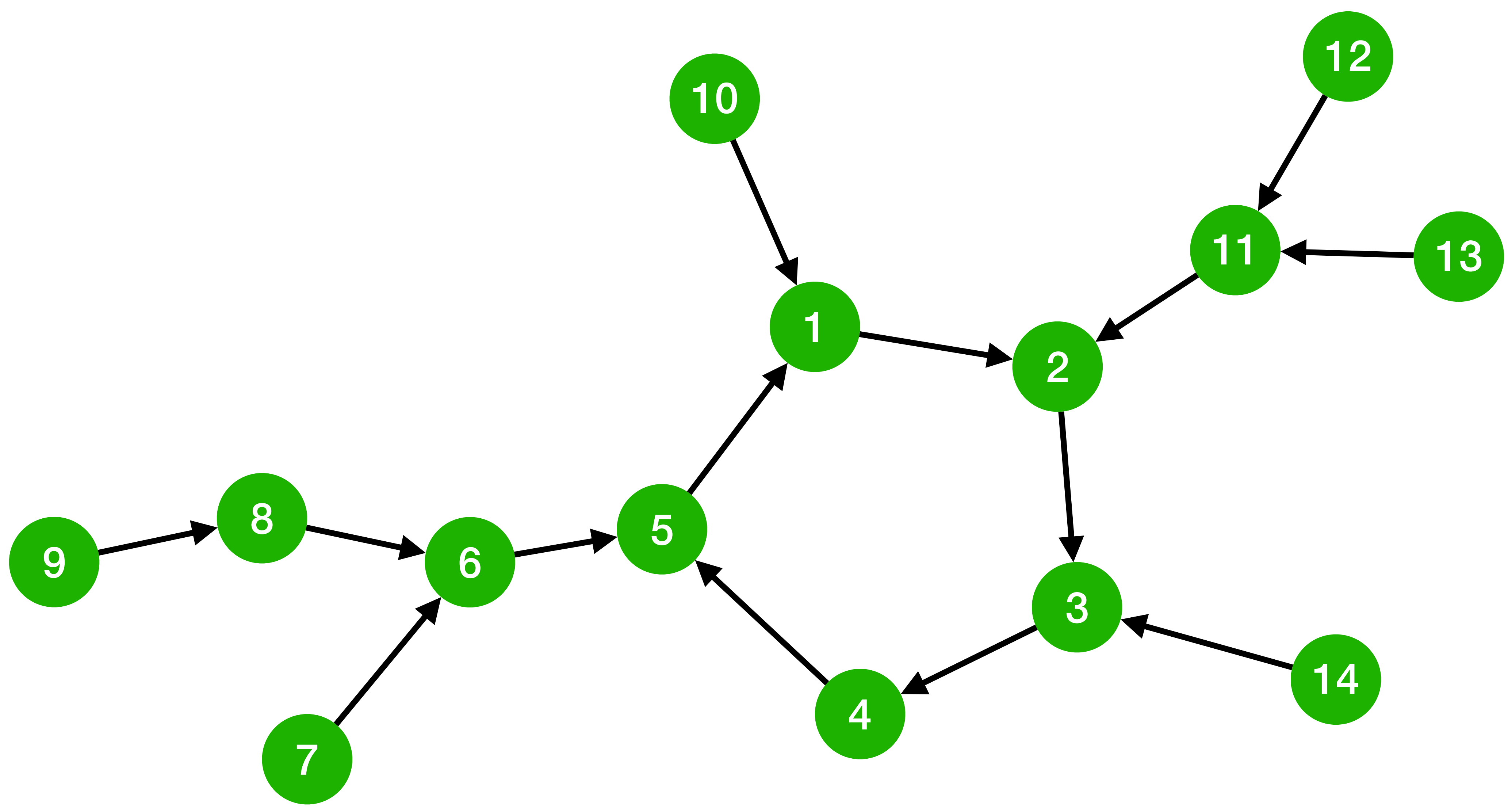
Analysis of discrete dynamical systems

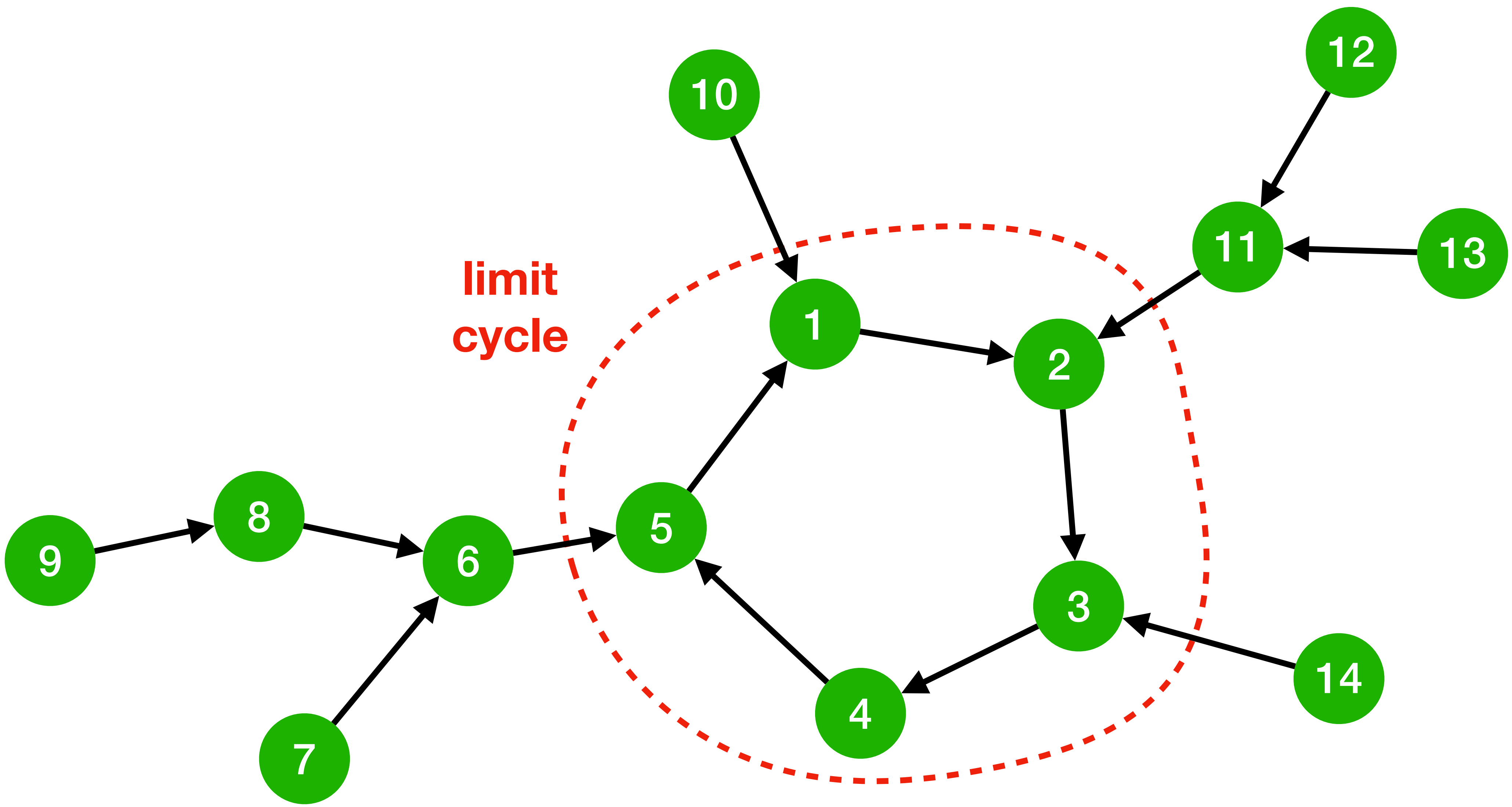
An algebraic perspective

Antonio E. Porreca • aeporreca.org
Aix-Marseille Université & LIS 🇫🇷

