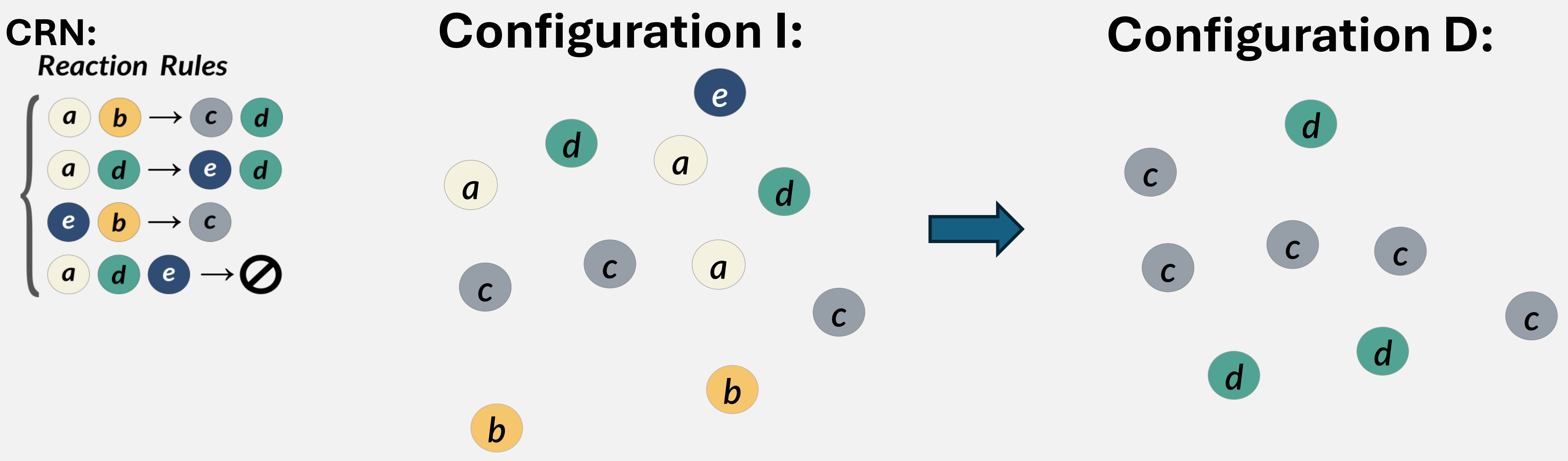
Reachability Problem:

Input:

1) Reaction Rules (a CRN)
2) An initial configuration I
3) a destination configuration D

Output: Is D reachable from I with these reactions?

CRNs



D is **Reachable** from I

Reachability Problem:

Input:

1) Reaction Rules (a CRN)
2) An initial configuration I
3) a destination configuration D

Output: Is D reachable from I with these reactions?

Reachability is Ackerman-Complete

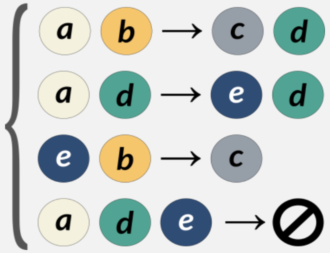
[Wojciech Czerwiński and Łukasz Orlikowski, FOCS'21]

[Jérôme Leroux, FOCS'21]

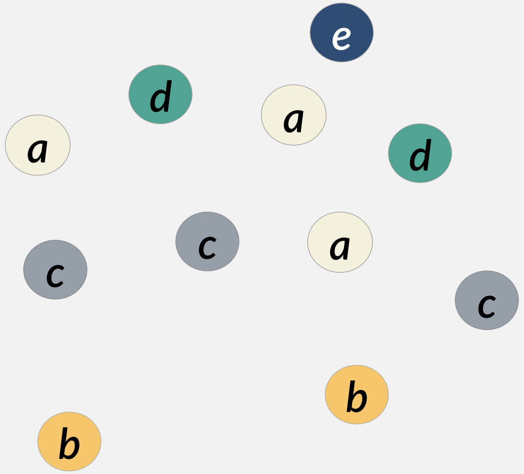
CRNs

CRN:

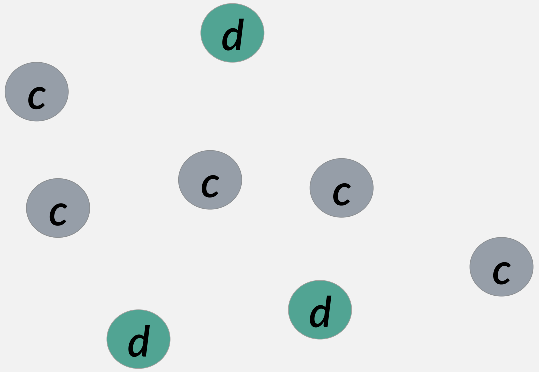
Reaction Rules



Configuration I:



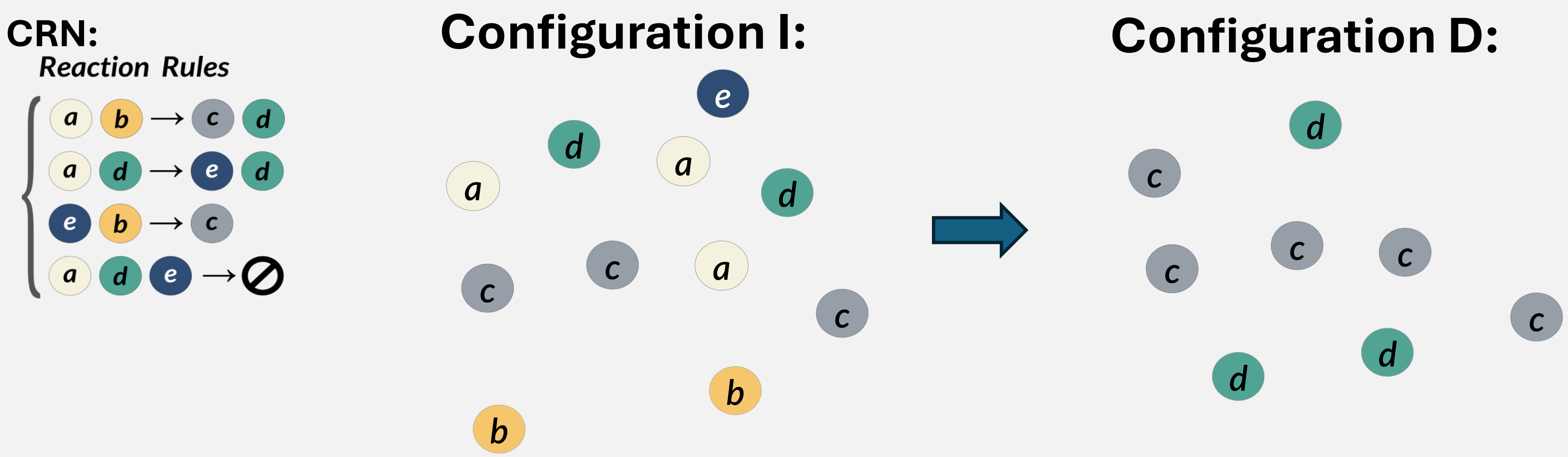
Configuration D:



D is **Reachable** from I

- General CRNs reactions can **ADD** and **DELETE**

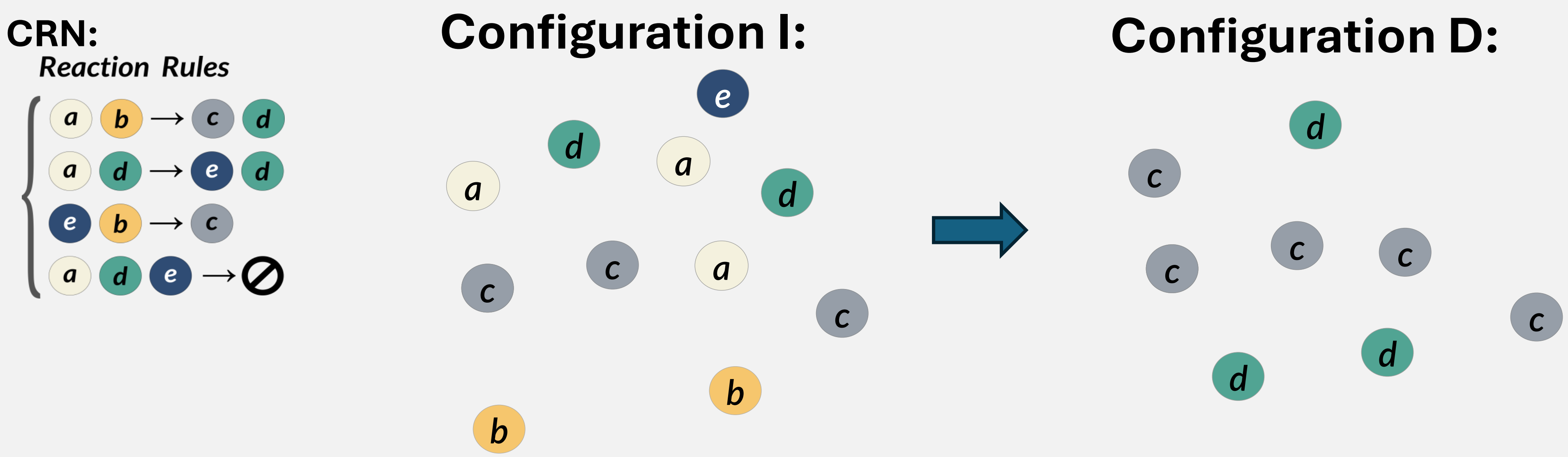
  => Reachability is **Ackermann-complete** [FOCS 2021]



D is **Reachable** from I

- General CRNs reactions can **ADD** and **DELETE**
  => Reachability is **Ackermann-complete** [FOCS 2021]

- What if a system could only ADD **OR** DELETE (but not both)?
  ⇒ Reachability is in NP



D is **Reachable** from I

- General CRNs reactions can **ADD** and **DELETE**
  => Reachability is **Ackermann-complete** [FOCS 2021]

- What if a system could only ADD **OR** DELETE (but not both)?
  ⇒ Reachability is in NP

Our Focus:

We characterize Reachability in **Deletion-Only** systems
Based on rule size.

Deletion-Only Reactions (Void Rules)

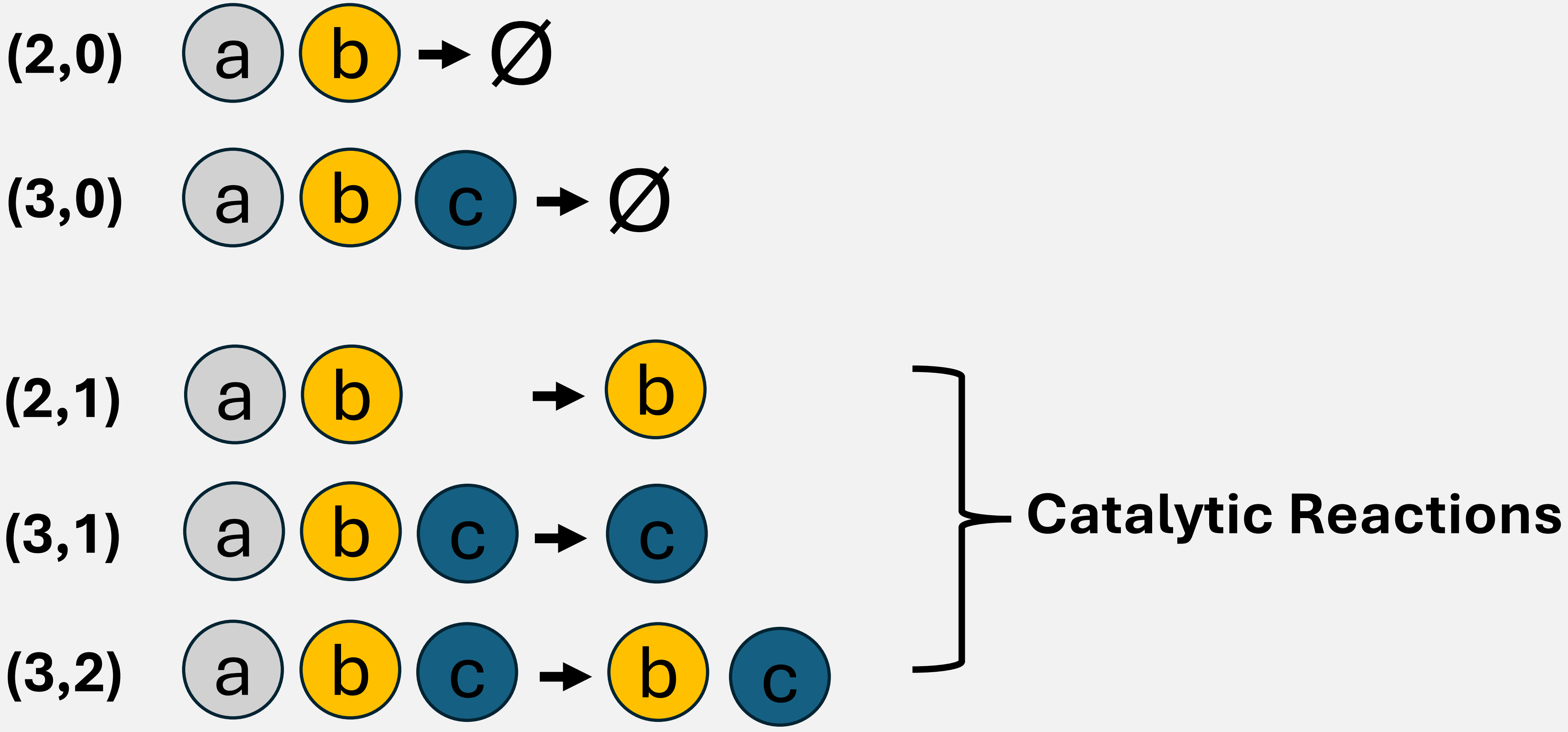
Rule Size:

(2,0) a b → Ø

(3,0) a b c → Ø

Deletion-Only Reactions (Void Rules)

Rule Size:



Deletion-Only CRNs

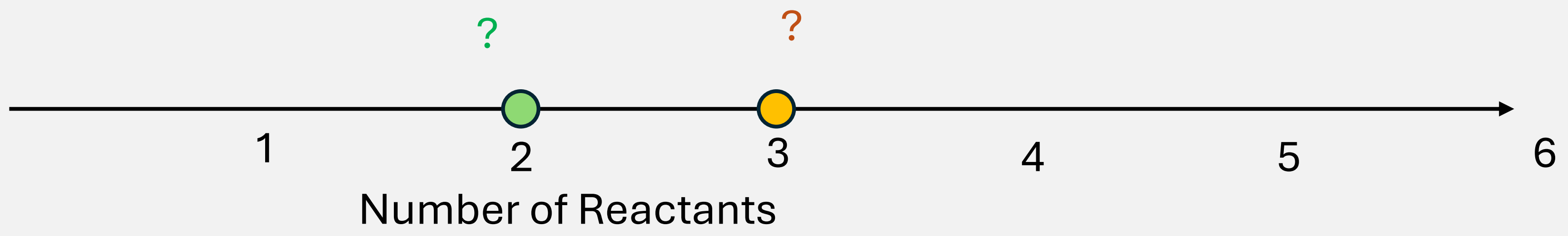
(2,0) rules

a + b → Ø

Versus:

(3,0) rules

a + b + c → Ø

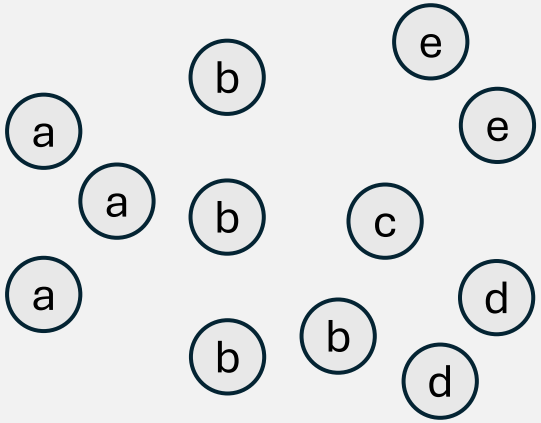


(2,0) Algorithm

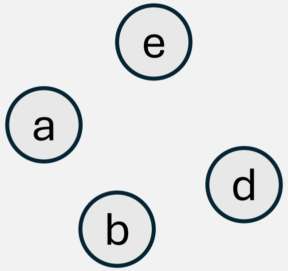
Reactions:

a b => Ø
b c => Ø
e d => Ø

Initial Configuration:

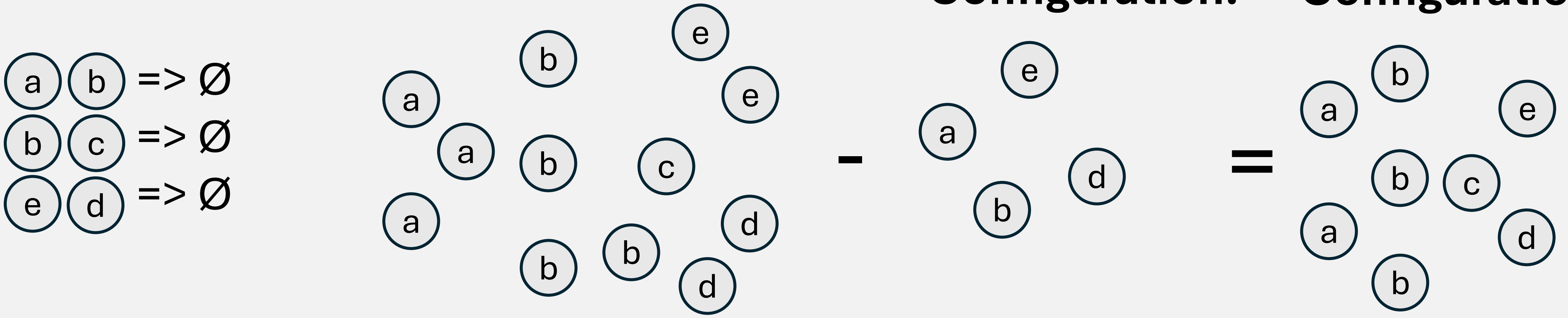


Target Configuration:



(2,0) Algorithm

Reactions: **Initial Configuration:** **Target Configuration:** **Difference Configuration:**

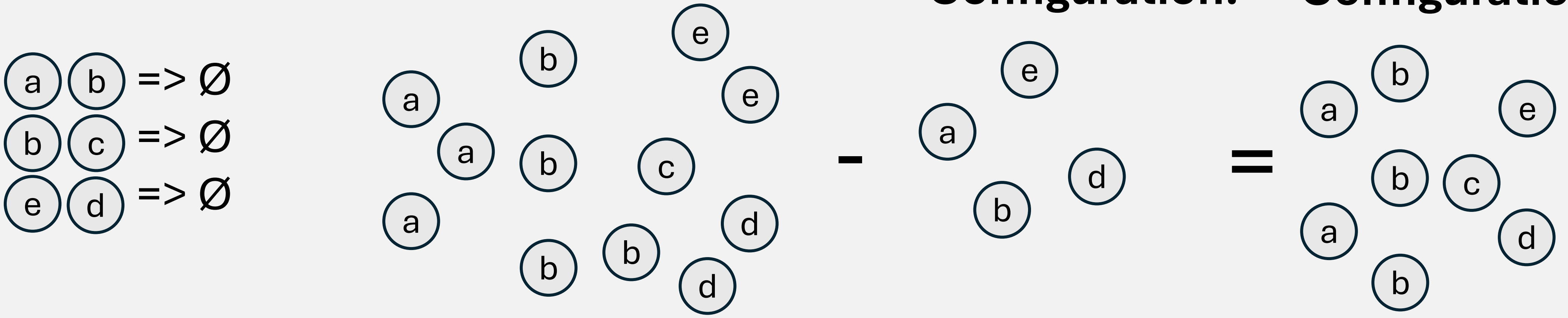


Algorithm:

- **Compute the “Difference” configuration**

(2,0) Algorithm

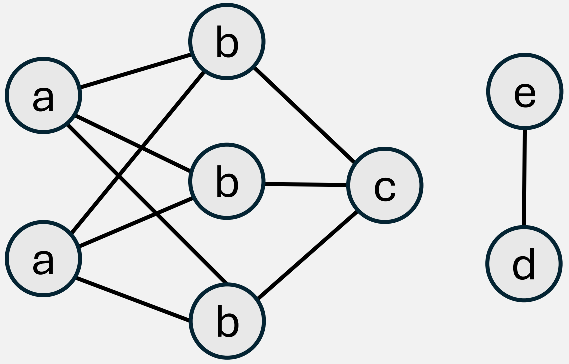
Reactions: **Initial Configuration:** **Target Configuration:** **Difference Configuration:**



Algorithm:

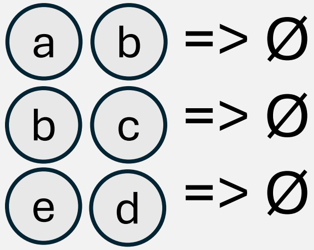
- **Compute the “Difference” configuration**
- **Create the Reaction Graph**

Reaction Graph:

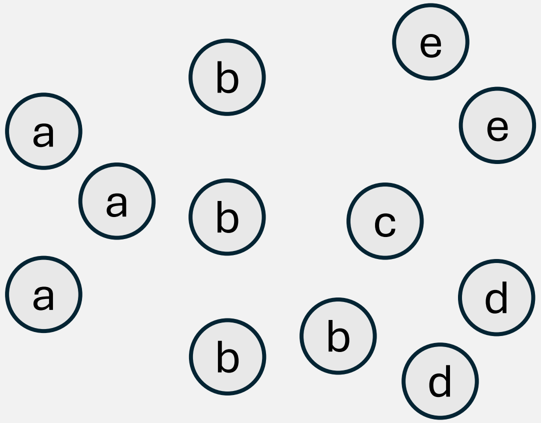


(2,0) Algorithm

Reactions:

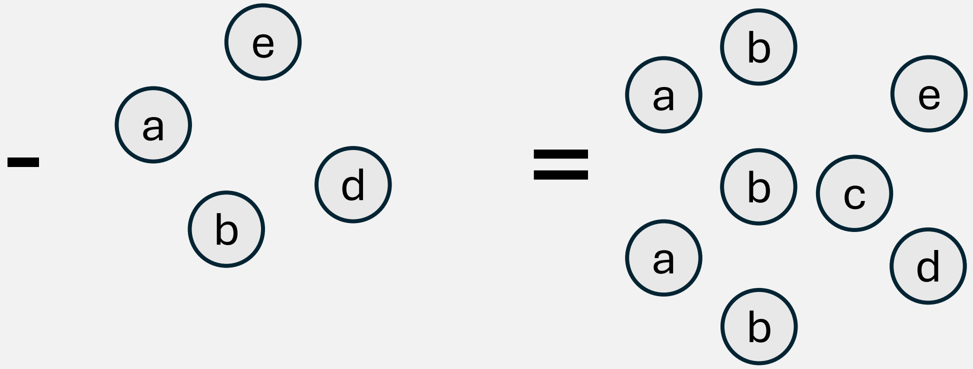


Initial Configuration:



Target Configuration:

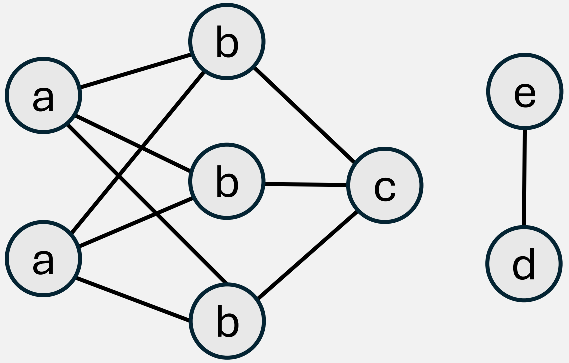
Difference Configuration:



Algorithm:

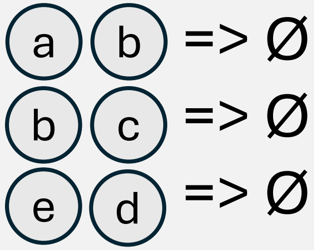
- **Compute the “Difference” configuration**
- **Create the Reaction Graph**
- **Run a Perfect Matching Algorithm**

Reaction Graph:

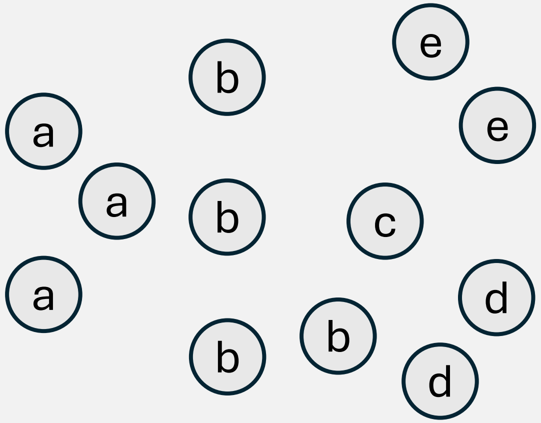


(2,0) Algorithm

Reactions:

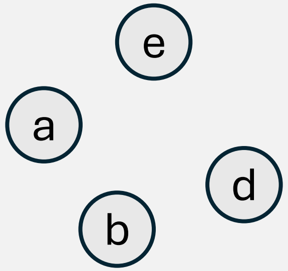


Initial Configuration:



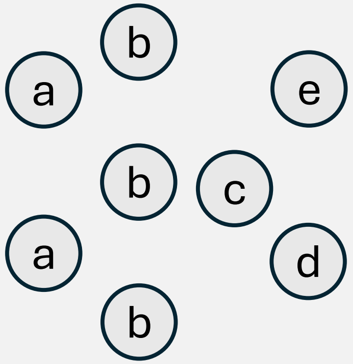
\-

Target Configuration:



=

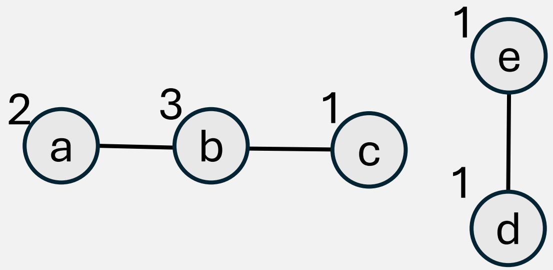
Difference Configuration:



Algorithm:

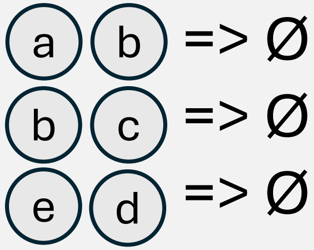
- **Compute the "Difference" configuration**
- **Create the Reaction Graph**
- **Run a Perfect Matching Algorithm**

Reaction Graph:

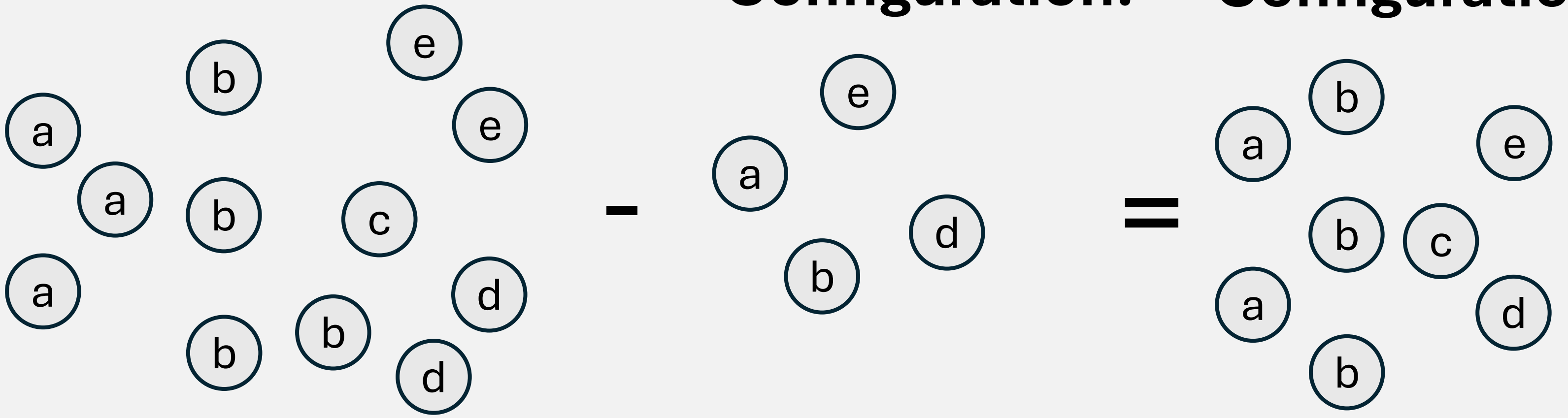


(2,0) Algorithm

Reactions:



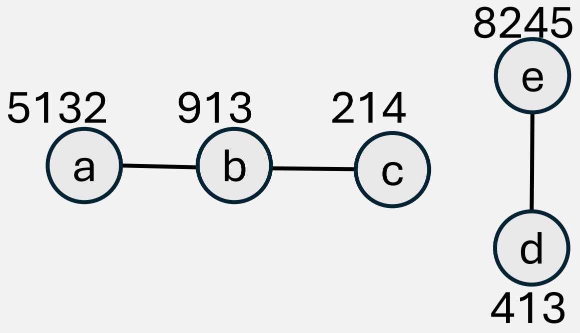
Initial Configuration: **Target Configuration:** **Difference Configuration:**



Algorithm:

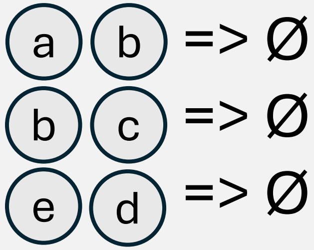
- **Compute the "Difference" configuration**
- **Create the Reaction Graph**
- **Run a Perfect Matching Algorithm**

Reaction Graph:

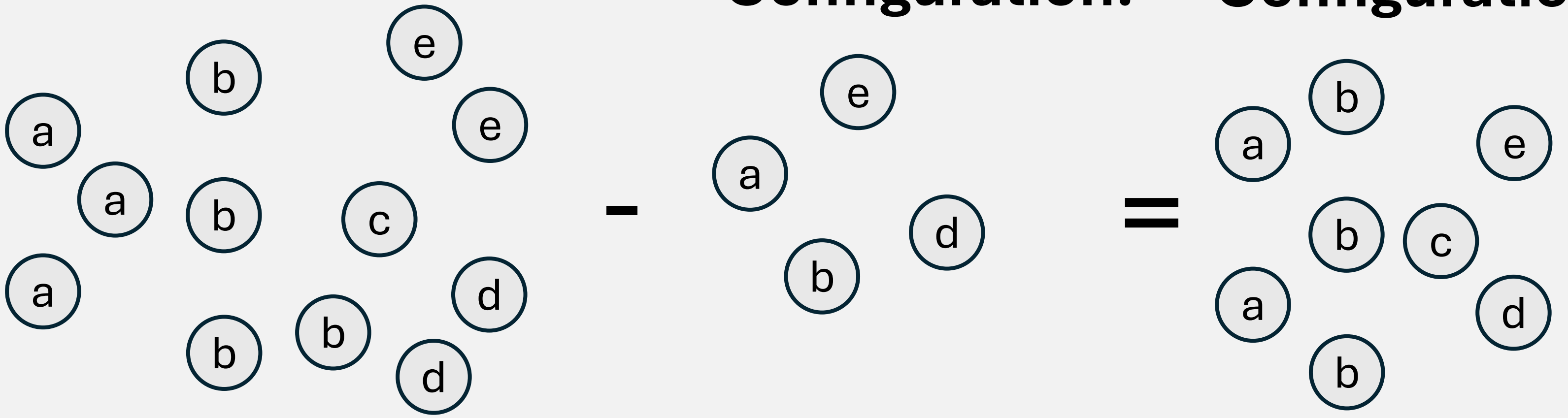


(2,0) Algorithm

Reactions:



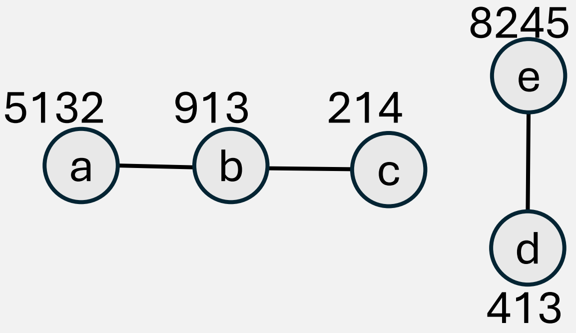
Initial Configuration: **Target Configuration:** **Difference Configuration:**



Algorithm:

- **Compute the "Difference" configuration**
- **Create the Reaction Graph**
- **Run a Perfect ~~Matching Algorithm~~ b-Matching Algorithm** [Cunningham, Marsh, 1979]

Reaction Graph:



Deletion-Only CRNs

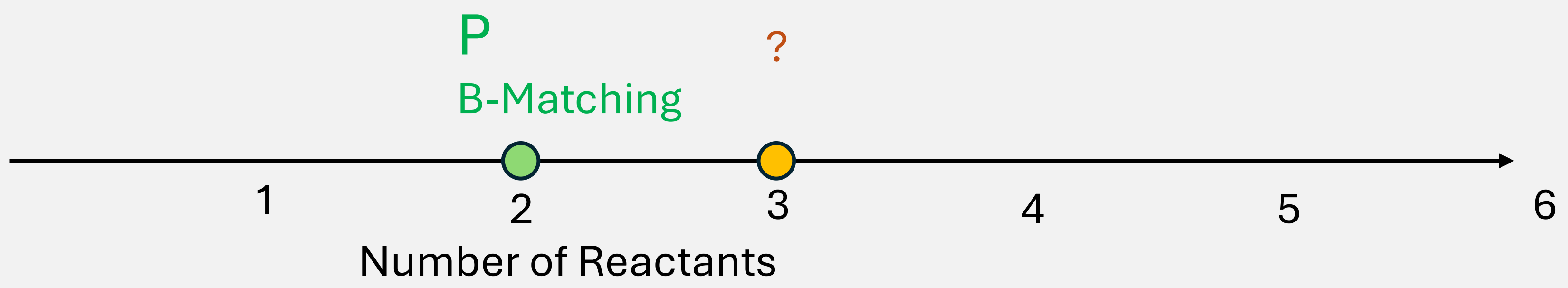
(2,0) rules

a + b → Ø

Versus:

(3,0) rules

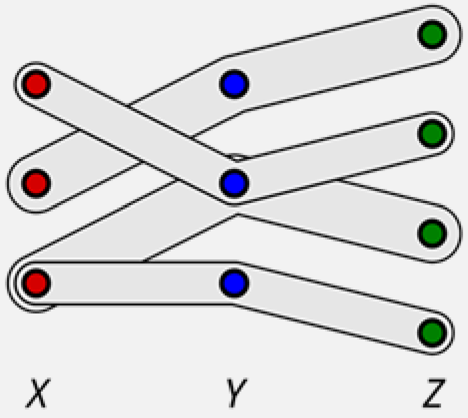
a + b + c → Ø



(3,0) Reachability: NP completeness

Reduction from 3D Matching

3D Matching Problem:
Does there exist a subset of non-overlapping triplets that cover the tripartite graph?



(3,0) Reachability Instance:

Initial Configuration:

x_1 y_1 z_1

x_2 y_2 z_2

x_3 y_3 z_3

z_4

Target Configuration:

(empty)

Reactions:

$x_1 + y_2 + z_2 \rightarrow \emptyset$

$x_2 + y_1 + z_1 \rightarrow \emptyset$

$x_3 + y_2 + z_3 \rightarrow \emptyset$

$x_3 + y_3 + z_4 \rightarrow \emptyset$

Deletion-Only CRNs

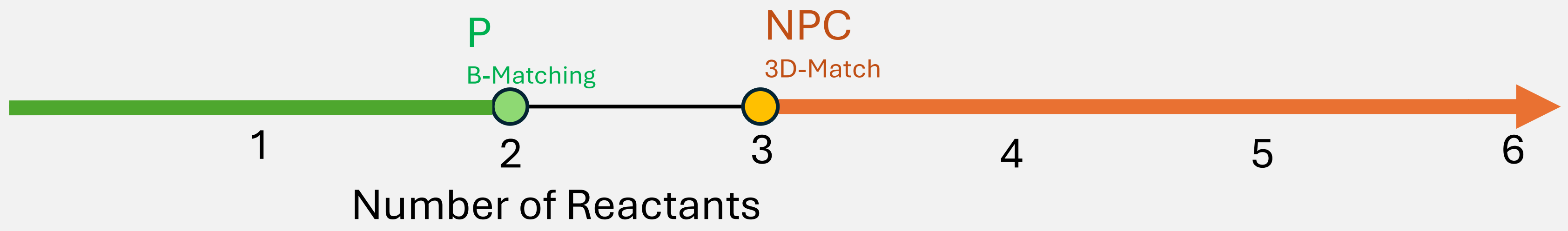
(2,0) rules

$a + b \to \varnothing$

Versus:

(3,0) rules

$a + b + c \to \varnothing$



Deletion-Only CRNs

(2,1) rules

a + b → a

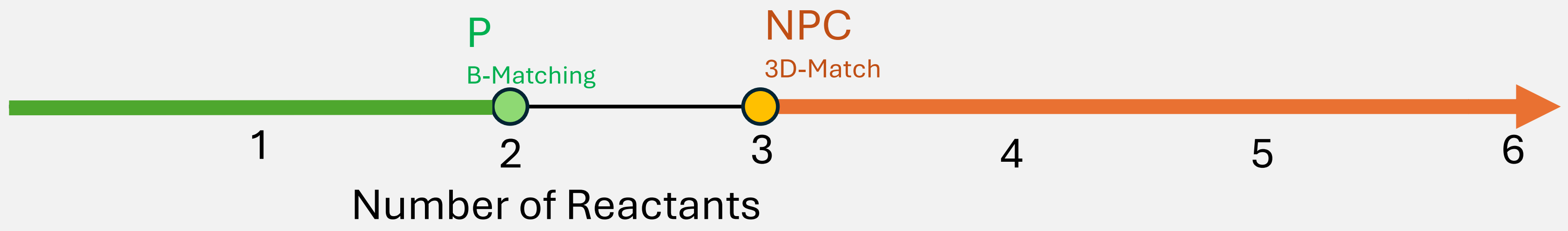
(3,1) rules

a + b + c → a

(3,2) rules

a + b + c → b + c

(etc.)