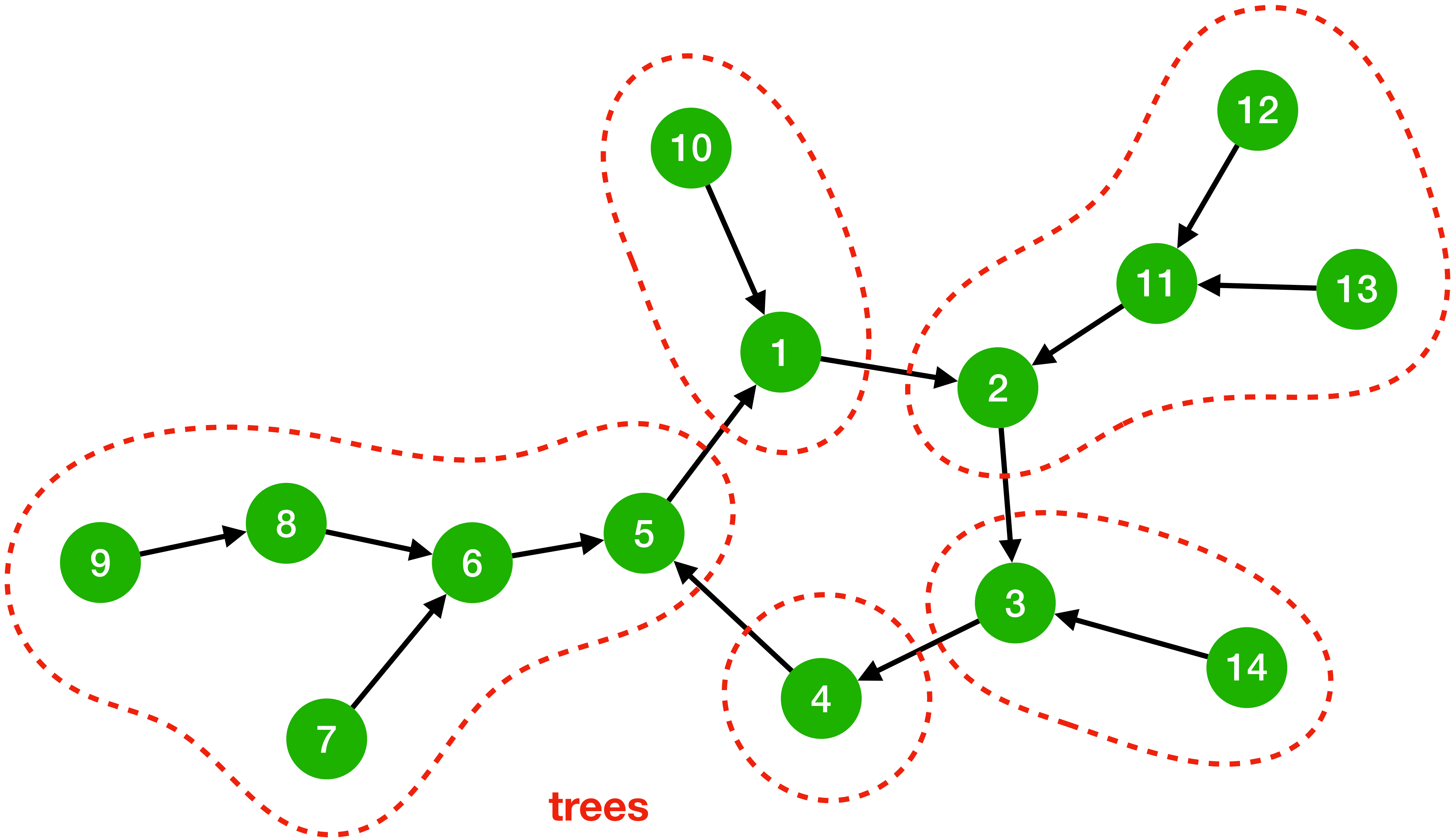
Discrete (finite, deterministic) dynamical systems



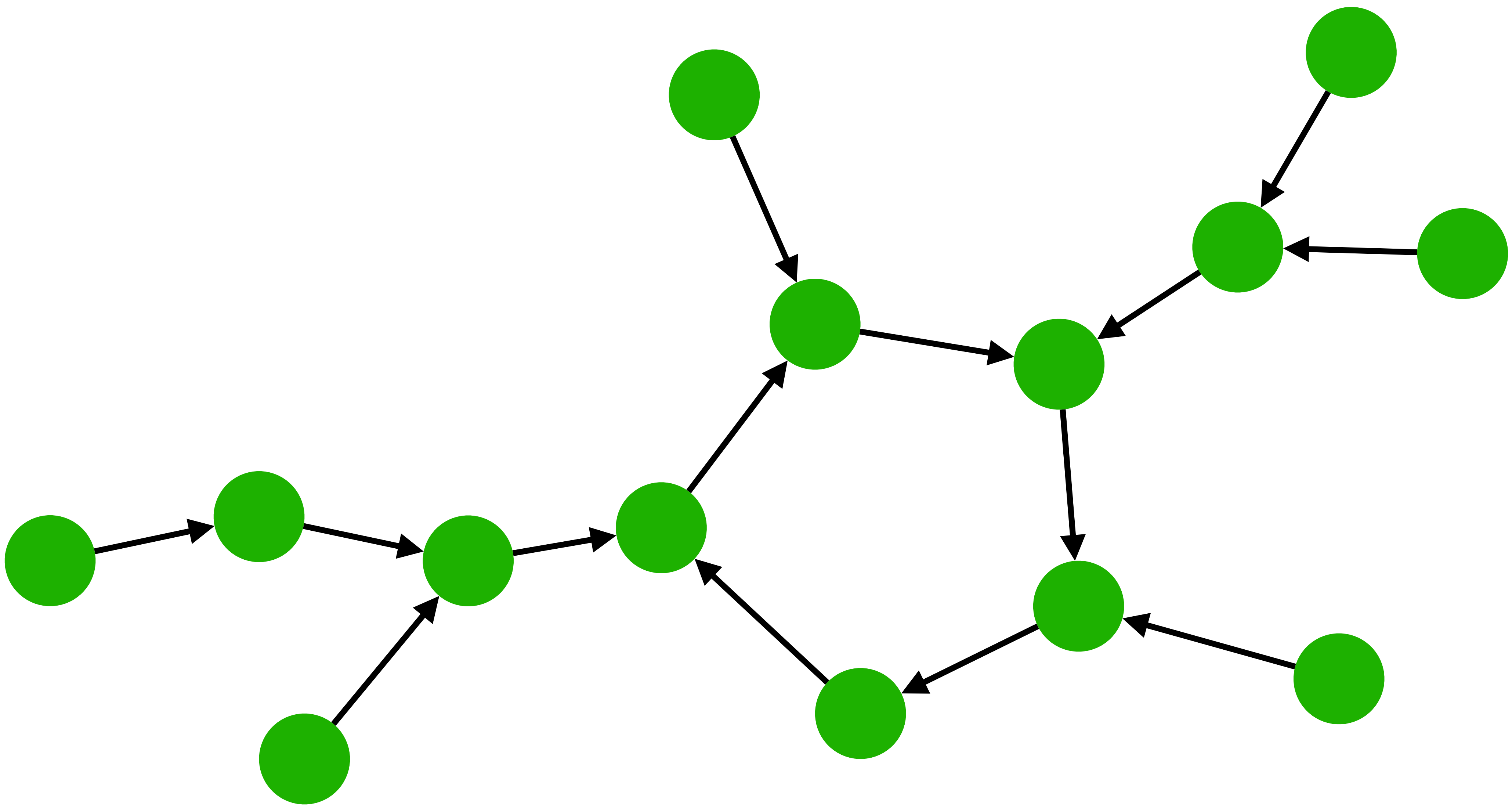
Discrete (finite, deterministic) dynamical systems



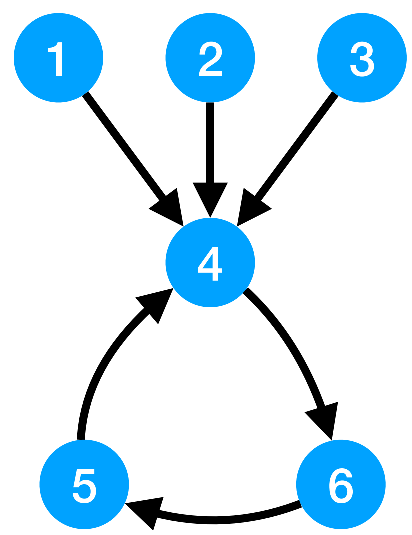
Discrete (finite, deterministic) dynamical systems

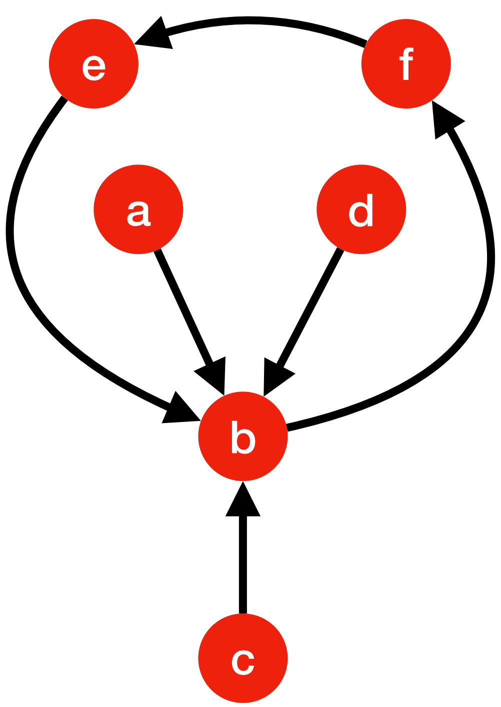


Discrete (finite, deterministic) dynamical systems up to isomorphisms