Deletion-Only CRNs

(2,1) rules

a + b → a

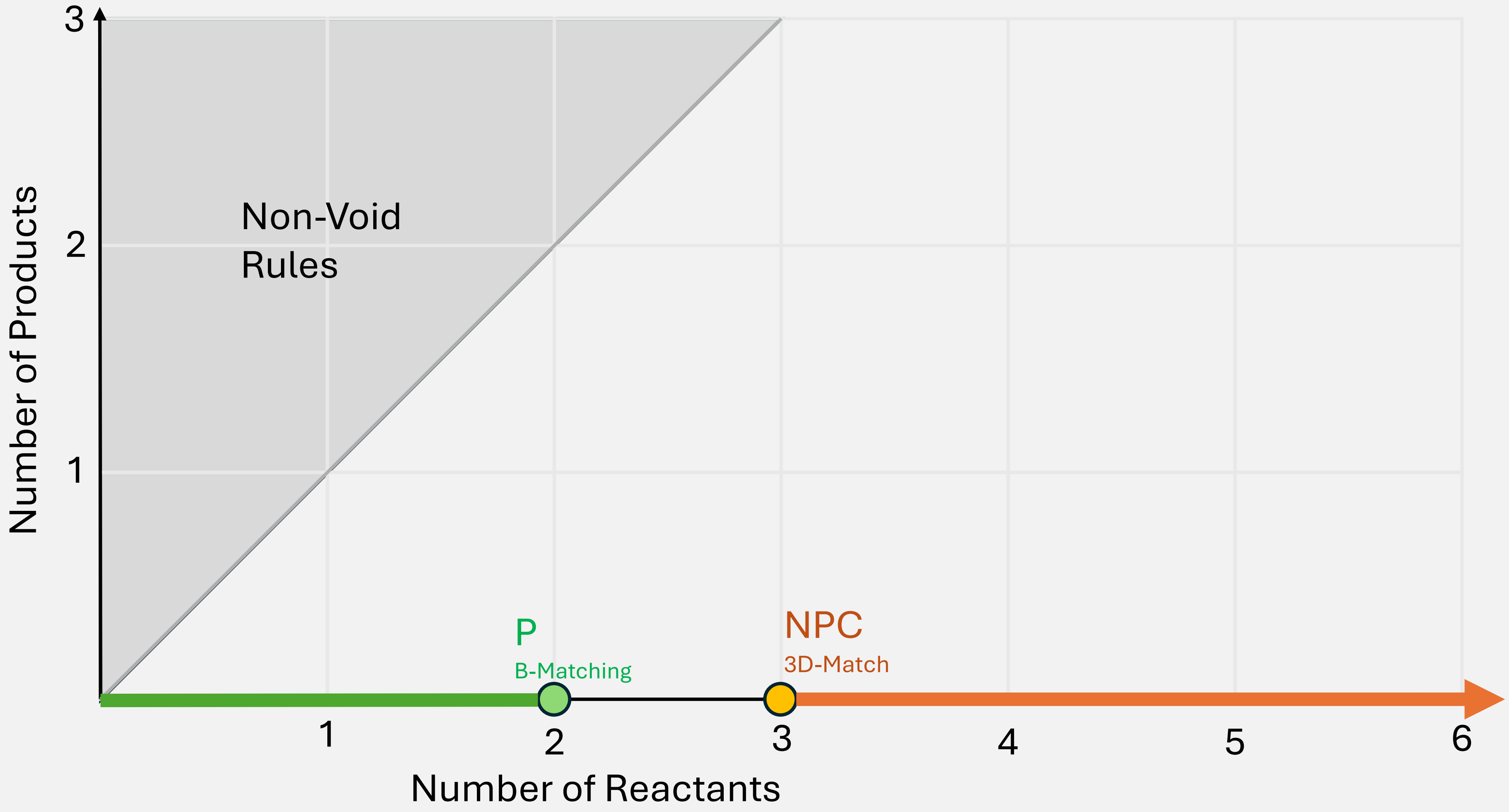
(3,1) rules

a + b + c → a

(3,2) rules

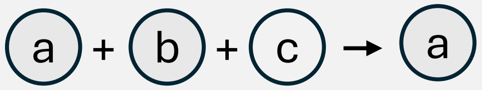
a + b + c → b + c

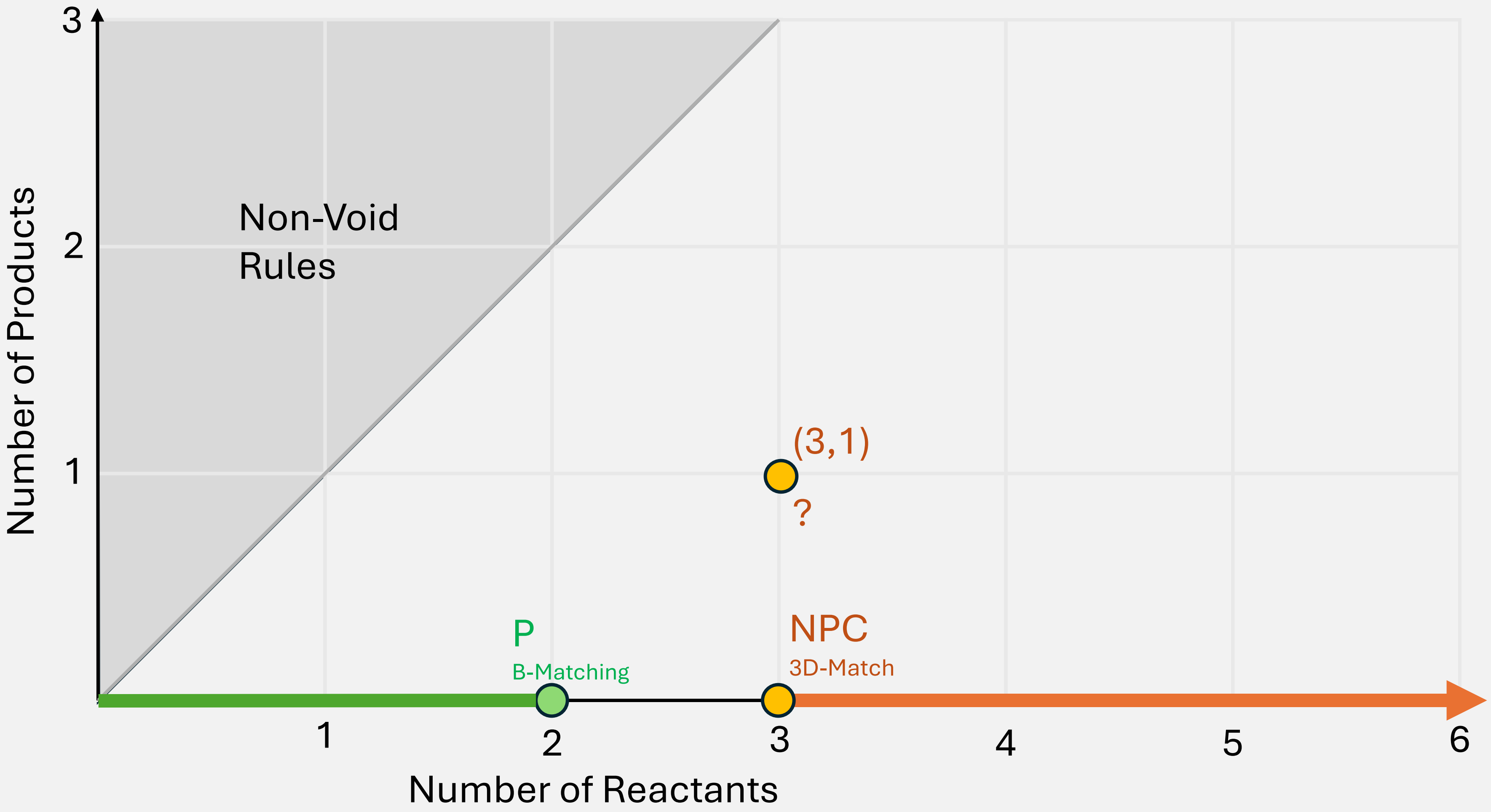
(etc.)



Deletion-Only CRNs

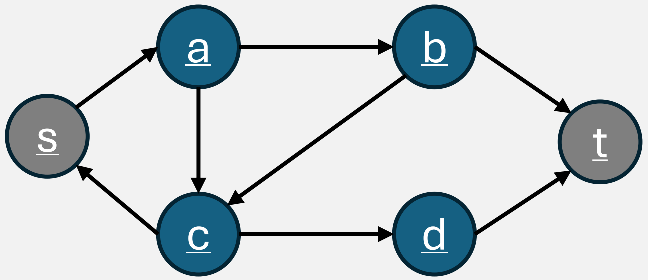
(3,1) rules





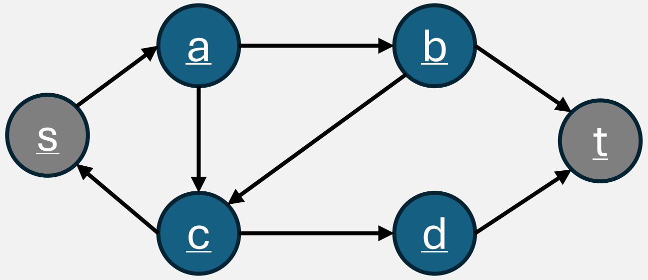
Deletion-Only CRNs: (3,1) NP-complete

<u>Hamiltonian Path from s to t?</u>



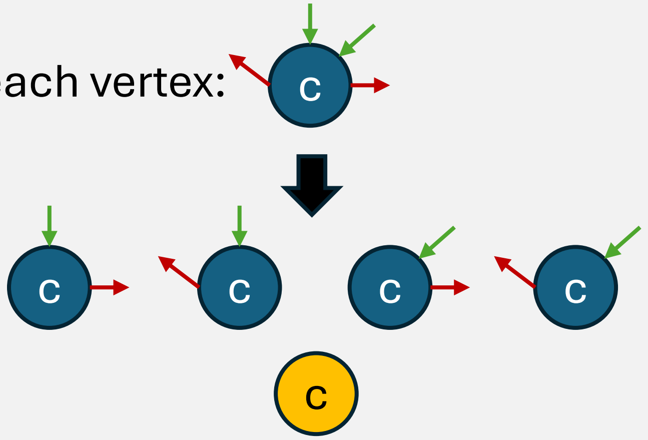
Deletion-Only CRNs: (3,1) NP-complete

Hamiltonian Path from s to t?



Initial Configuration:

For each vertex:

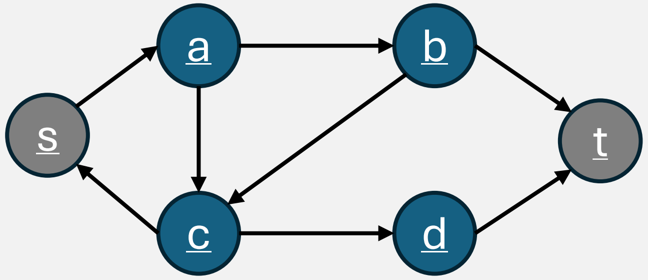


Target Configuration:

Eliminate **all** of these species
except for a single “t”

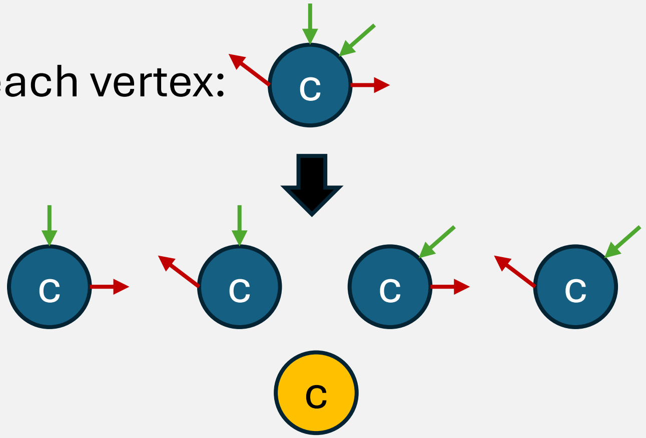
Deletion-Only CRNs: (3,1) NP-complete

Hamiltonian Path from s to t?



Initial Configuration:

For each vertex:



Target Configuration:

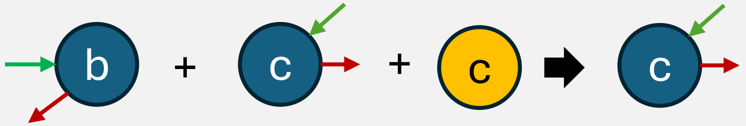
Eliminate **all** of these species except for a single “t”

Reactions:

1) Non-deterministically guess A traversal for each vertex:

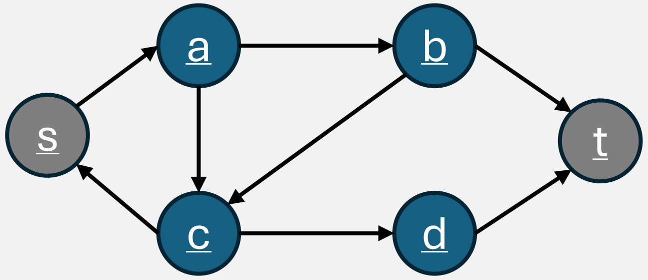
A+B+C -> A
A+B+C -> B
A+B+C -> C

2) Check if the guessed traversal constitutes a Hamiltonian Path:



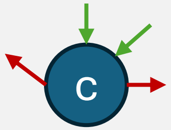
Deletion-Only CRNs: (3,1) NP-complete

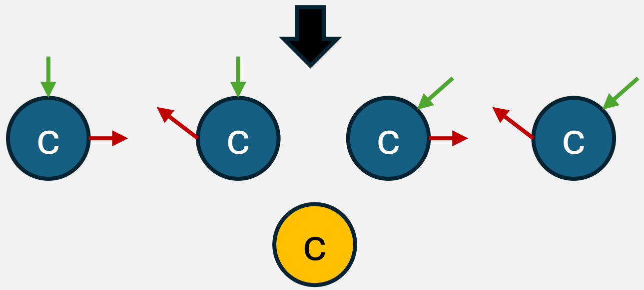
Hamiltonian Path from s to t?



Initial Configuration:

For each vertex:





Target Configuration:

Eliminate **all** of these species except for a single “t”

Reactions:

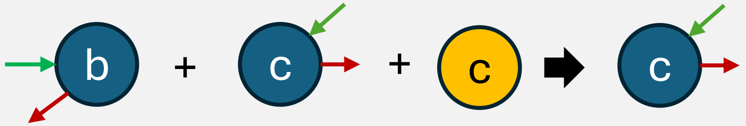
1) Non-deterministically guess A traversal for each vertex:

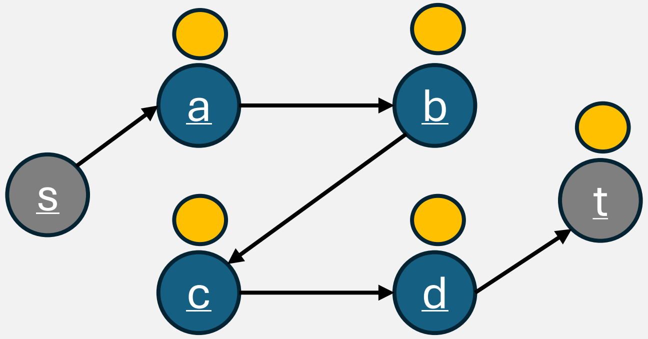
A+B+C -> A

A+B+C -> B

A+B+C -> C

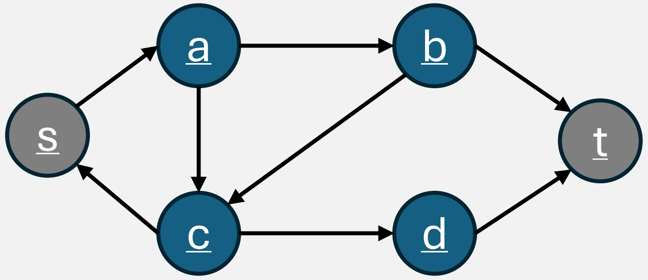
2) Check if the guessed traversal constitutes a Hamiltonian Path:





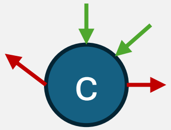
Deletion-Only CRNs: (3,1) NP-complete

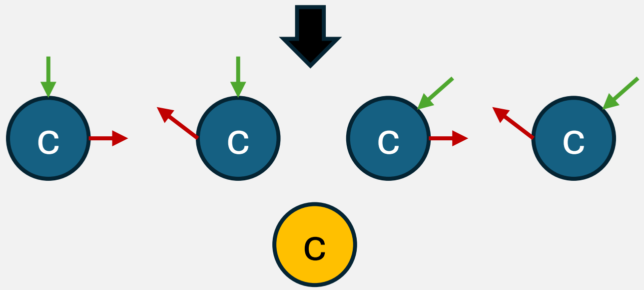
Hamiltonian Path from s to t?



Initial Configuration:

For each vertex:





Target Configuration:

Eliminate **all** of these species except for a single "t"

Reactions:

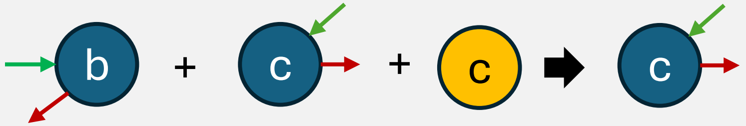
1) Non-deterministically guess A traversal for each vertex:

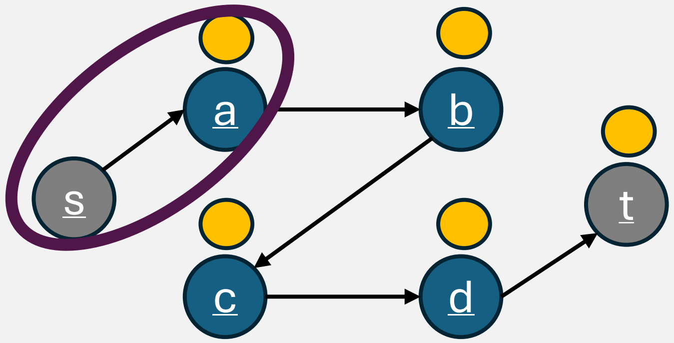
A+B+C -> A

A+B+C -> B

A+B+C -> C

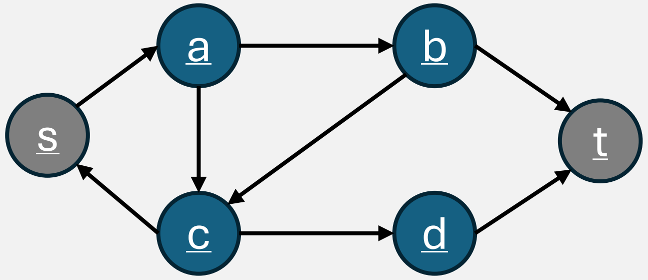
2) Check if the guessed traversal constitutes a Hamiltonian Path:



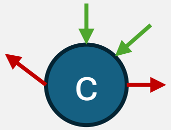


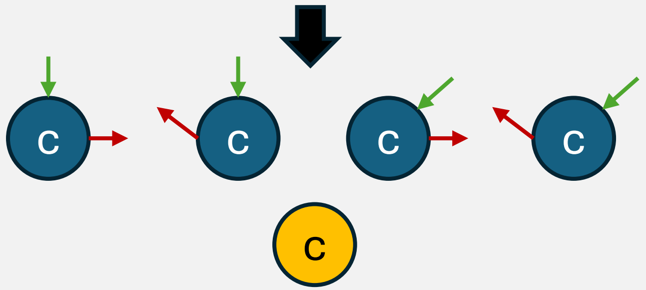
Deletion-Only CRNs: (3,1) NP-complete

Hamiltonian Path from s to t?