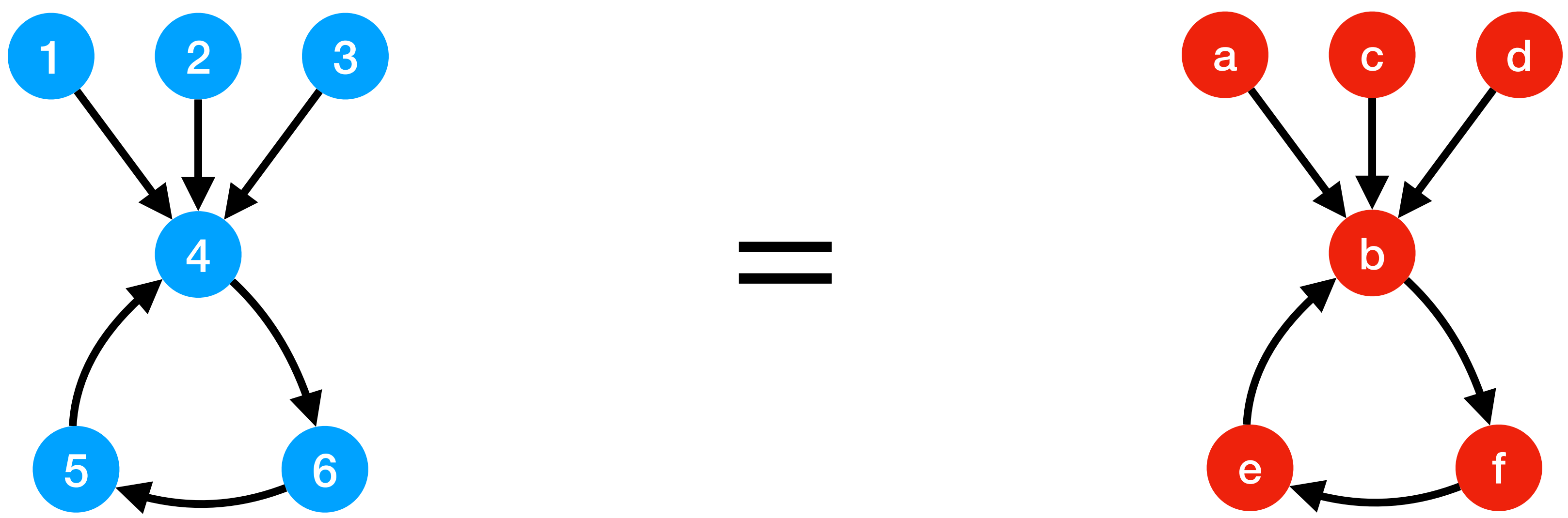


Discrete (finite, deterministic) dynamical systems up to isomorphisms



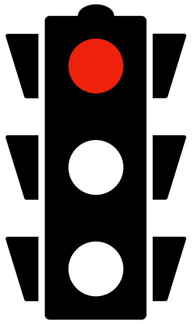