Initial Configuration:

For each vertex: 



Target Configuration:

Eliminate **all** of these species except for a single “t”

Reactions:

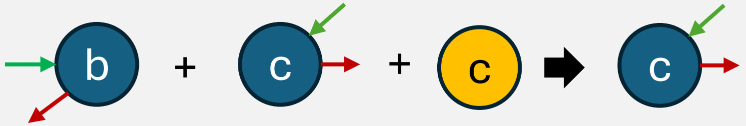
1) Non-deterministically guess A traversal for each vertex:

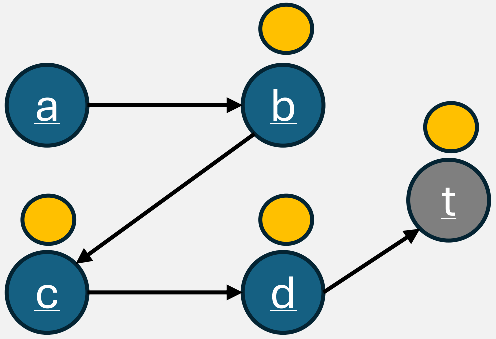
A+B+C -> A

A+B+C -> B

A+B+C -> C

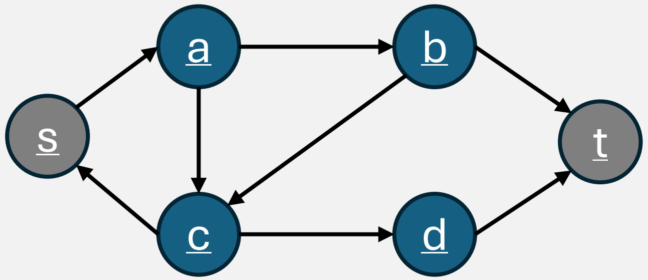
2) Check if the guessed traversal constitutes a Hamiltonian Path:





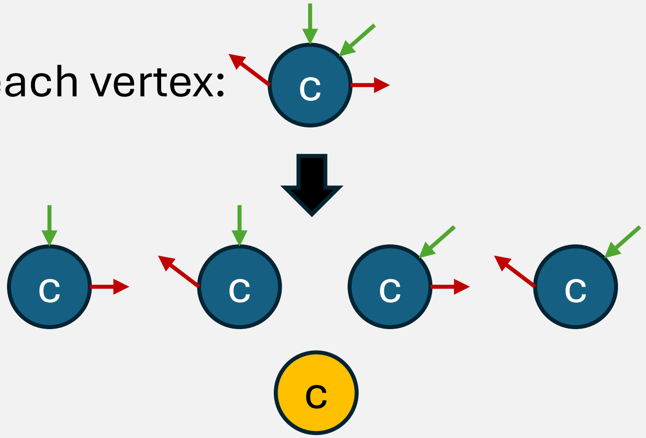
Deletion-Only CRNs: (3,1) NP-complete

<u>Hamiltonian Path from s to t?</u>



Initial Configuration:

For each vertex:



Target Configuration:

Eliminate **all** of these species except for a single “t”

Reactions:

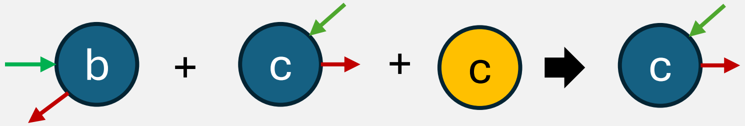
1) Non-deterministically guess A traversal for each vertex:

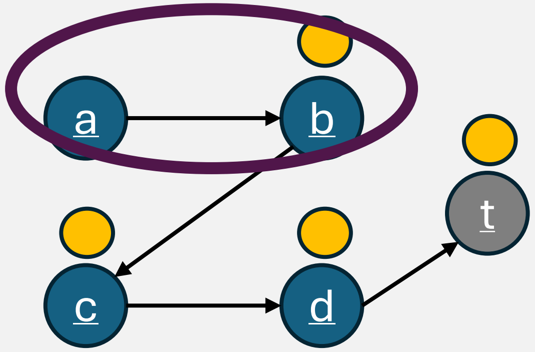
A+B+C -> A

A+B+C -> B

A+B+C -> C

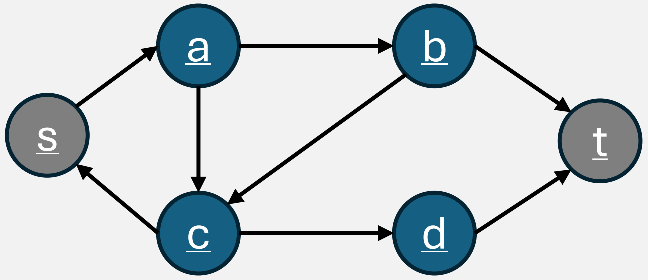
2) Check if the guessed traversal constitutes a Hamiltonian Path:





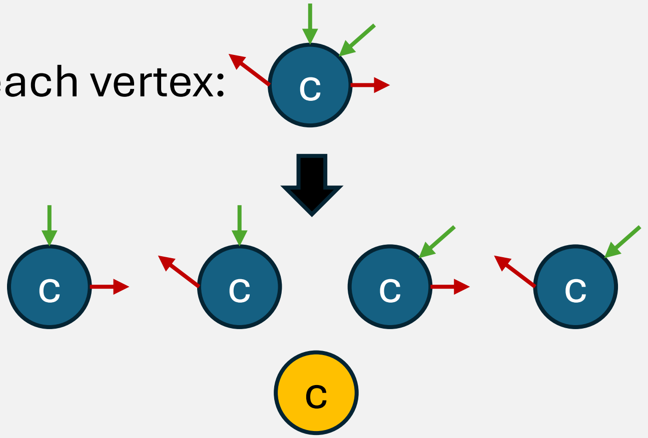
Deletion-Only CRNs: (3,1) NP-complete

Hamiltonian Path from s to t?



Initial Configuration:

For each vertex:



Target Configuration:

Eliminate **all** of these species except for a single “t”

Reactions:

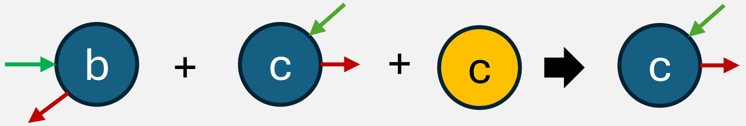
1) Non-deterministically guess A traversal for each vertex:

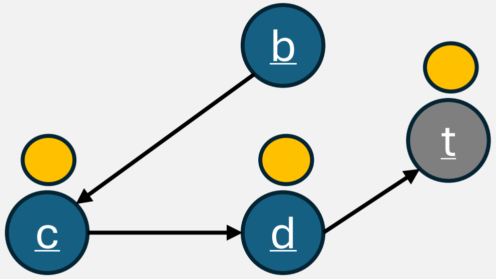
A+B+C -> A

A+B+C -> B

A+B+C -> C

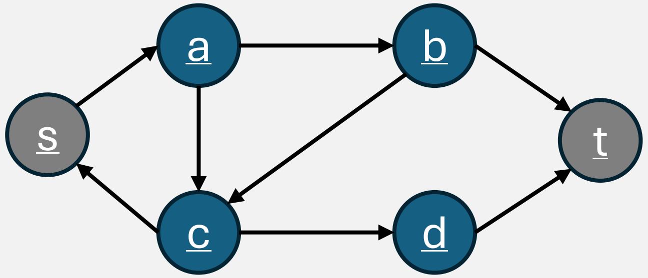
2) Check if the guessed traversal constitutes a Hamiltonian Path:





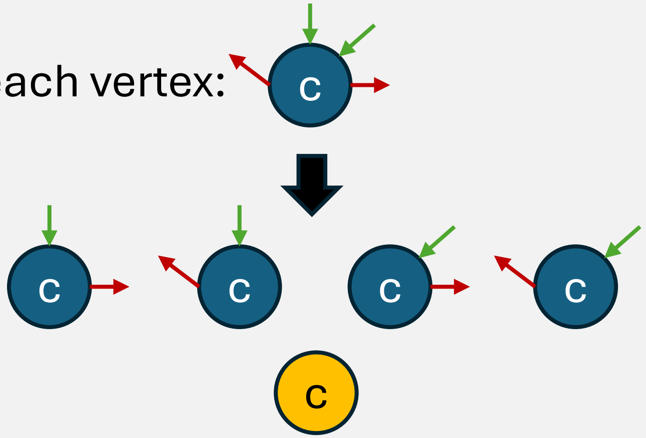
Deletion-Only CRNs: (3,1) NP-complete

Hamiltonian Path from s to t?



Initial Configuration:

For each vertex:



Target Configuration:

Eliminate **all** of these species except for a single “t”

Reactions:

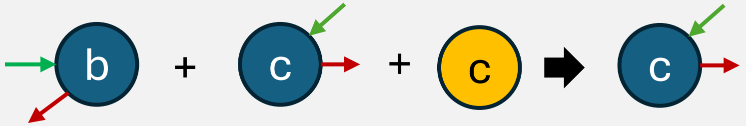
1) Non-deterministically guess A traversal for each vertex:

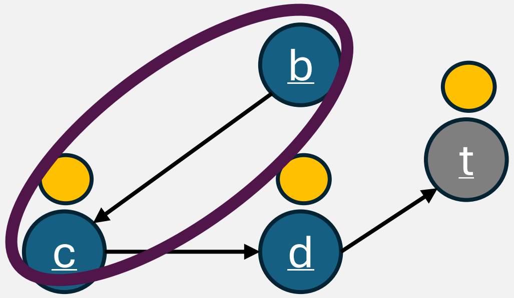
A+B+C -> A

A+B+C -> B

A+B+C -> C

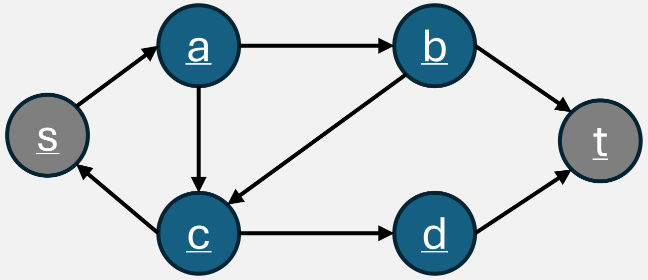
2) Check if the guessed traversal constitutes a Hamiltonian Path:



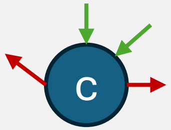


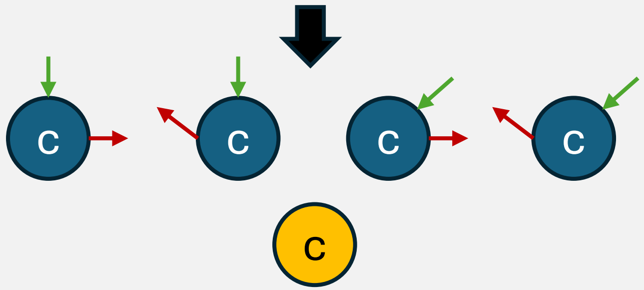
Deletion-Only CRNs: (3,1) NP-complete

Hamiltonian Path from s to t?



Initial Configuration:

For each vertex: 



Target Configuration:

Eliminate **all** of these species except for a single “t”

Reactions:

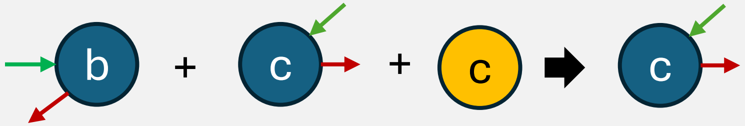
1) Non-deterministically guess A traversal for each vertex:

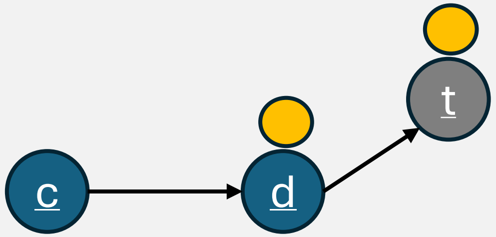
A+B+C -> A

A+B+C -> B

A+B+C -> C

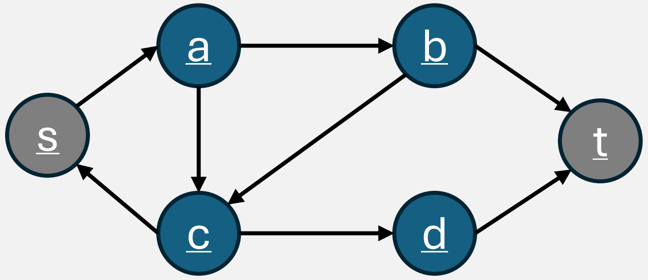
2) Check if the guessed traversal constitutes a Hamiltonian Path:





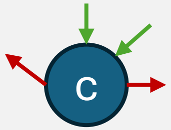
Deletion-Only CRNs: (3,1) NP-complete

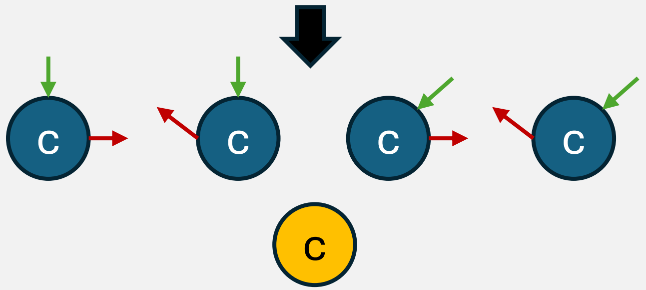
Hamiltonian Path from s to t?



Initial Configuration:

For each vertex:





Target Configuration:

Eliminate **all** of these species except for a single "t"

Reactions:

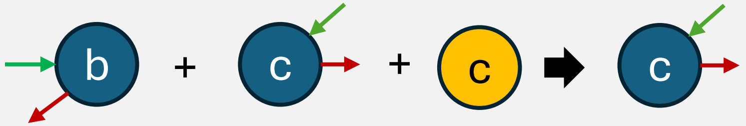
1) Non-deterministically guess A traversal for each vertex:

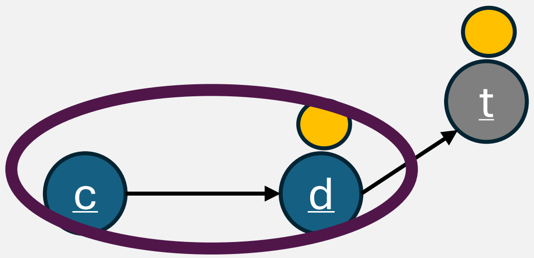
A+B+C -> A

A+B+C -> B

A+B+C -> C

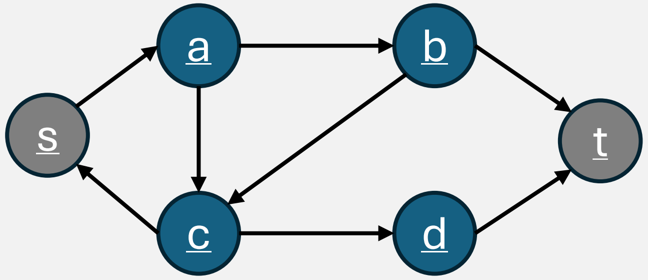
2) Check if the guessed traversal constitutes a Hamiltonian Path:





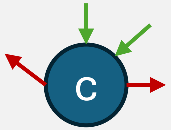
Deletion-Only CRNs: (3,1) NP-complete

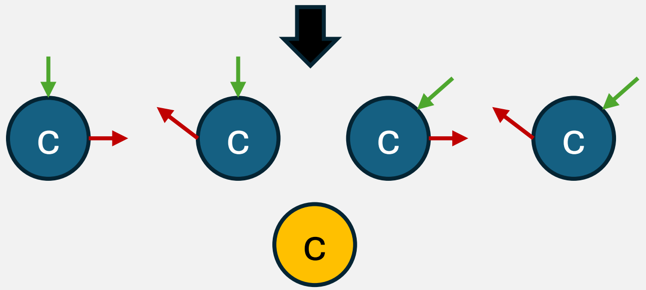
Hamiltonian Path from s to t?



Initial Configuration:

For each vertex:





Target Configuration:

Eliminate **all** of these species except for a single "t"

Reactions:

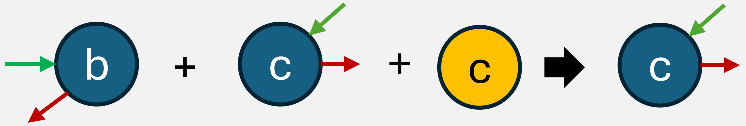
1) Non-deterministically guess A traversal for each vertex:

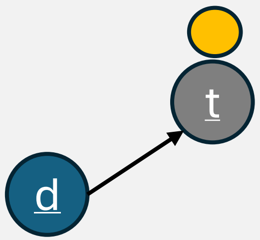
A+B+C -> A

A+B+C -> B

A+B+C -> C

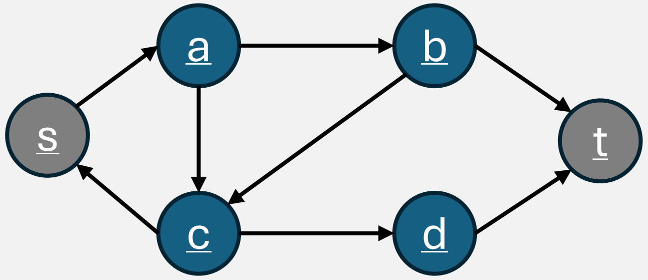
2) Check if the guessed traversal constitutes a Hamiltonian Path:



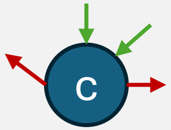


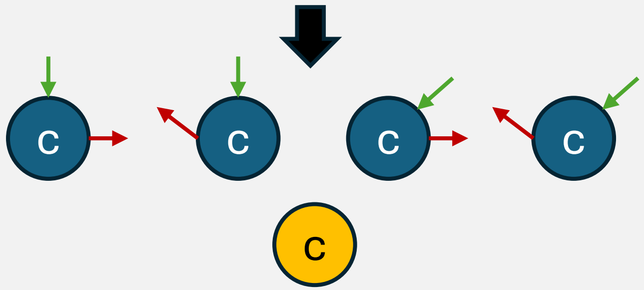
Deletion-Only CRNs: (3,1) NP-complete

Hamiltonian Path from s to t?



Initial Configuration:

For each vertex: 



Target Configuration:

Eliminate **all** of these species except for a single “t”

Reactions:

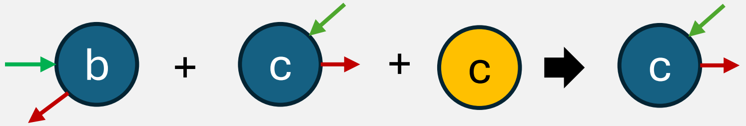
1) Non-deterministically guess A traversal for each vertex:

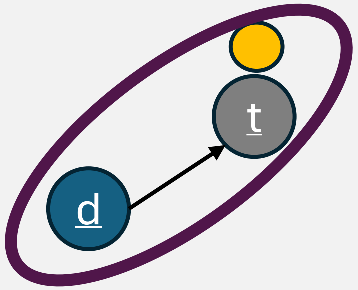
A+B+C -> A

A+B+C -> B

A+B+C -> C

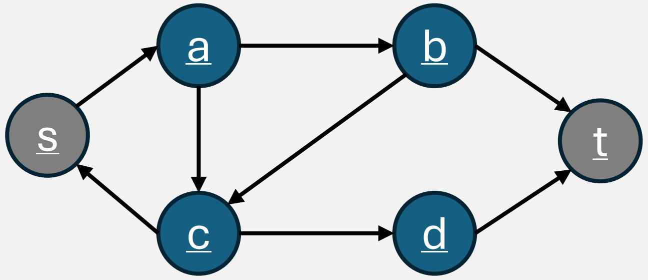
2) Check if the guessed traversal constitutes a Hamiltonian Path:



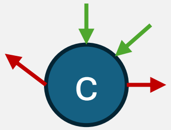


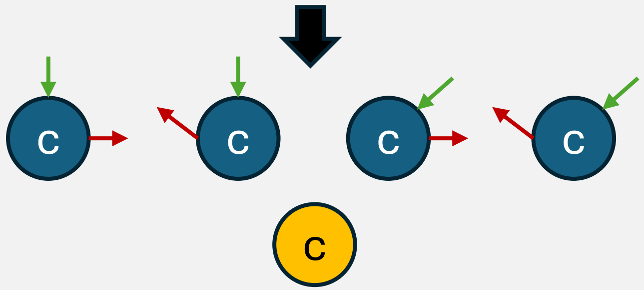
Deletion-Only CRNs: (3,1) NP-complete

Hamiltonian Path from s to t?



Initial Configuration:

For each vertex: 



Target Configuration:

Eliminate **all** of these species except for a single “t”

Reactions:

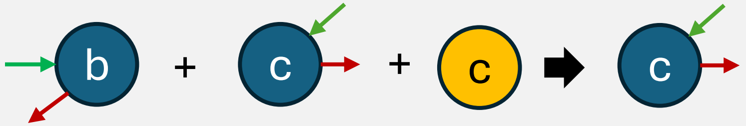
1) Non-deterministically guess A traversal for each vertex:

A+B+C -> A

A+B+C -> B

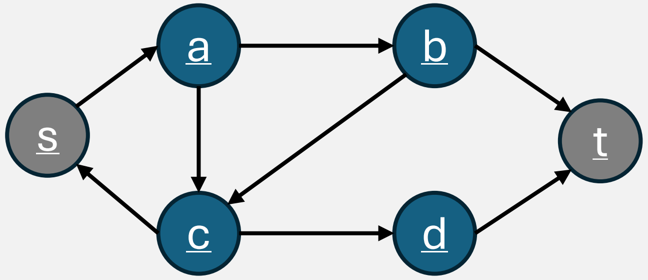
A+B+C -> C

2) Check if the guessed traversal constitutes a Hamiltonian Path:



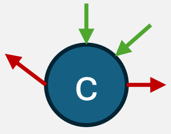
Deletion-Only CRNs: (3,1) NP-complete

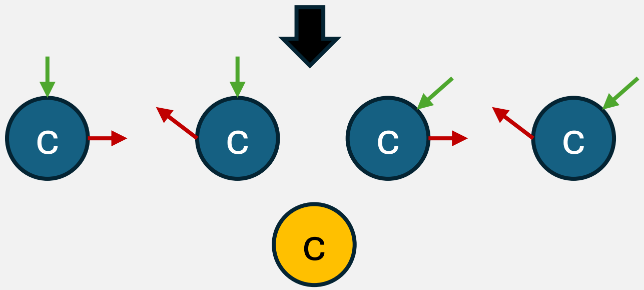
<u>Hamiltonian Path from s to t?</u>



Initial Configuration:

For each vertex:





Target Configuration:

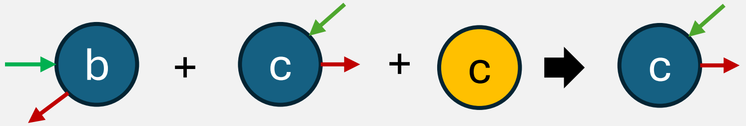
Eliminate **all** of these species except for a single "t"

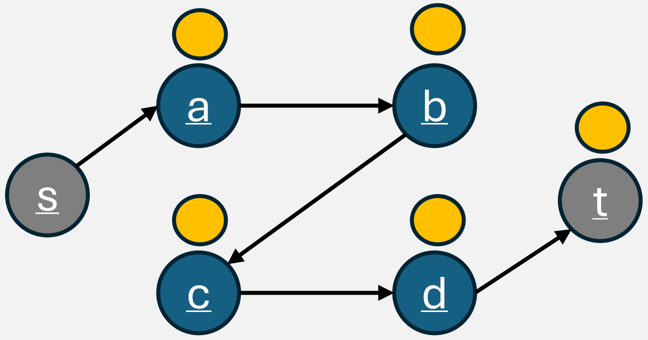
Reactions:

1) Non-deterministically guess A traversal for each vertex:

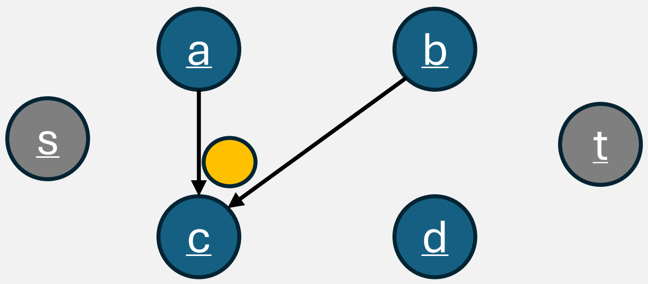
A+B+C -> A
A+B+C -> B
A+B+C -> C

2) Check if the guessed traversal constitutes a Hamiltonian Path:



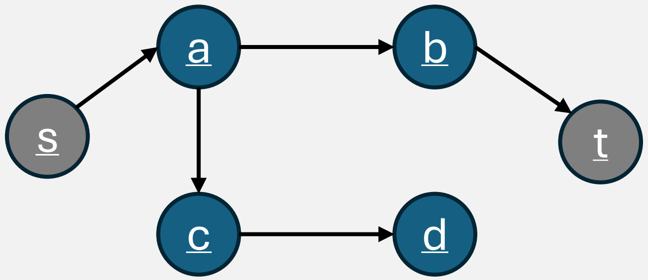


No "double visits"