Discrete (finite, deterministic) dynamical systems up to isomorphisms

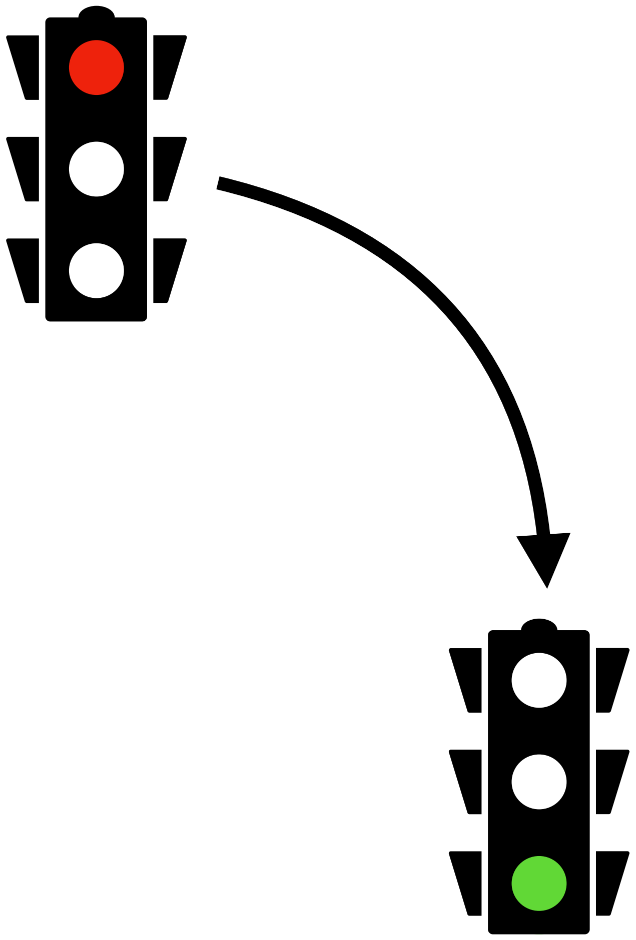


An example from engineering

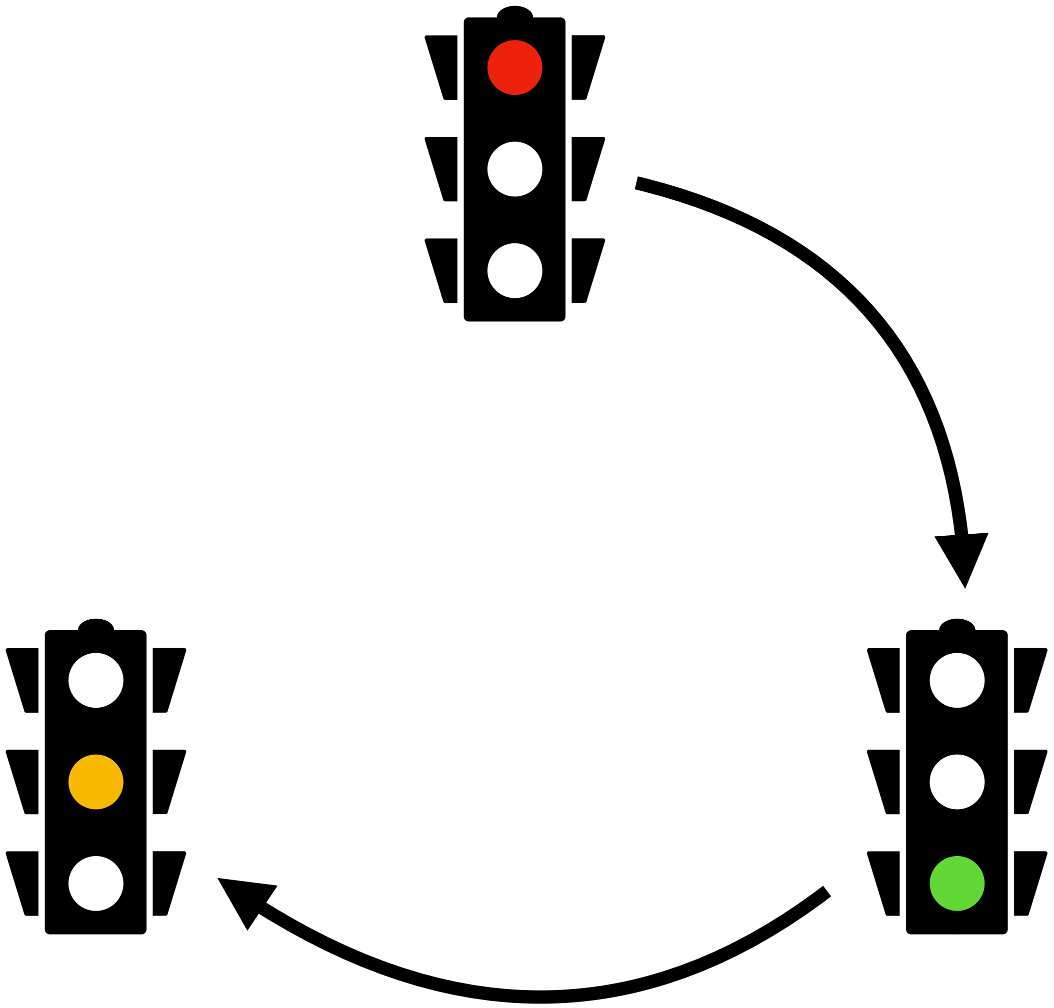
Traffic lights



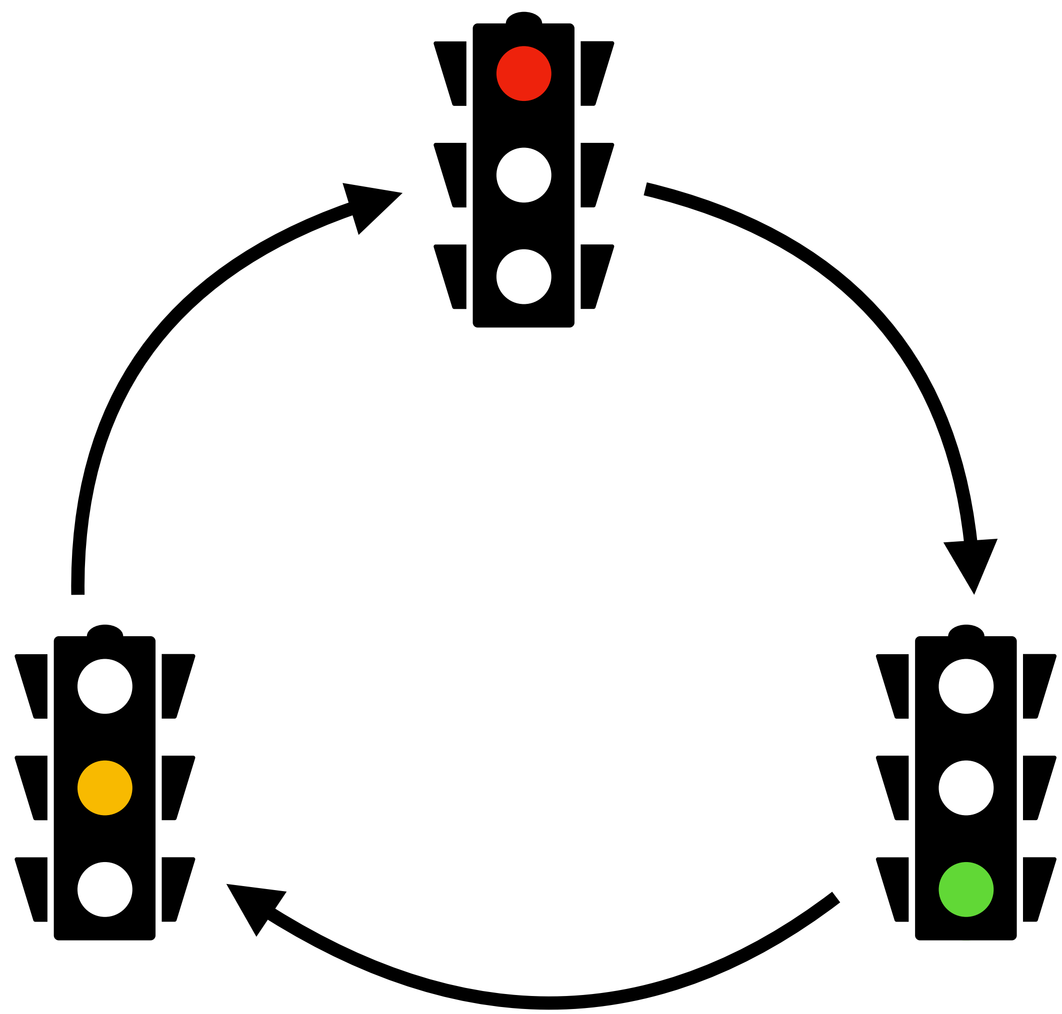
Traffic lights



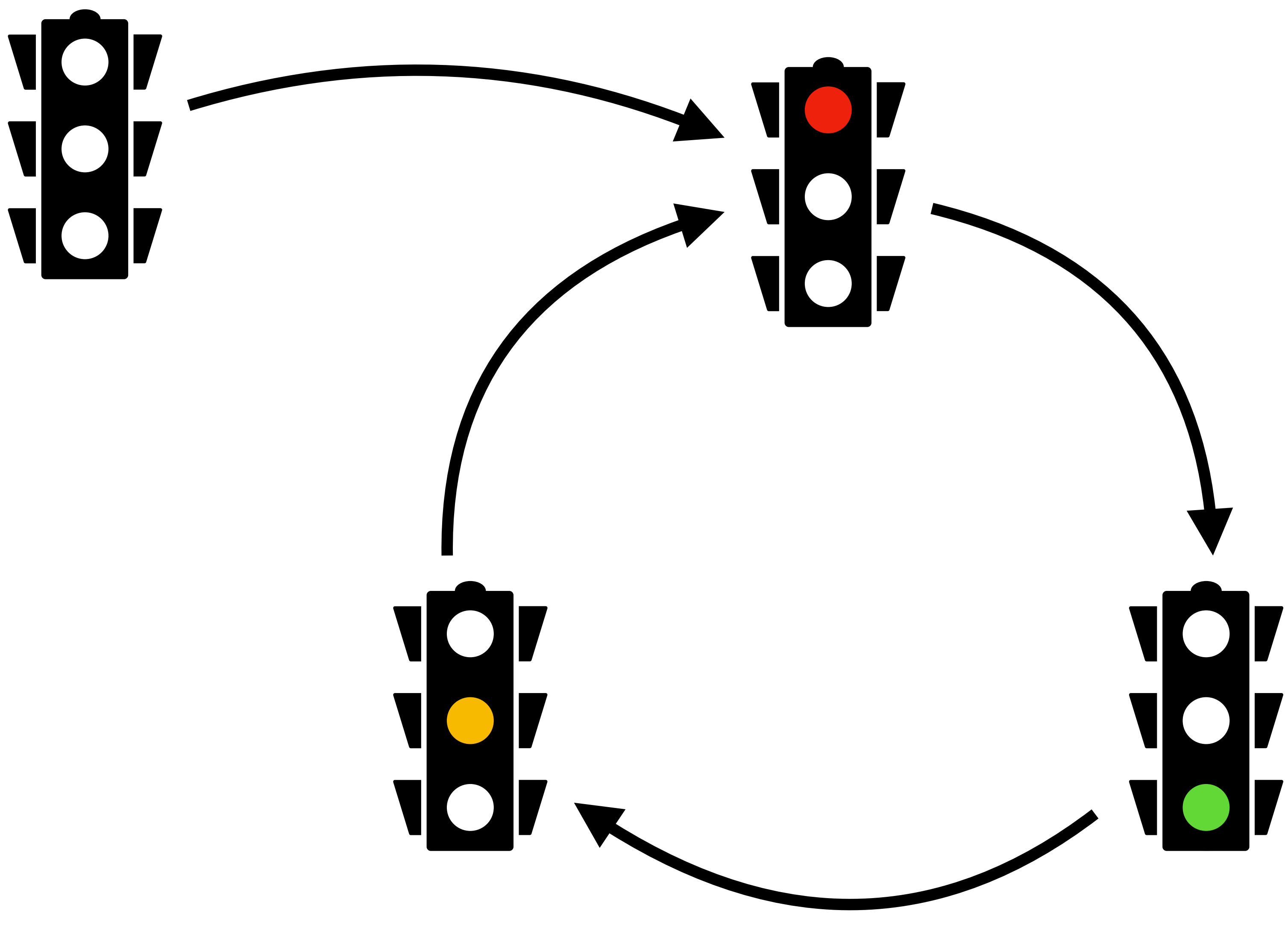
Traffic lights



Traffic lights

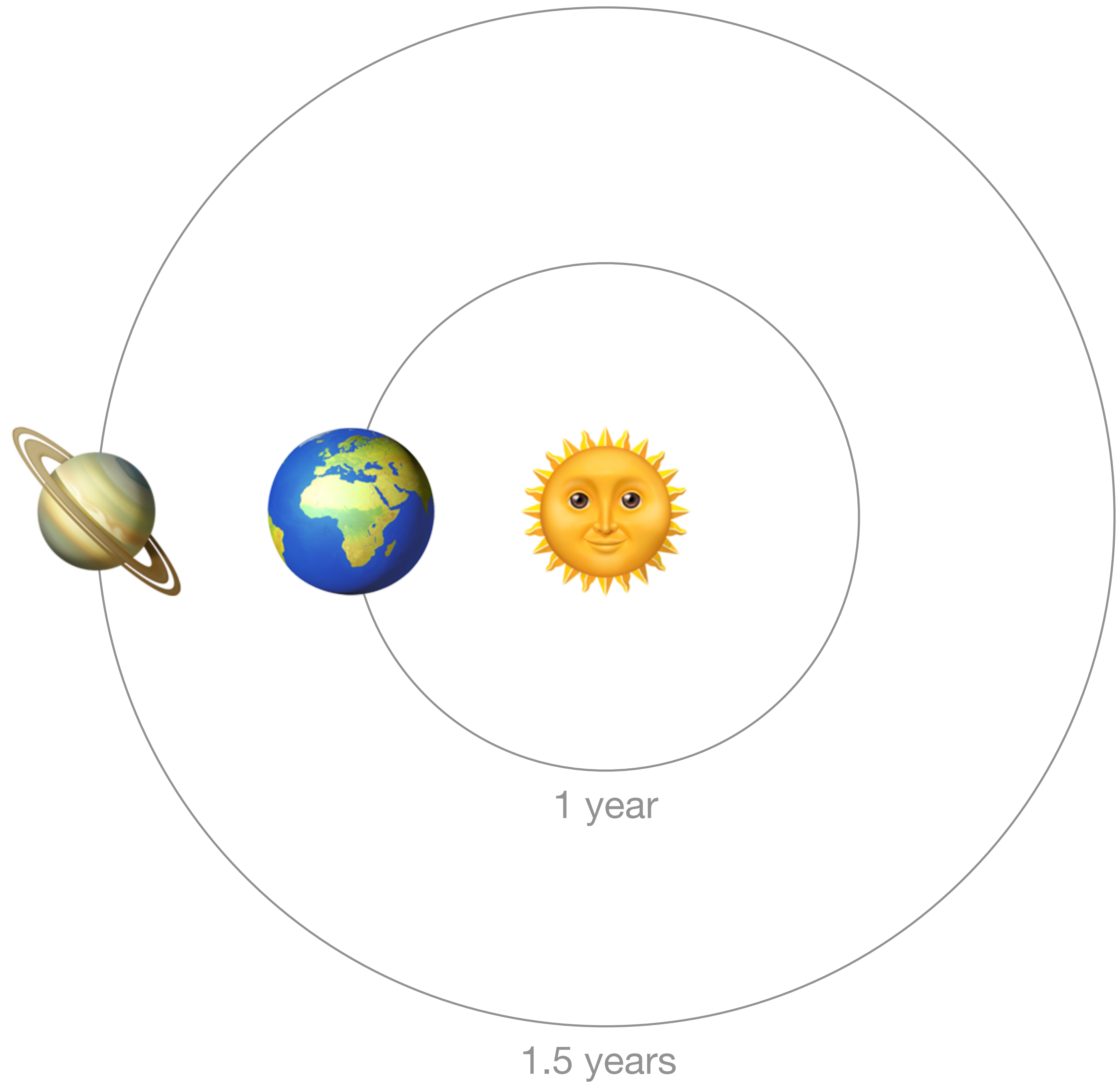


Traffic lights