Only one c

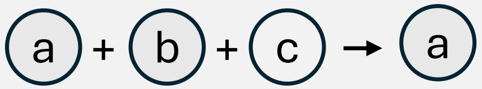
No branching

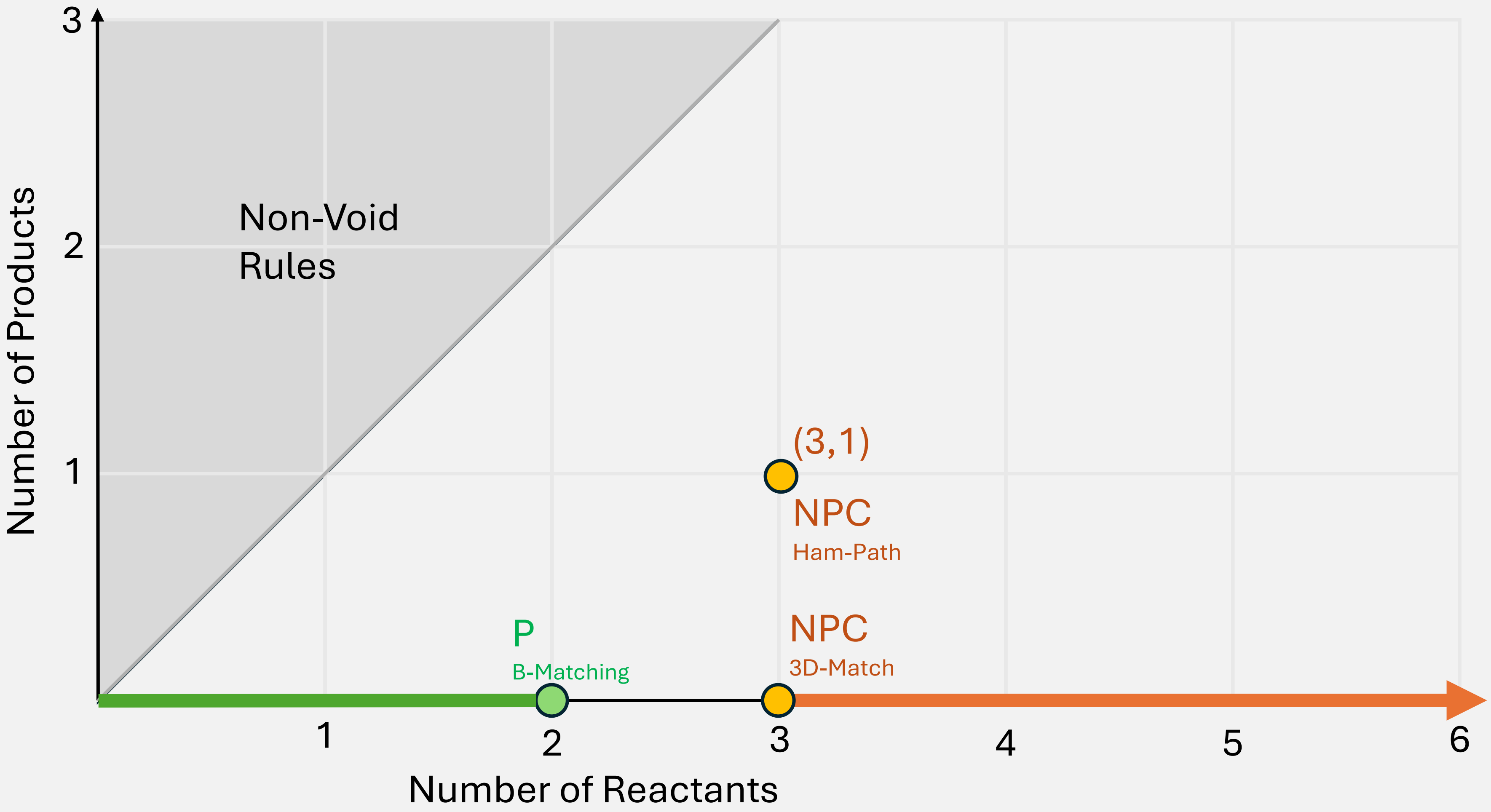


Fails to delete d

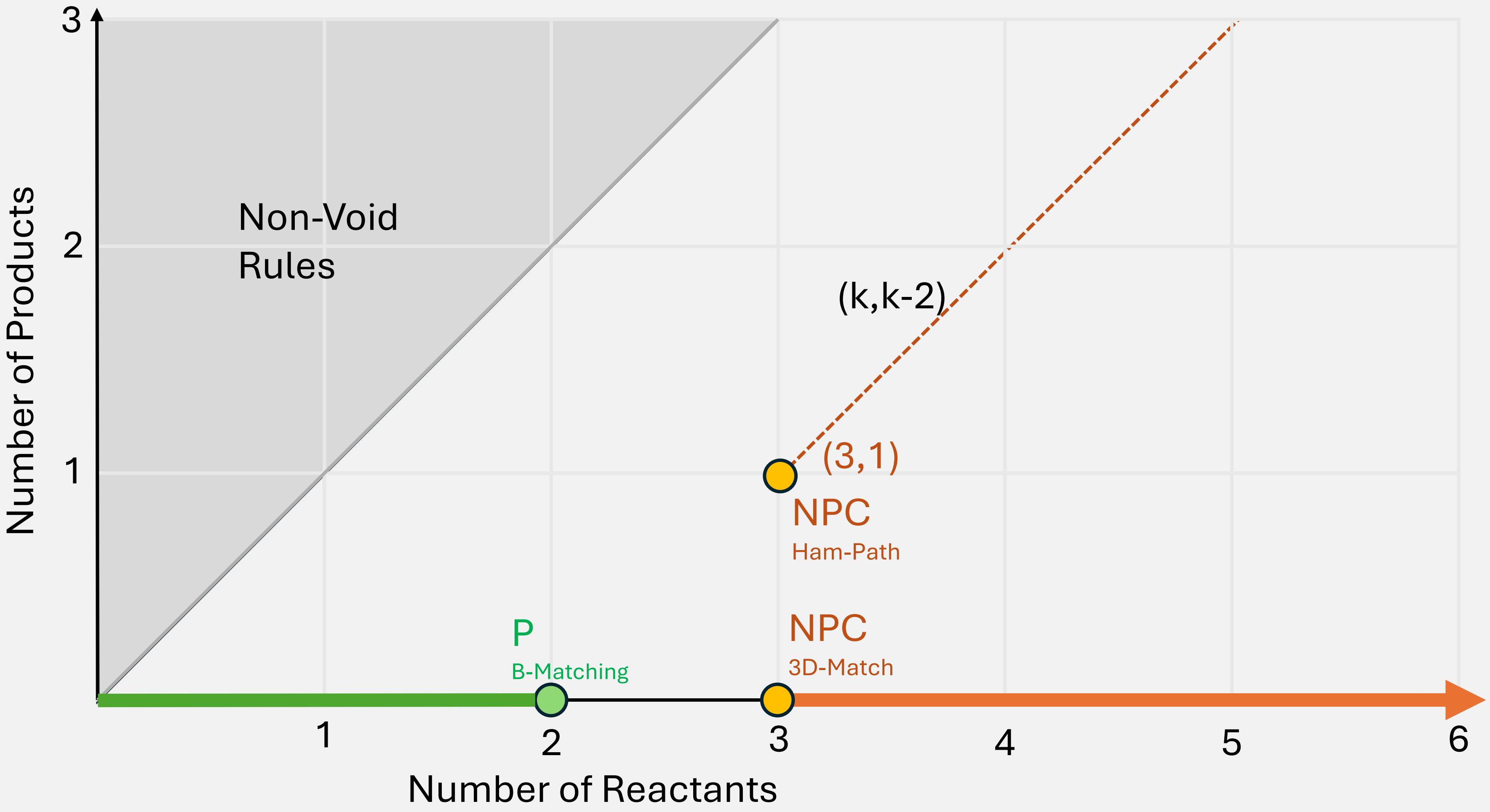
Deletion-Only CRNs

(3,1) rules

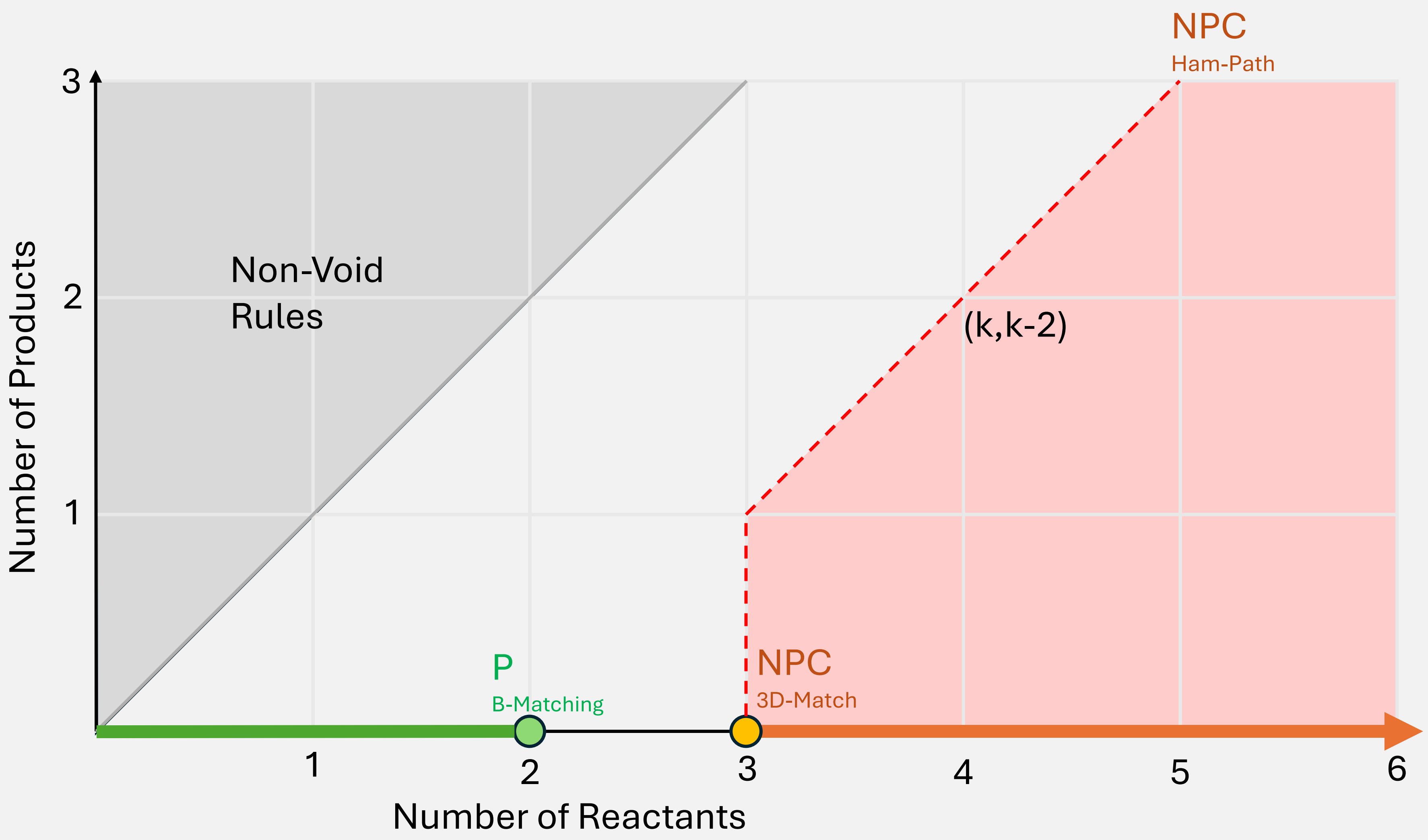




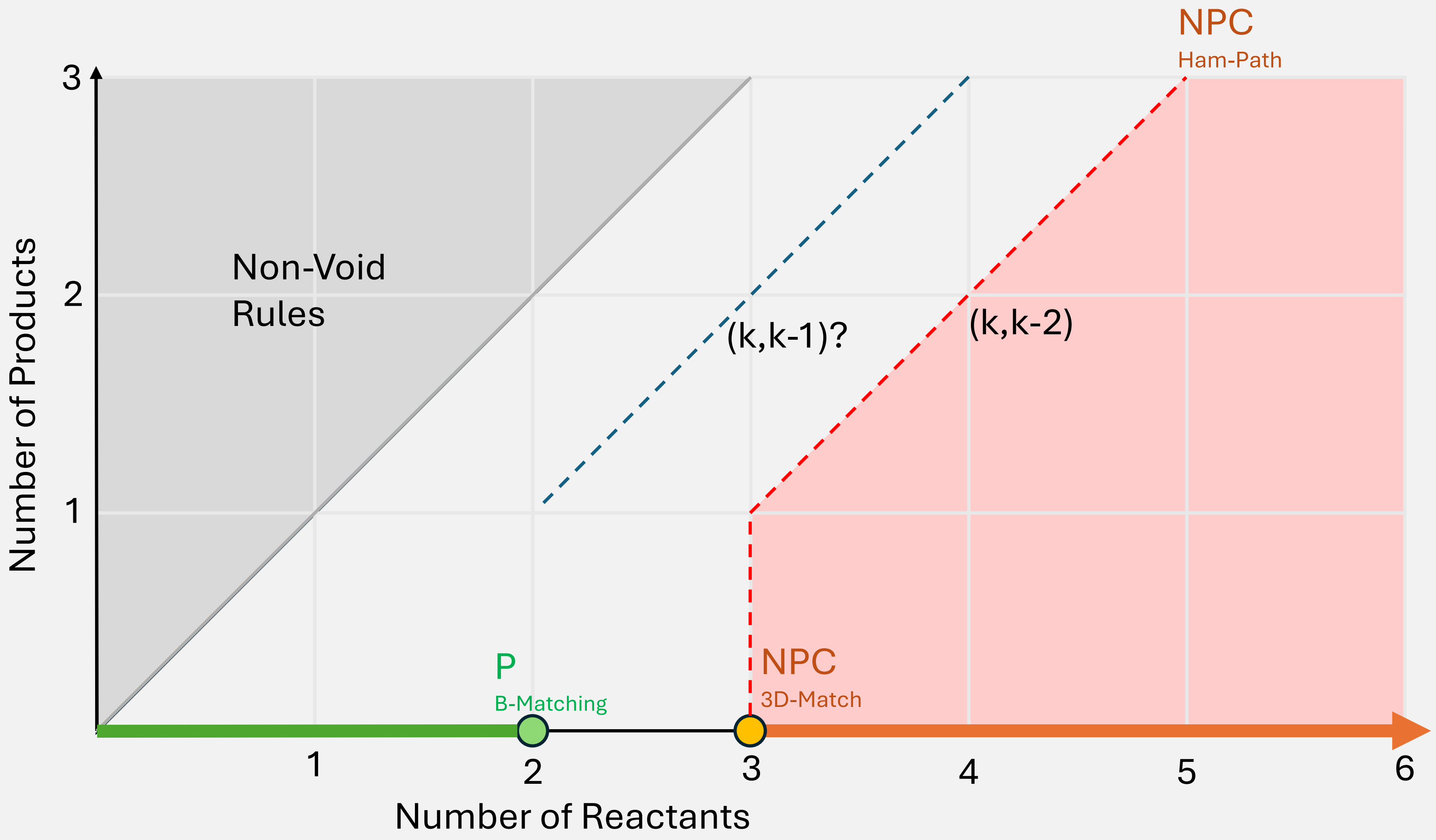
Deletion-Only CRNs



Deletion-Only CRNs

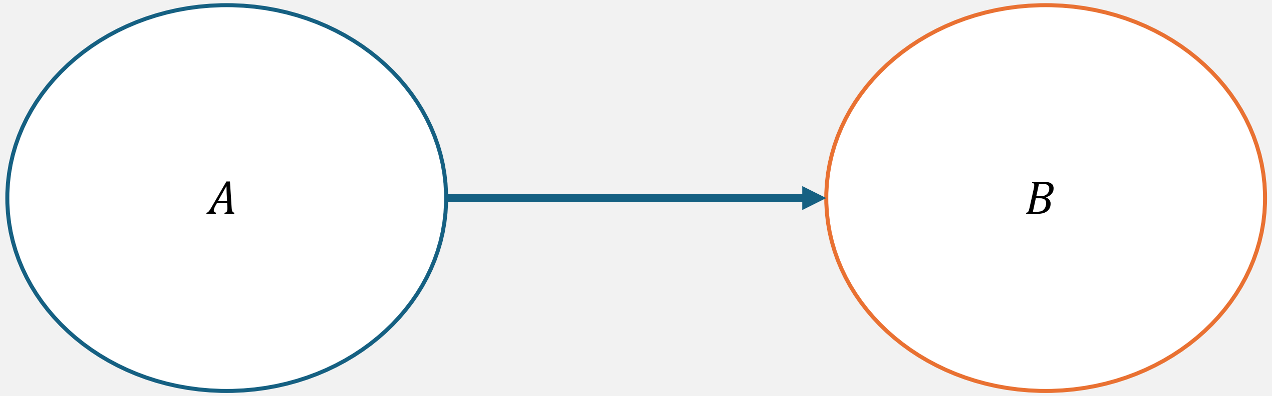


Deletion-Only CRNs



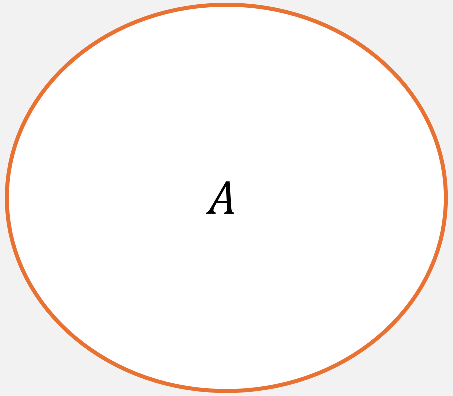
Reaction Graph

- For rule $A + B \rightarrow A$ add edge between A and B. Mark species with a target value > 0



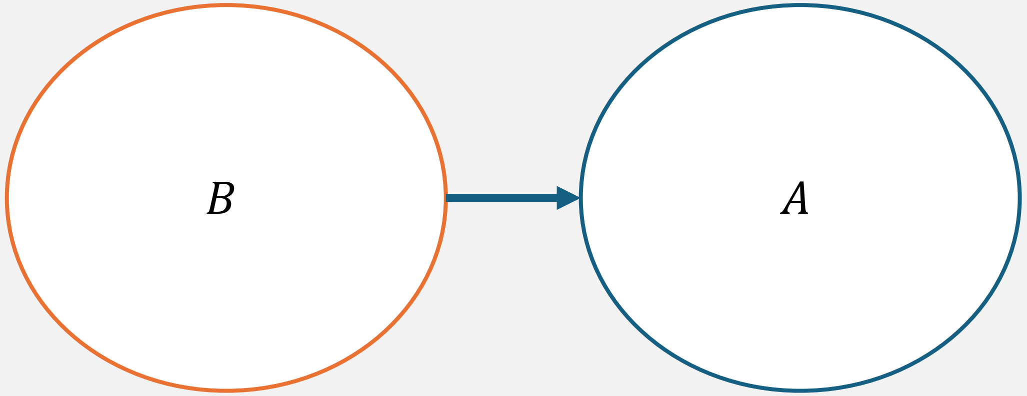
(2,1) Rules

- A species is a root if one of the following is true
  - The species starts with the correct number of copies



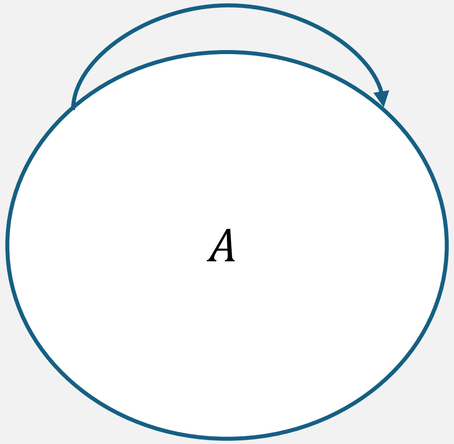
(2,1) Rules

- A species is a root if one of the following is true
  - The species starts with the correct number of copies
  - The species can be deleted by some marked species.
  - The species has a self loop.



(2,1) Rules

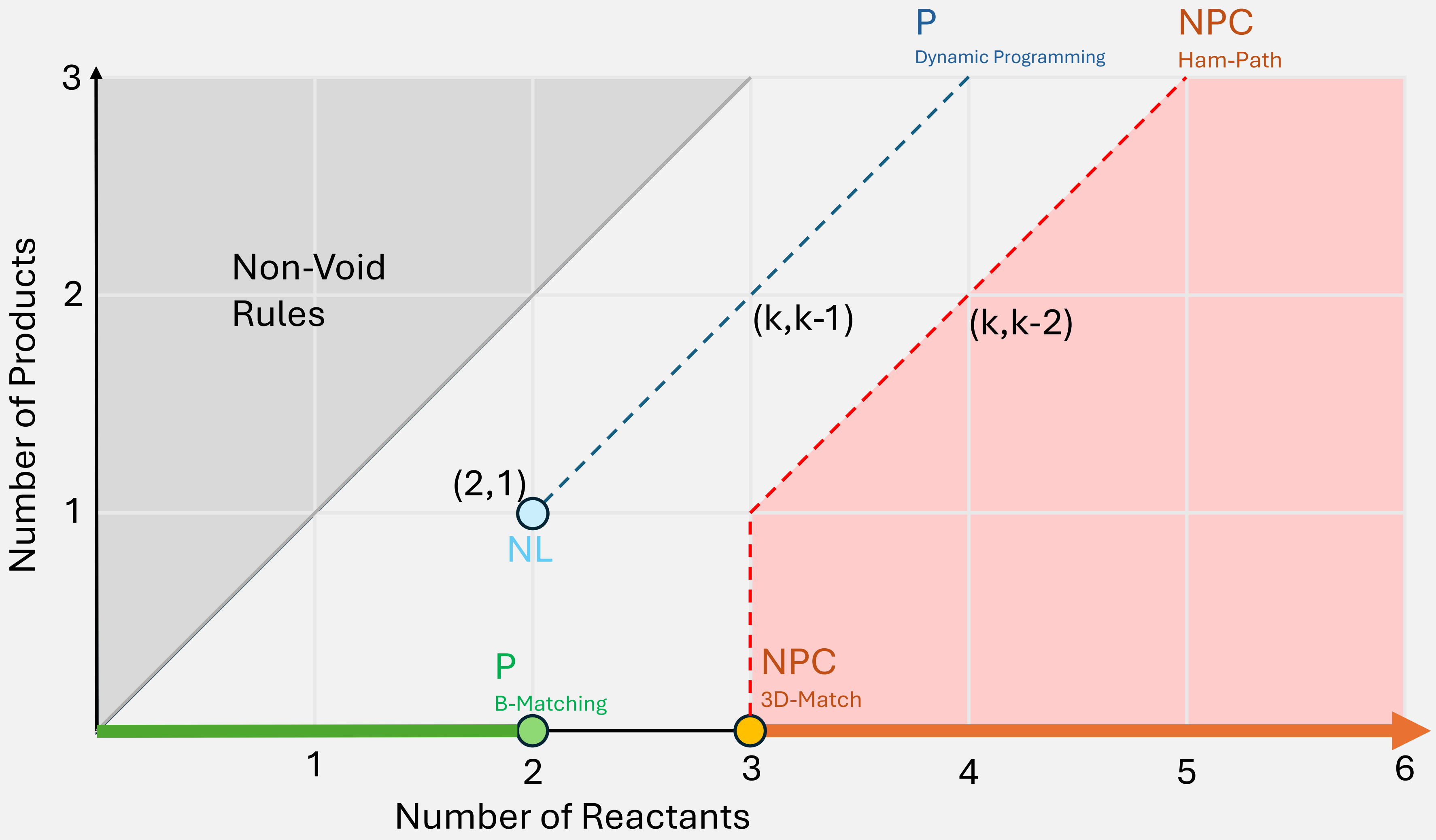
- A species is a root if one of the following is true
  - The species starts with the correct number of copies
  - The species can be deleted by some marked species.
  - The species has a self loop.



(2,1) Rules

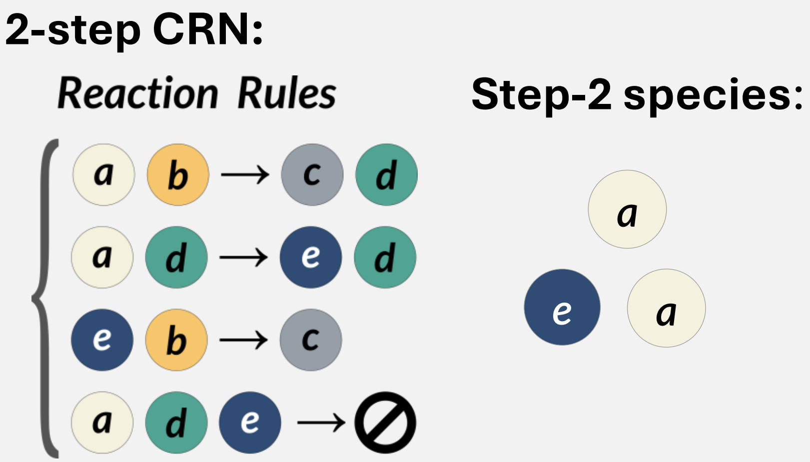
- Checking if a species is a root can be done in log space
- For each species we must find another one that
  - Is reachable in the Reaction Graph
  - Is a root species
  - This takes 2 pointers
- Reachability in a directed graph can be solved in non-deterministic log space.

Deletion-Only CRNs



2-Step CRNs

2-Step CRN systems include a multiset of “step-2” species to be added after the system is terminal:

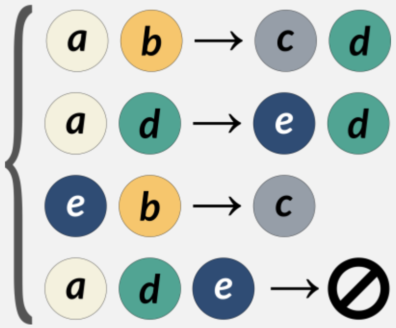


2-Step CRNs

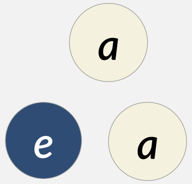
2-Step CRN systems include a multiset of "step-2" species to be added after the system is terminal:

2-step CRN:

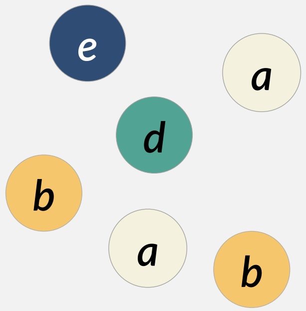
Reaction Rules



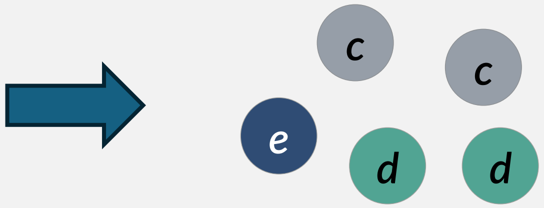
Step-2 species:



Configuration I:

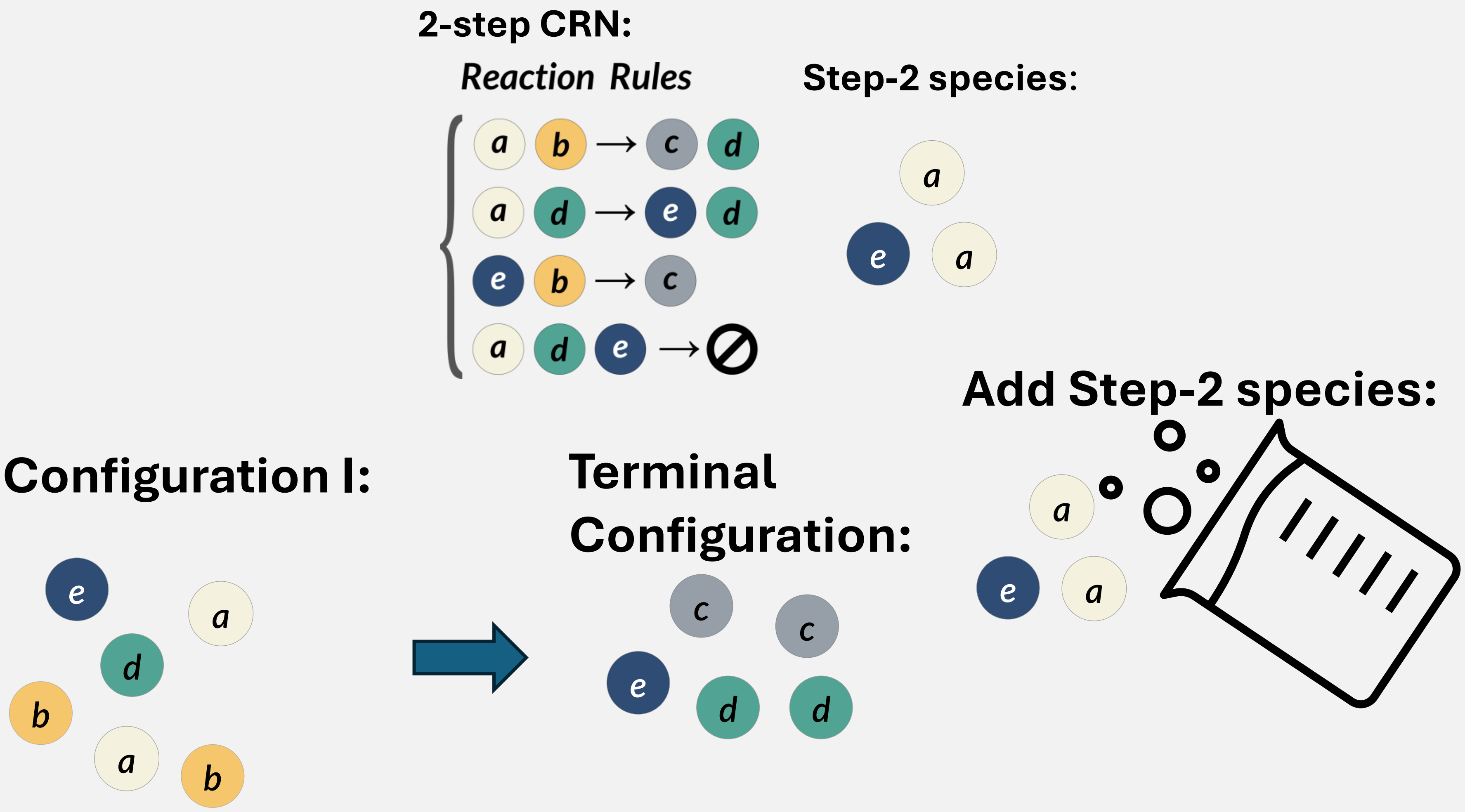


Terminal Configuration:



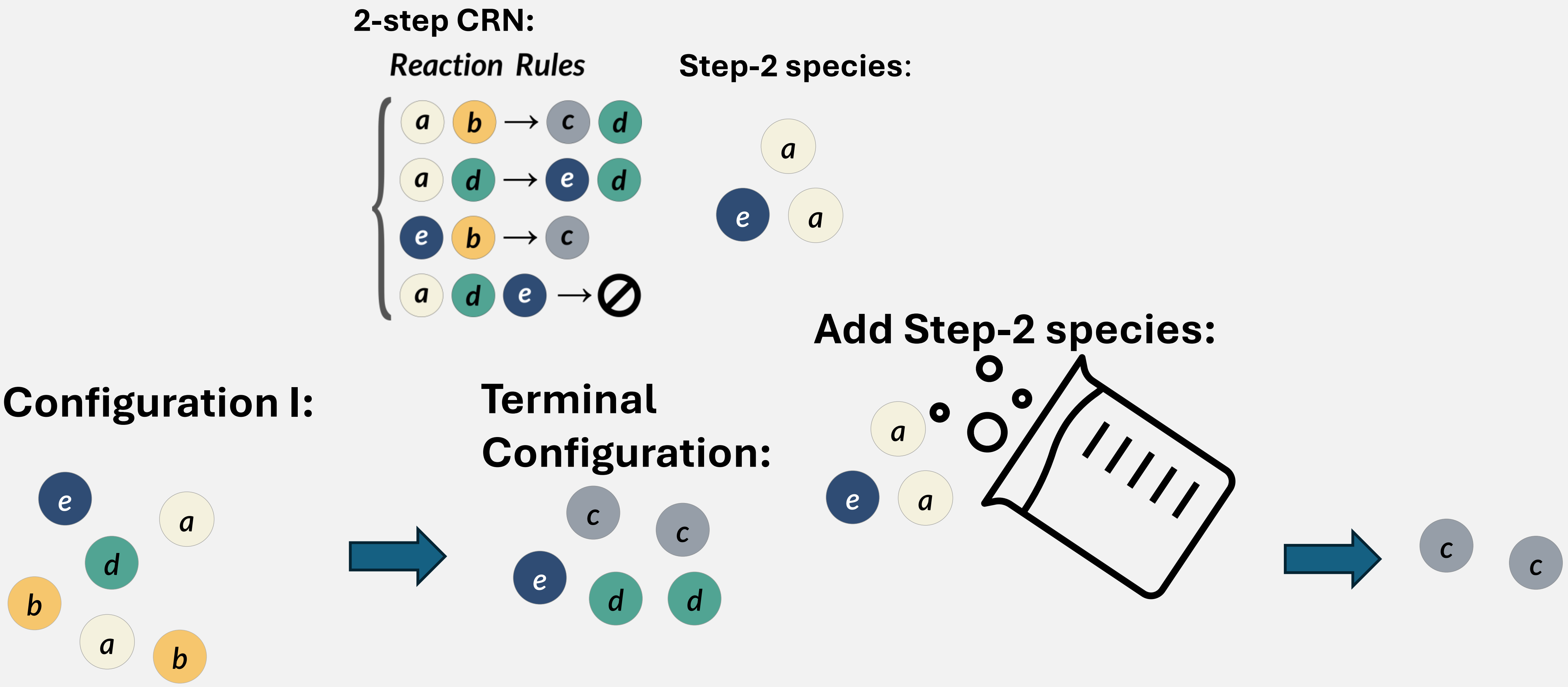
2-Step CRNs

2-Step CRN systems include a multiset of “step-2” species to be added after the system is terminal:

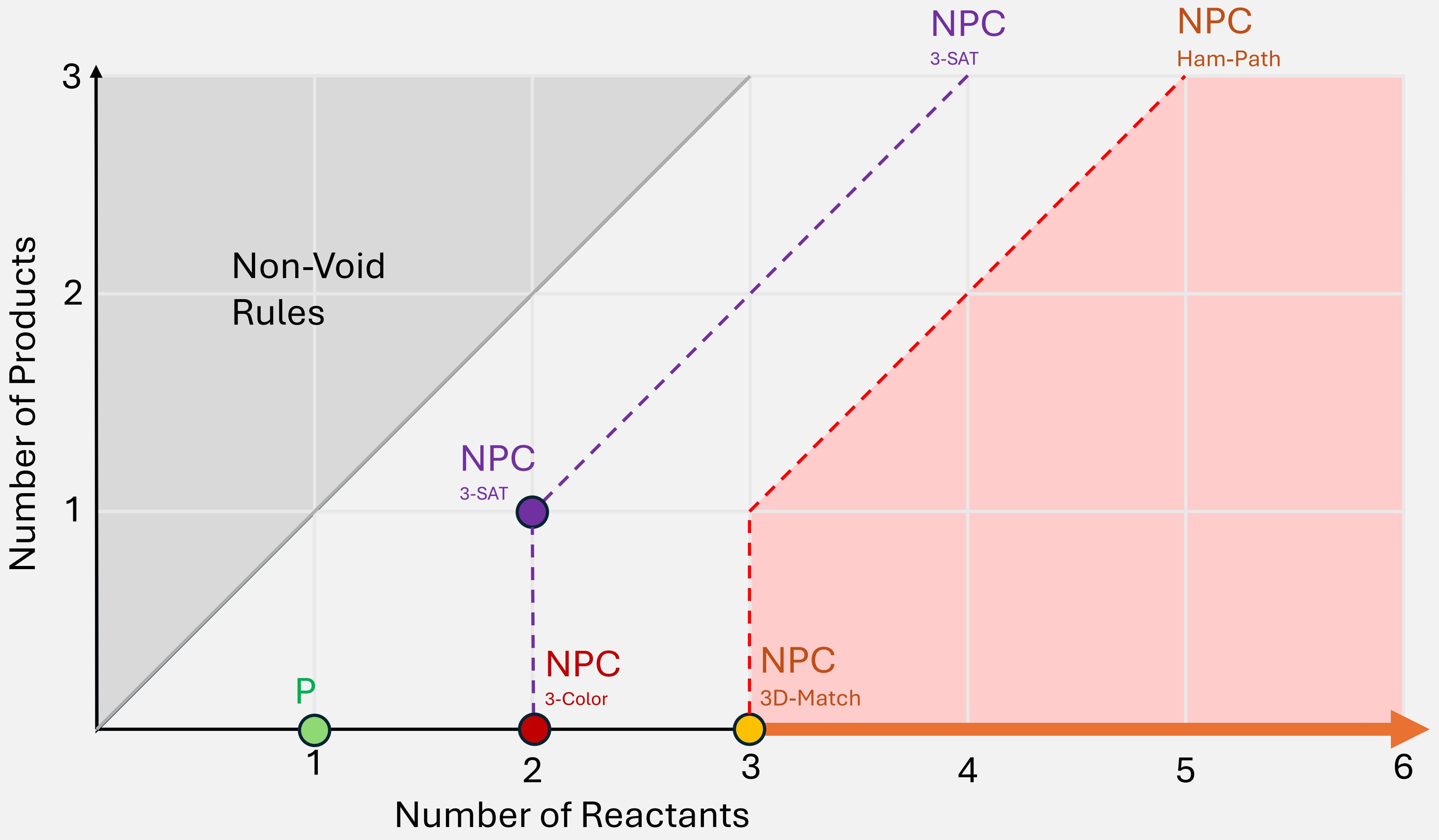


2-Step CRNs

2-Step CRN systems include a multiset of “step-2” species to be added after the system is terminal:

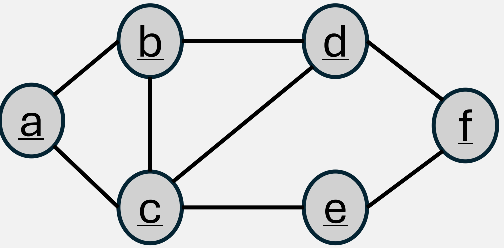


2-Step Deletion-Only CRNs



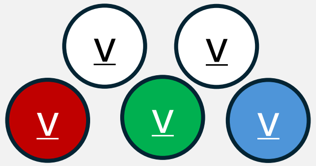
Deletion-Only 2-Step CRNs: (2,0) NP-complete

3-Color Input:



Initial Configuration:

For each vertex v:



Target Configuration:

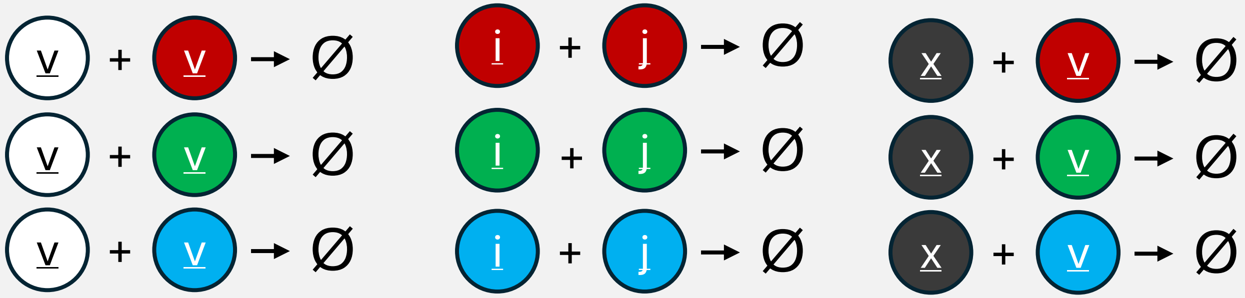
Eliminate **all** of these species

Reactions:

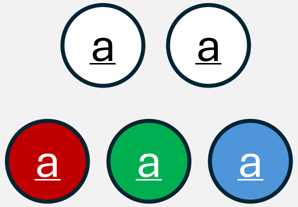
For each vertex v:

For each edge (i,j):

For all vertices V:

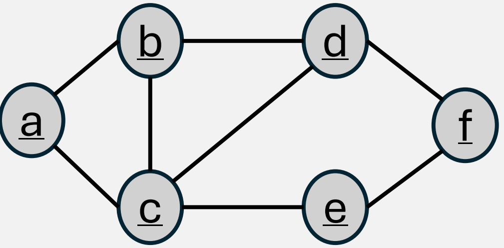


2nd Step Additions: |V| copies of 



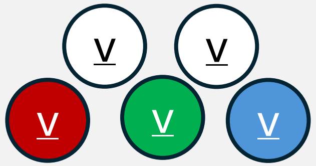
Deletion-Only 2-Step CRNs: (2,0) NP-complete

3-Color Input:



Initial Configuration:

For each vertex v:



Target Configuration:

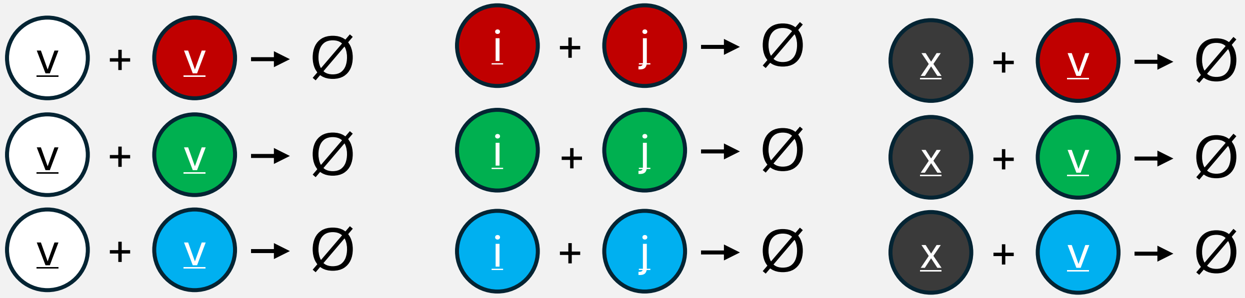
Eliminate **all** of these species

Reactions:

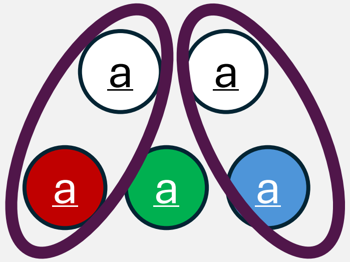
For each vertex v:

For each edge (i,j):

For all vertices V:

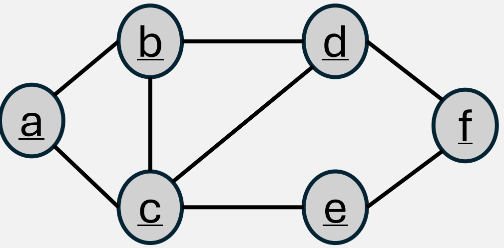


2nd Step Additions: |V| copies of x



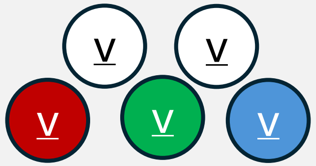
Deletion-Only 2-Step CRNs: (2,0) NP-complete

3-Color Input:



Initial Configuration:

For each vertex v:



Target Configuration:

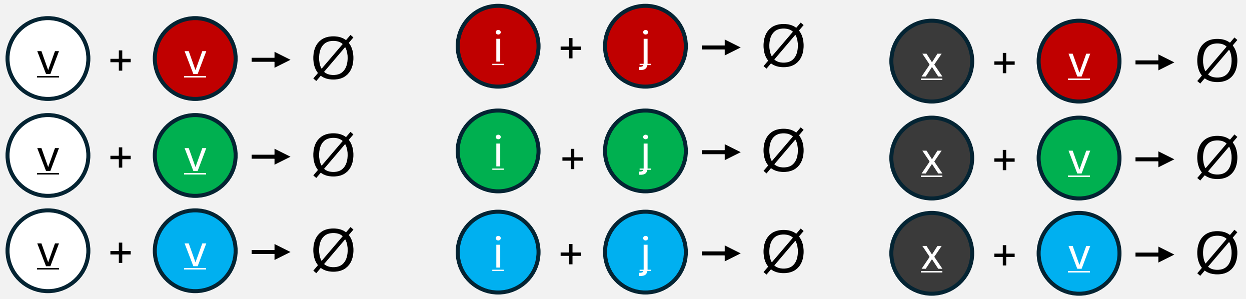
Eliminate **all** of these species

Reactions:

For each vertex v:

For each edge (i,j):

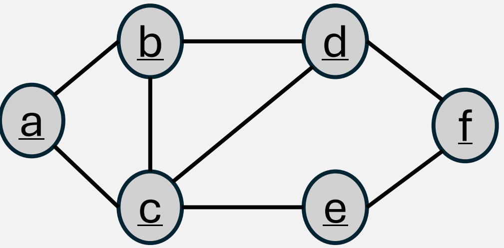
For all vertices V:



2nd Step Additions: |V| copies of 

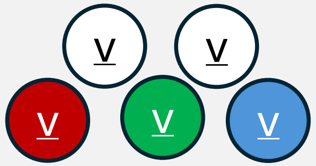
Deletion-Only 2-Step CRNs: (2,0) NP-complete

3-Color Input:



Initial Configuration:

For each vertex v:

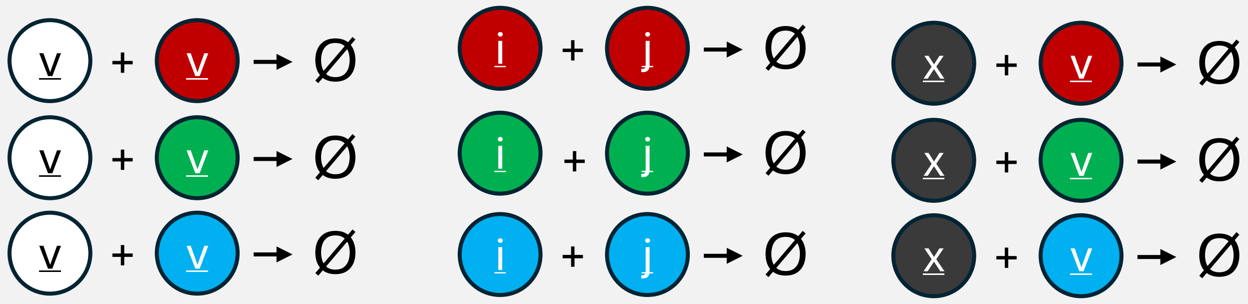


Target Configuration:

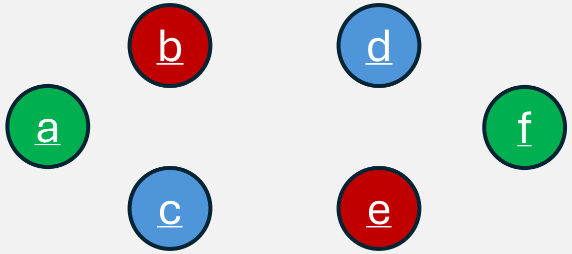
Eliminate **all** of these species

Reactions:

For each vertex v: For each edge (i,j): For all vertices V:

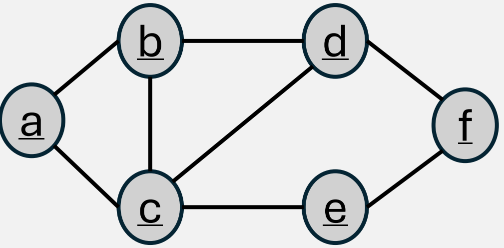


2nd Step Additions: |V| copies of 



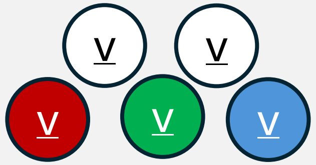
Deletion-Only 2-Step CRNs: (2,0) NP-complete

3-Color Input:



Initial Configuration:

For each vertex v:



Target Configuration:

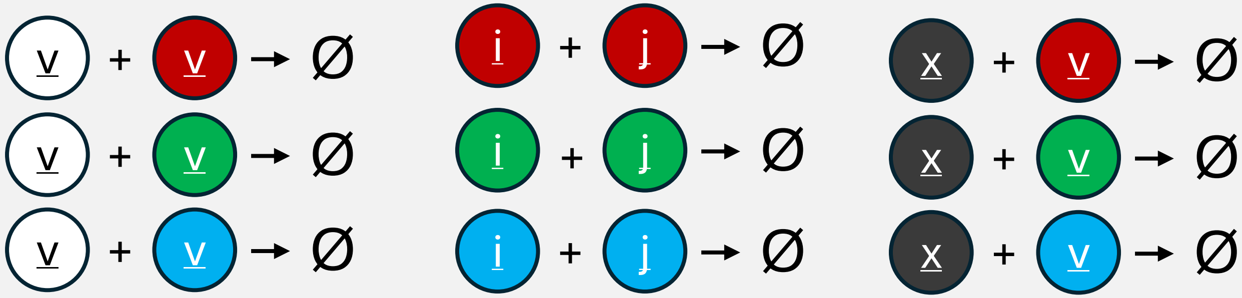
Eliminate **all** of these species

Reactions:

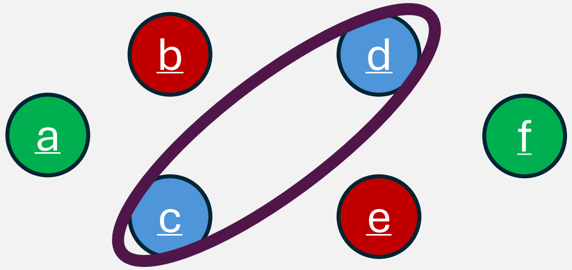
For each vertex v:

For each edge (i,j):

For all vertices V:

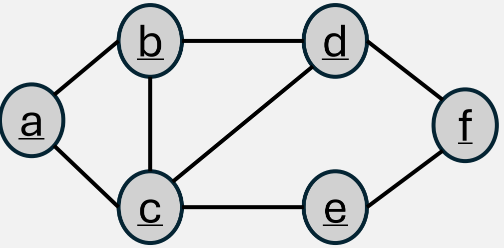


2nd Step Additions: |V| copies of 



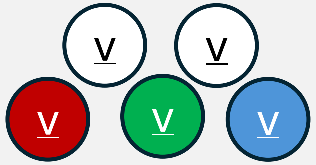
Deletion-Only 2-Step CRNs: (2,0) NP-complete

3-Color Input:



Initial Configuration:

For each vertex v:

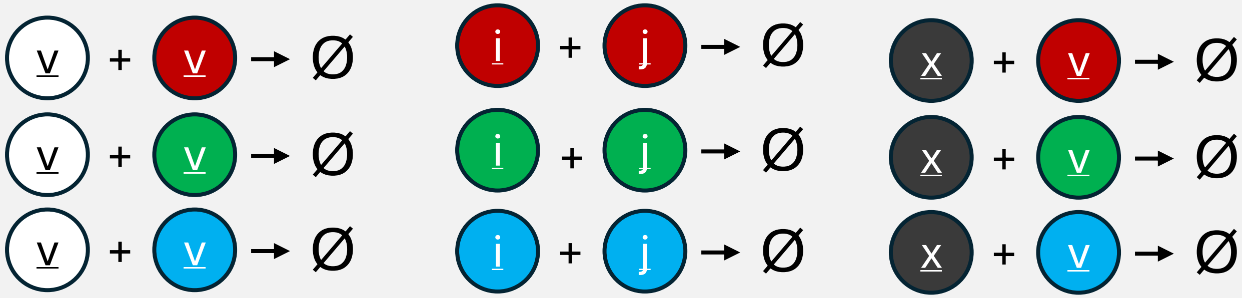


Target Configuration:

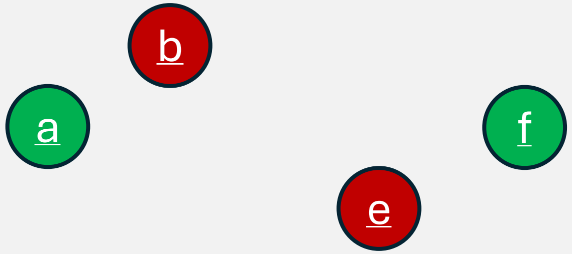
Eliminate **all** of these species

Reactions:

For each vertex v: For each edge (i,j): For all vertices V:

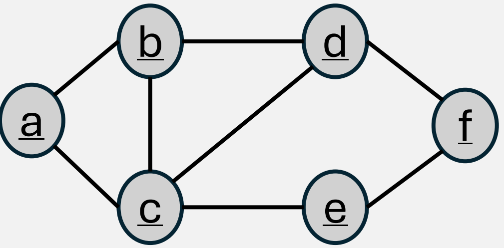


2nd Step Additions: |V| copies of 



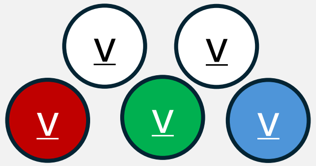
Deletion-Only 2-Step CRNs: (2,0) NP-complete

3-Color Input:



Initial Configuration:

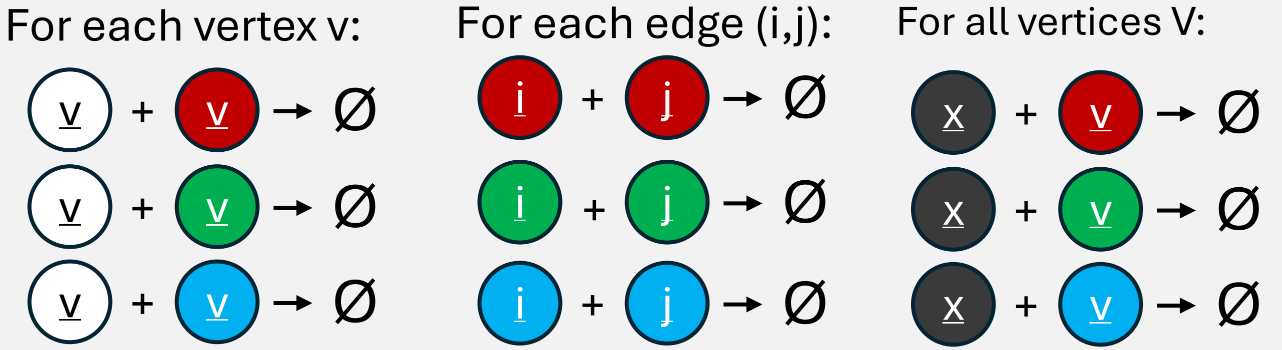
For each vertex v:



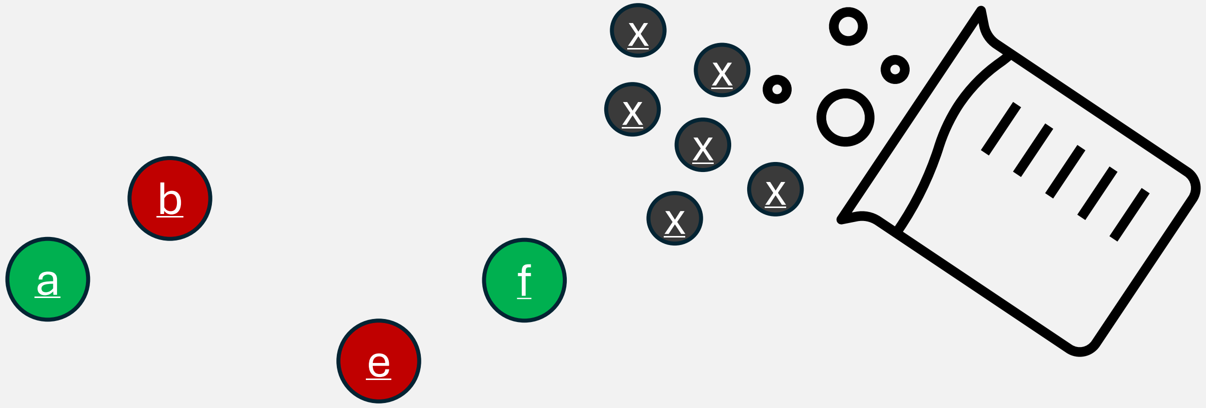
Target Configuration:

Eliminate **all** of these species

Reactions:

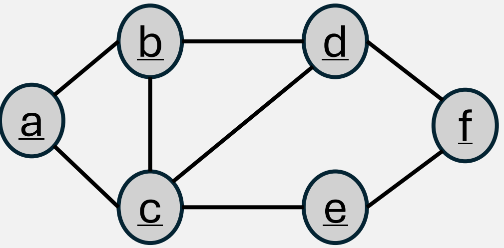


2nd Step Additions: |V| copies of 



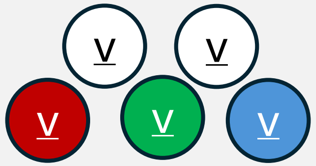
Deletion-Only 2-Step CRNs: (2,0) NP-complete

3-Color Input:



Initial Configuration:

For each vertex v:

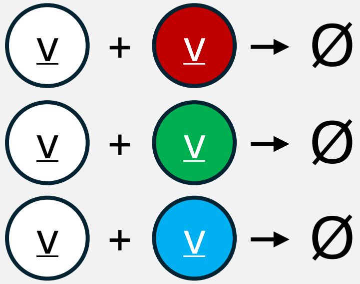


Target Configuration:

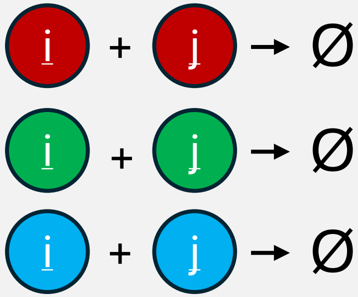
Eliminate **all** of these species

Reactions:

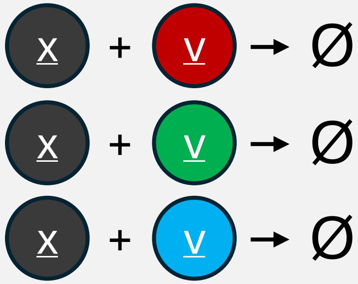
For each vertex v:



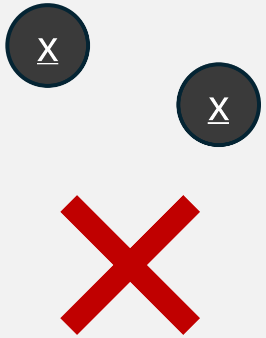
For each edge (i,j):



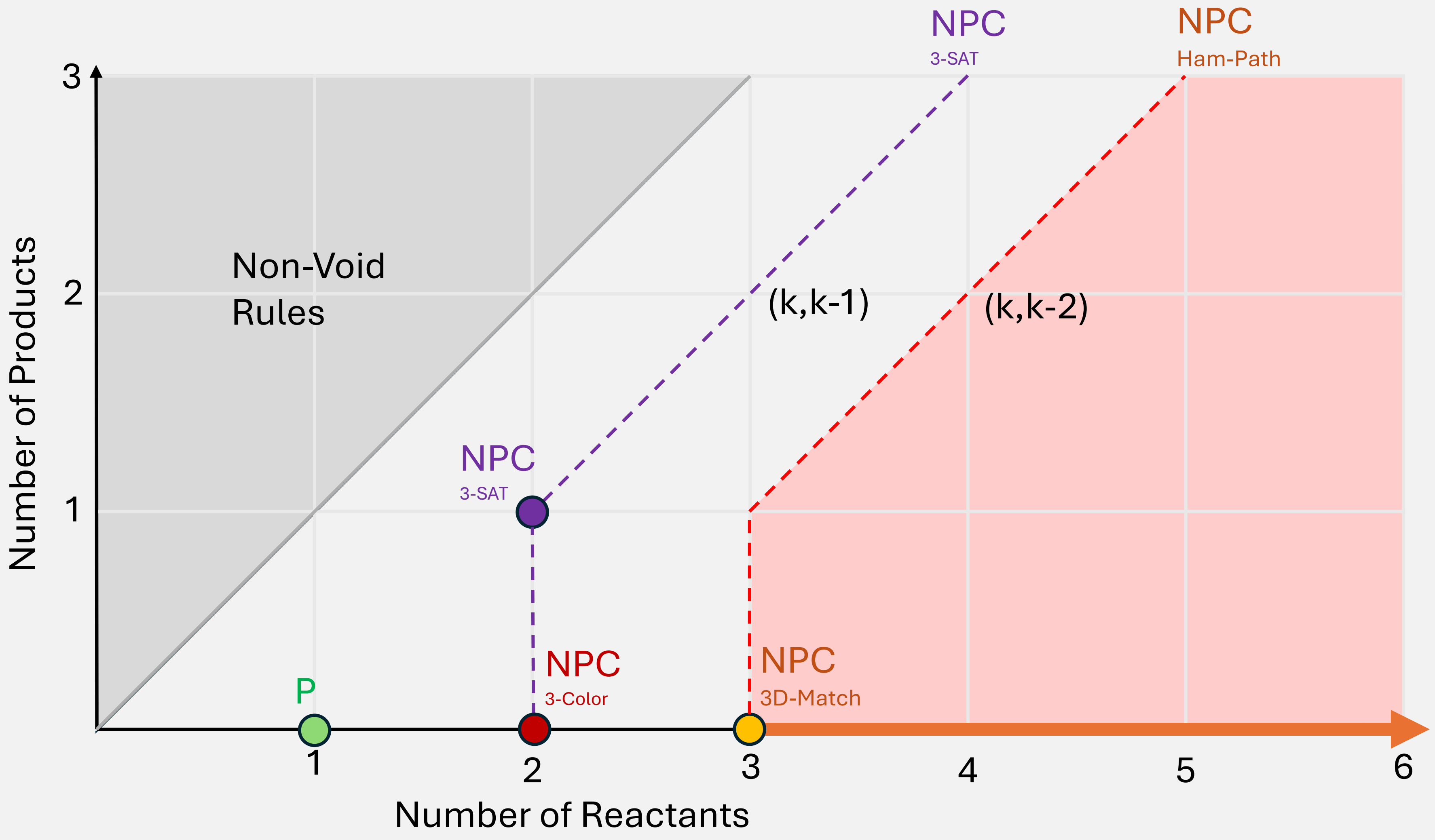
For all vertices V:



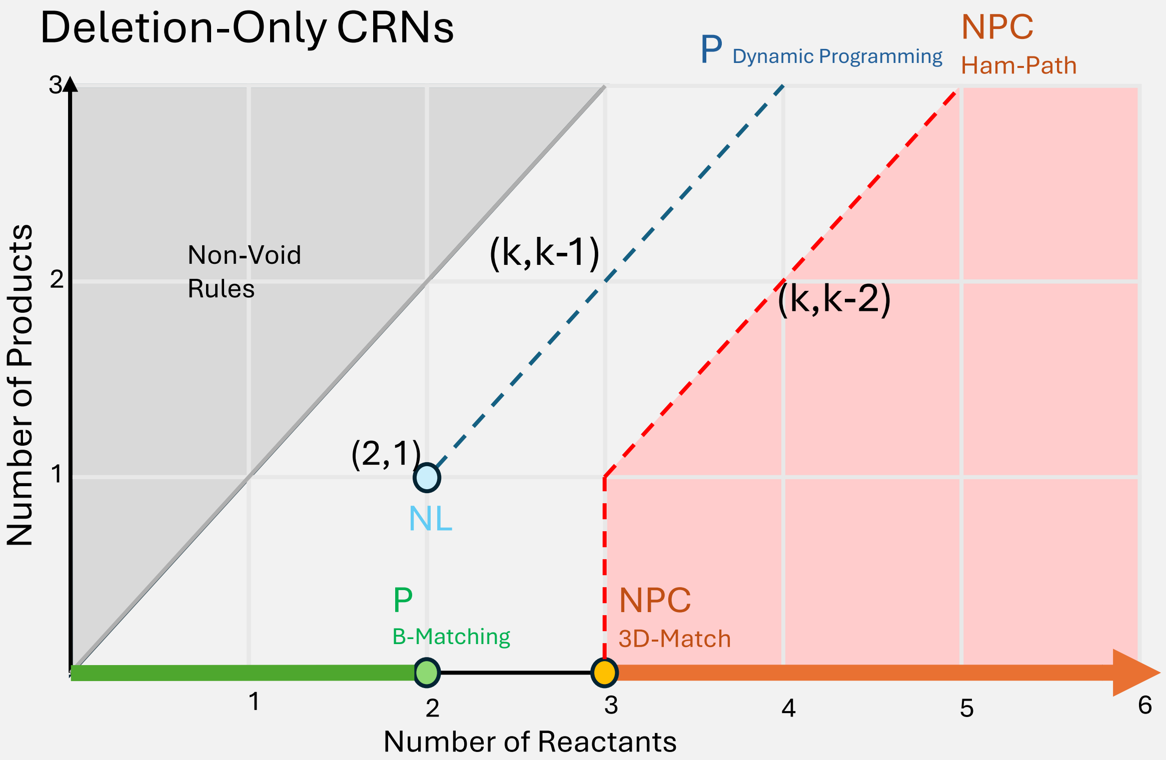
2nd Step Additions: |V| copies of 

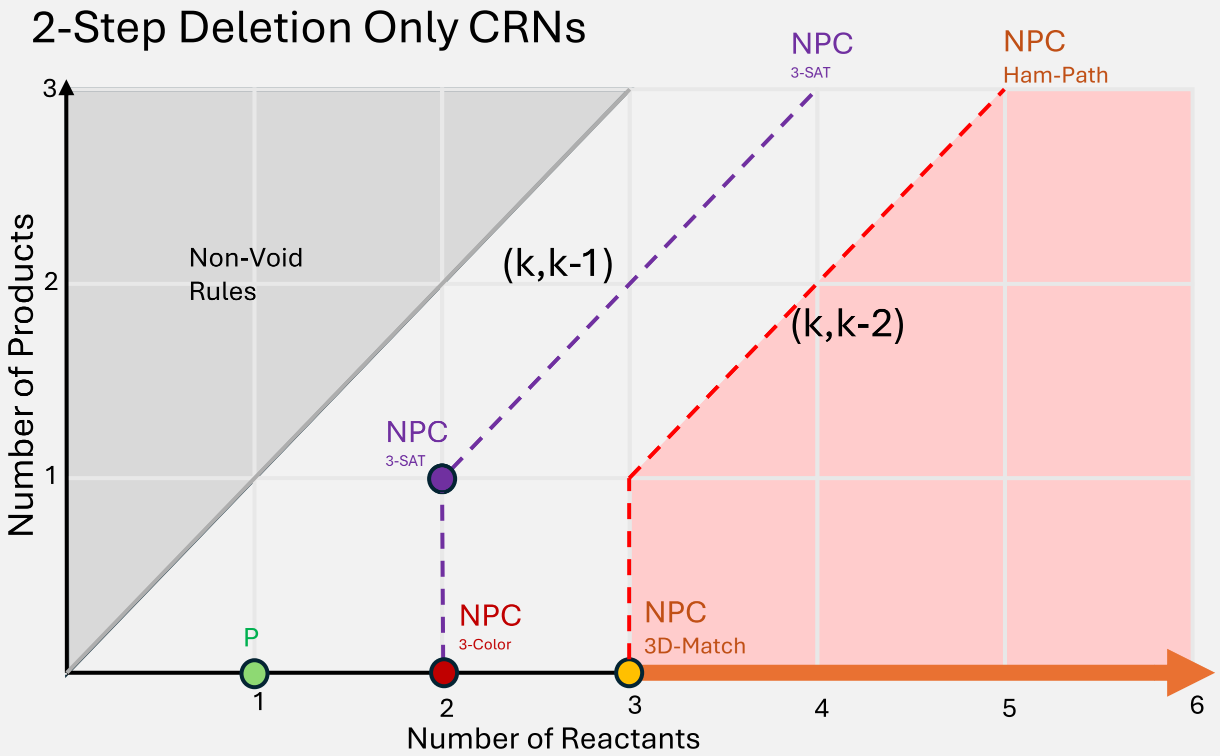


2-Step Deletion-Only CRNs



Thank you.
Questions?





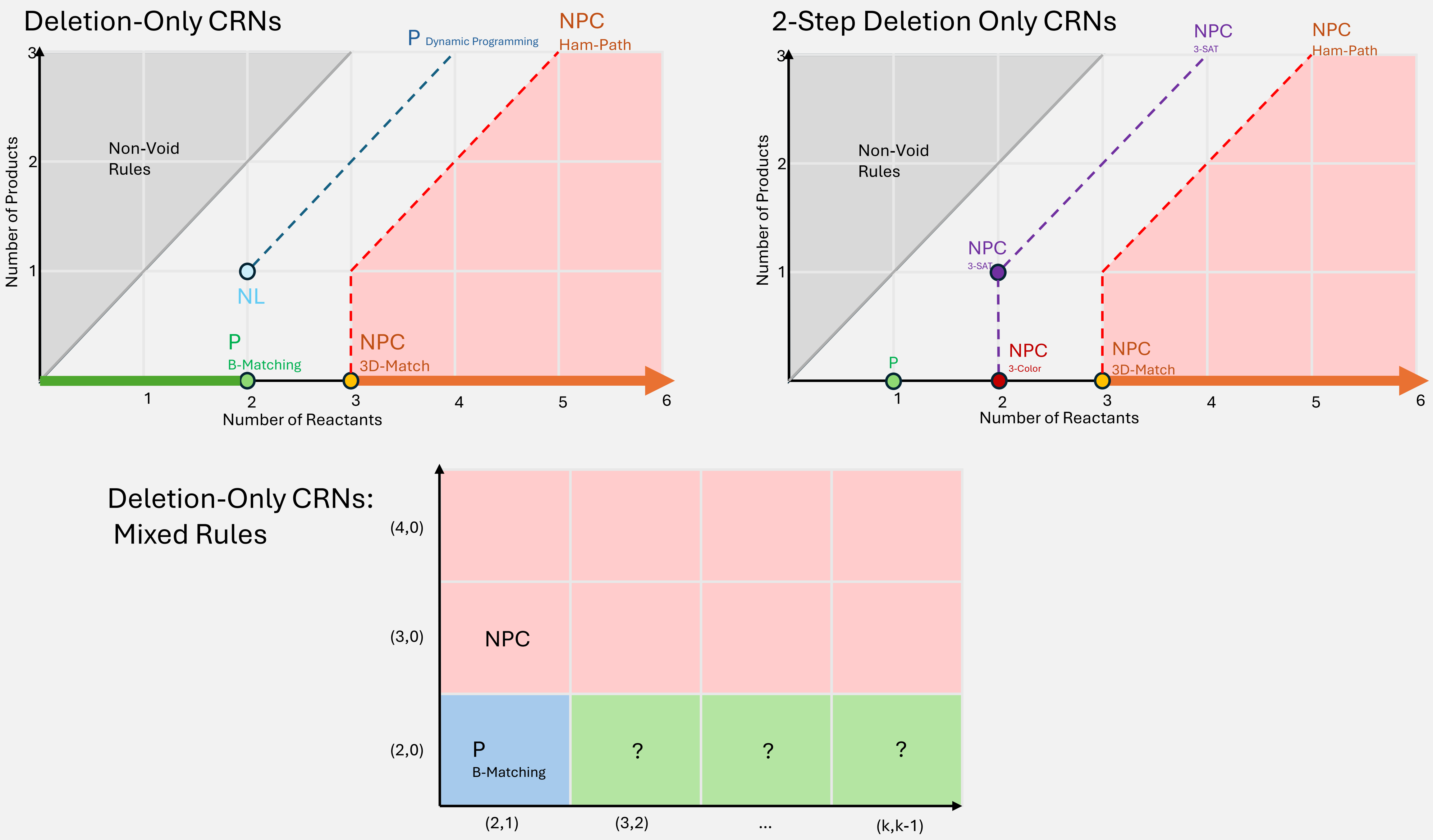
Brief Announcement: Reachability in Deletion-only Chemical Reaction Networks

Bin Fu, Timothy Gomez , Ryan Knobel, Austin Luchsinger, Aiden Massie, Marco Rodriguez, Adrian Salinas, Robert Schweller, Tim Wylie

Full Version: (to appear) The 31st International Conference on DNA Computing and Molecular Programming (DNA31)

Deletion-Only CRNs: Mixed Rules

	(2,1)	(3,2)	…	(k,k-1)
(4,0)	NPC	NPC	NPC	NPC
(3,0)	NPC	NPC	NPC	NPC
(2,0)	P B-Matching	?	?	?


Deletion-Only CRNs
P Dynamic Programming
NPC
Ham-Path
Non-Void
Rules
Number of Products
NL
P
B-Matching
NPC
3D-Match
Number of Reactants
2-Step Deletion Only CRNs
NPC
3-SAT
NPC
Ham-Path
Non-Void
Rules
Number of Products
NPC
3-SAT
P
NPC
3-Color
NPC
3D-Match
Number of Reactants
Deletion-Only CRNs:
Mixed Rules
(4,0)
(3,0)
(2,0)
NPC
P
B-Matching
?
?
?
(2,1)
(3,2)
...
(k,k-1)