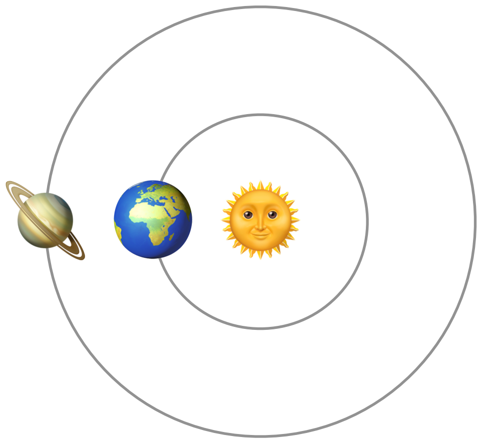


An example from science

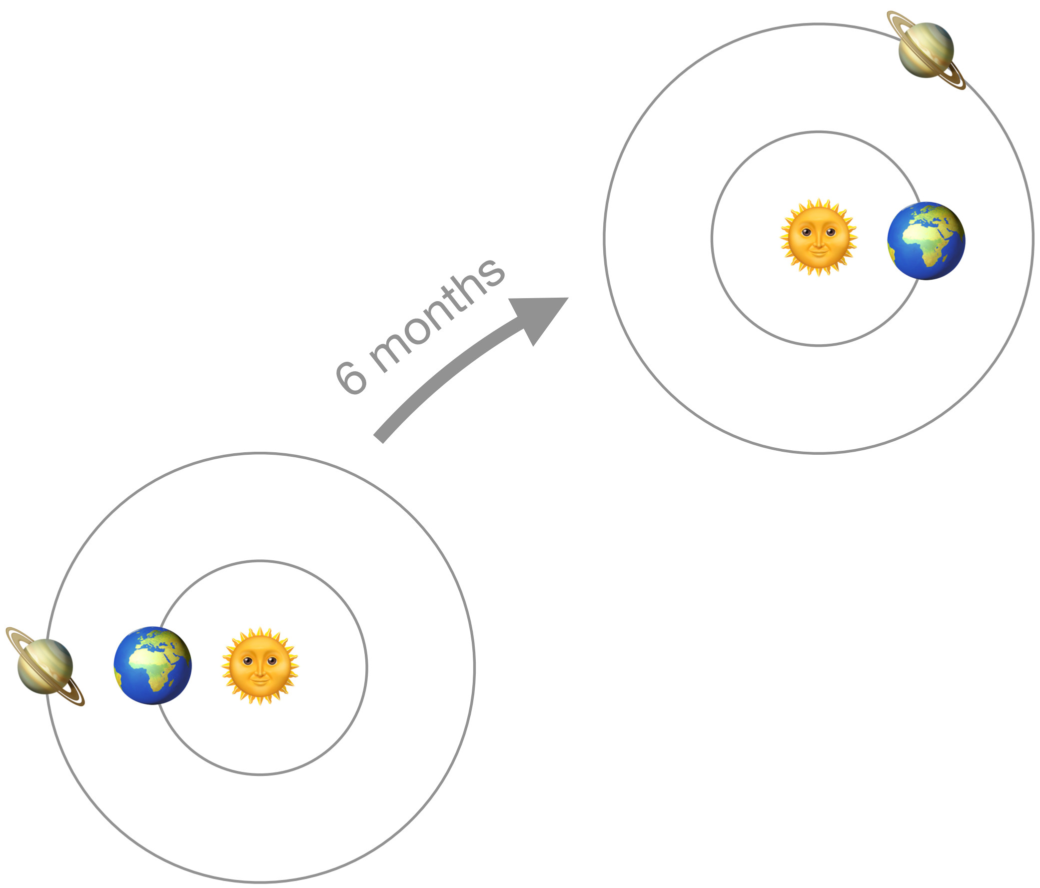
A planetary system



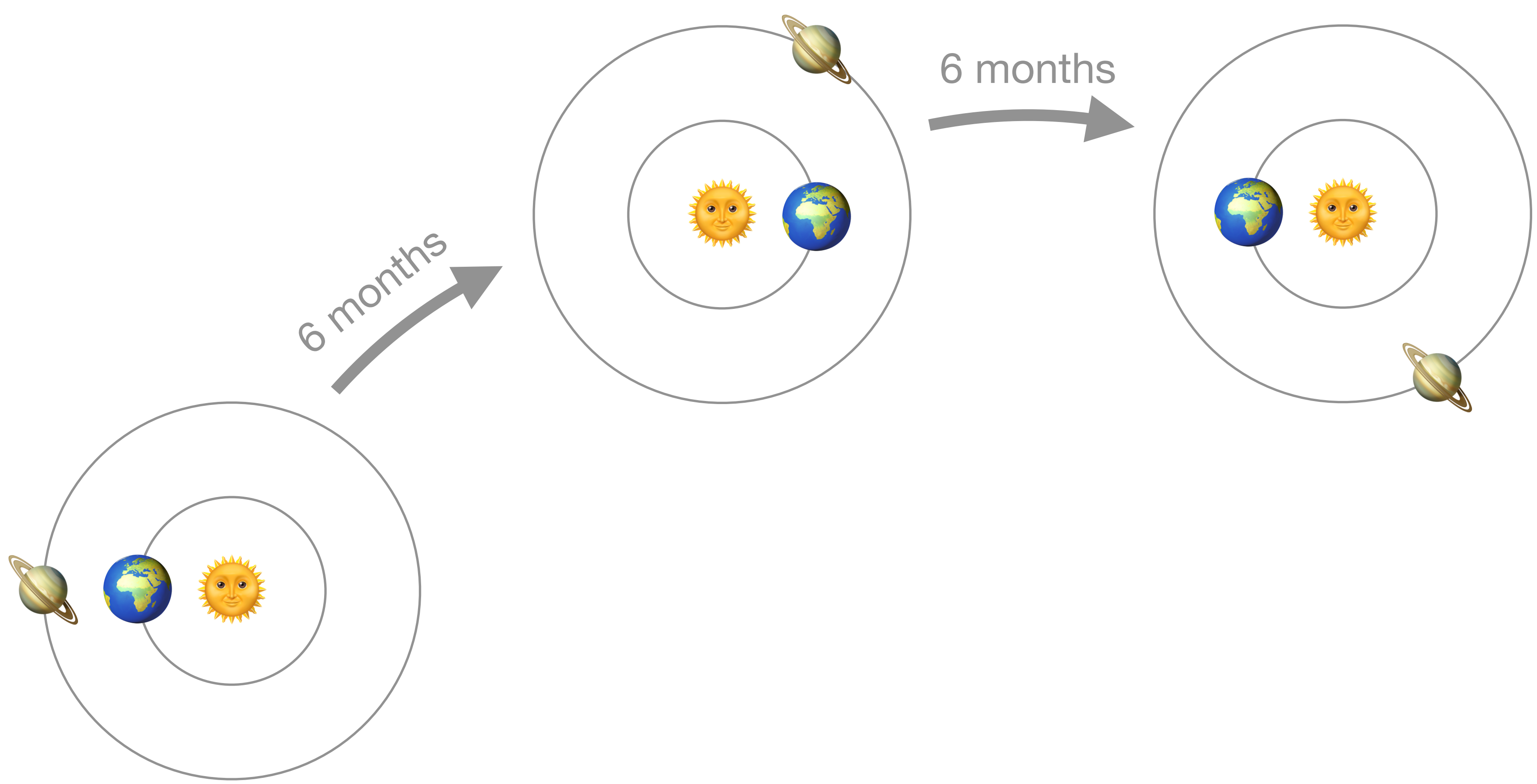
Evolution in time



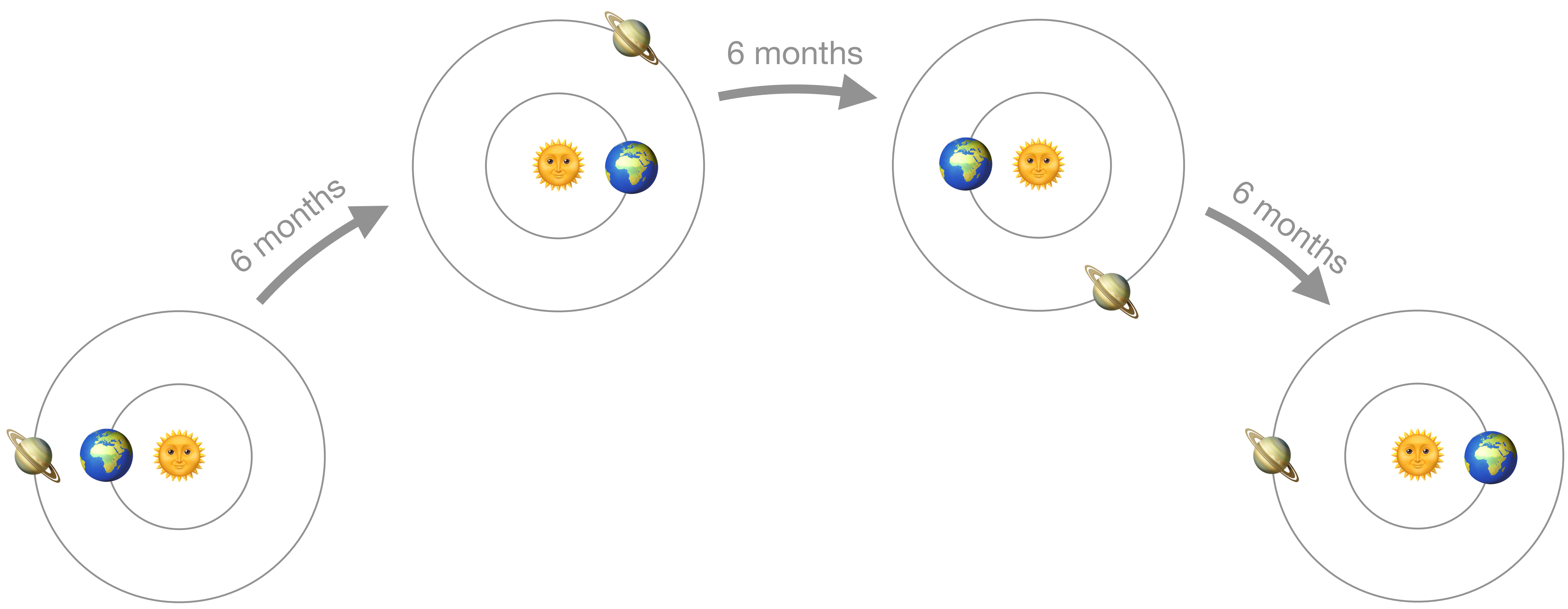
Evolution in time



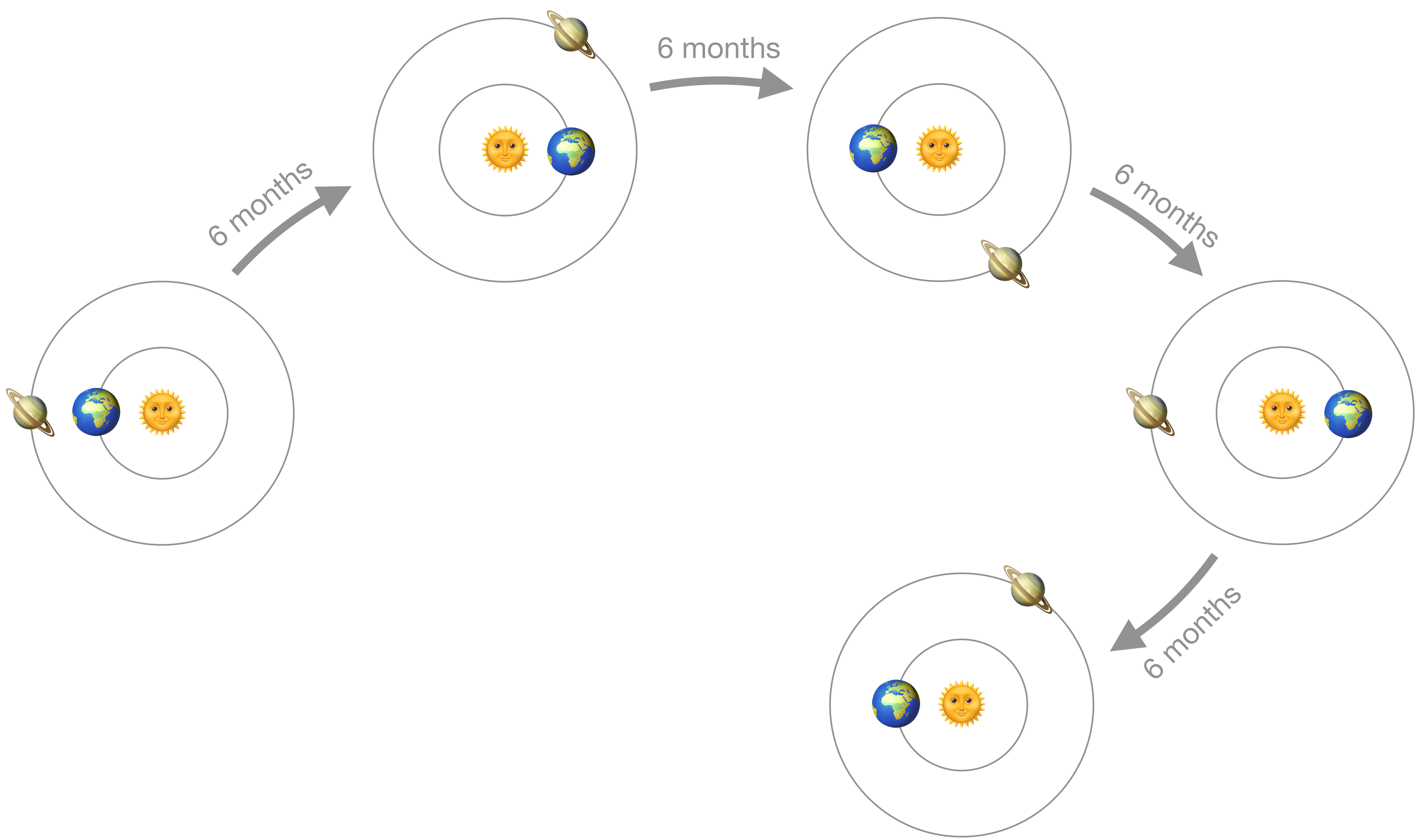
Evolution in time



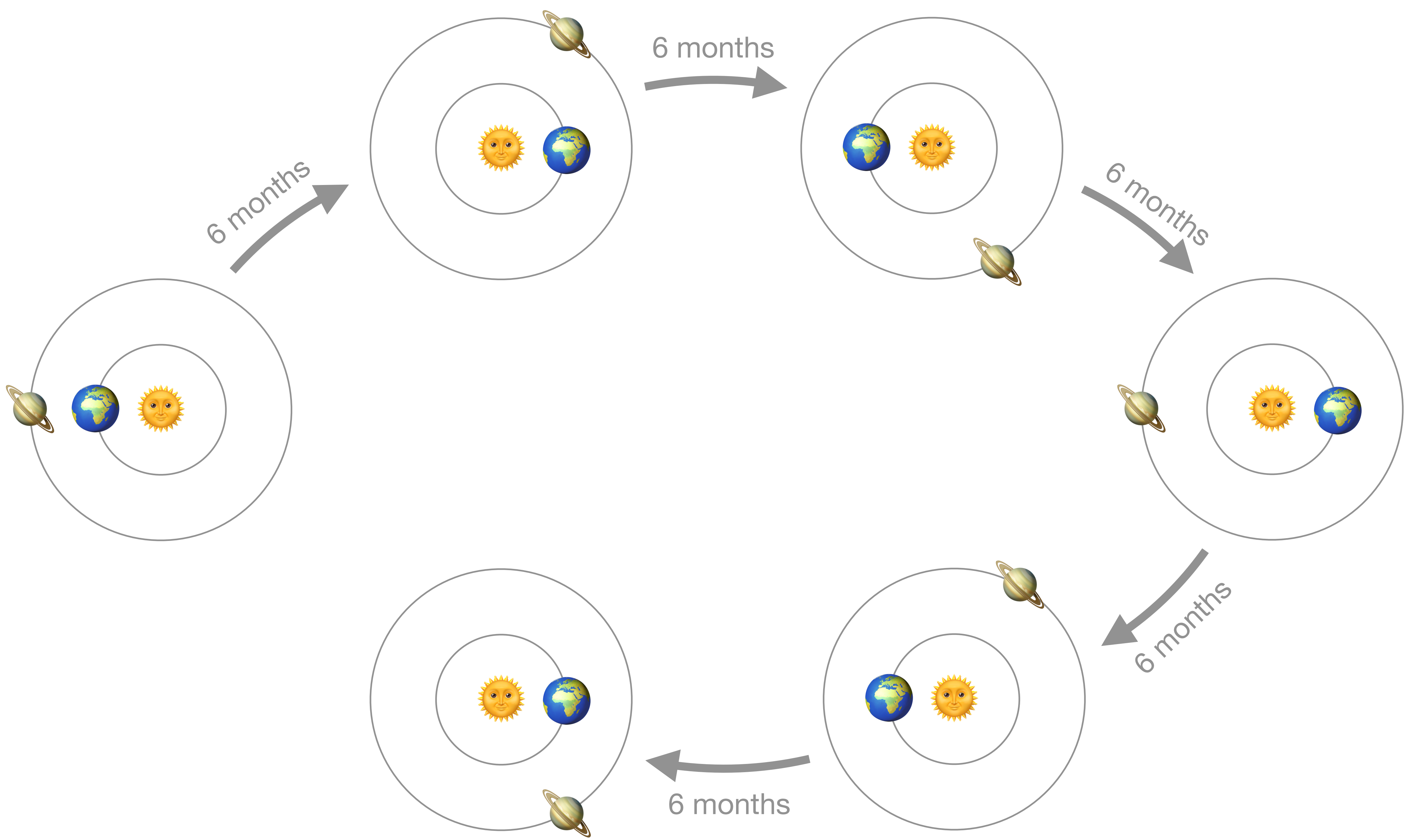
Evolution in time



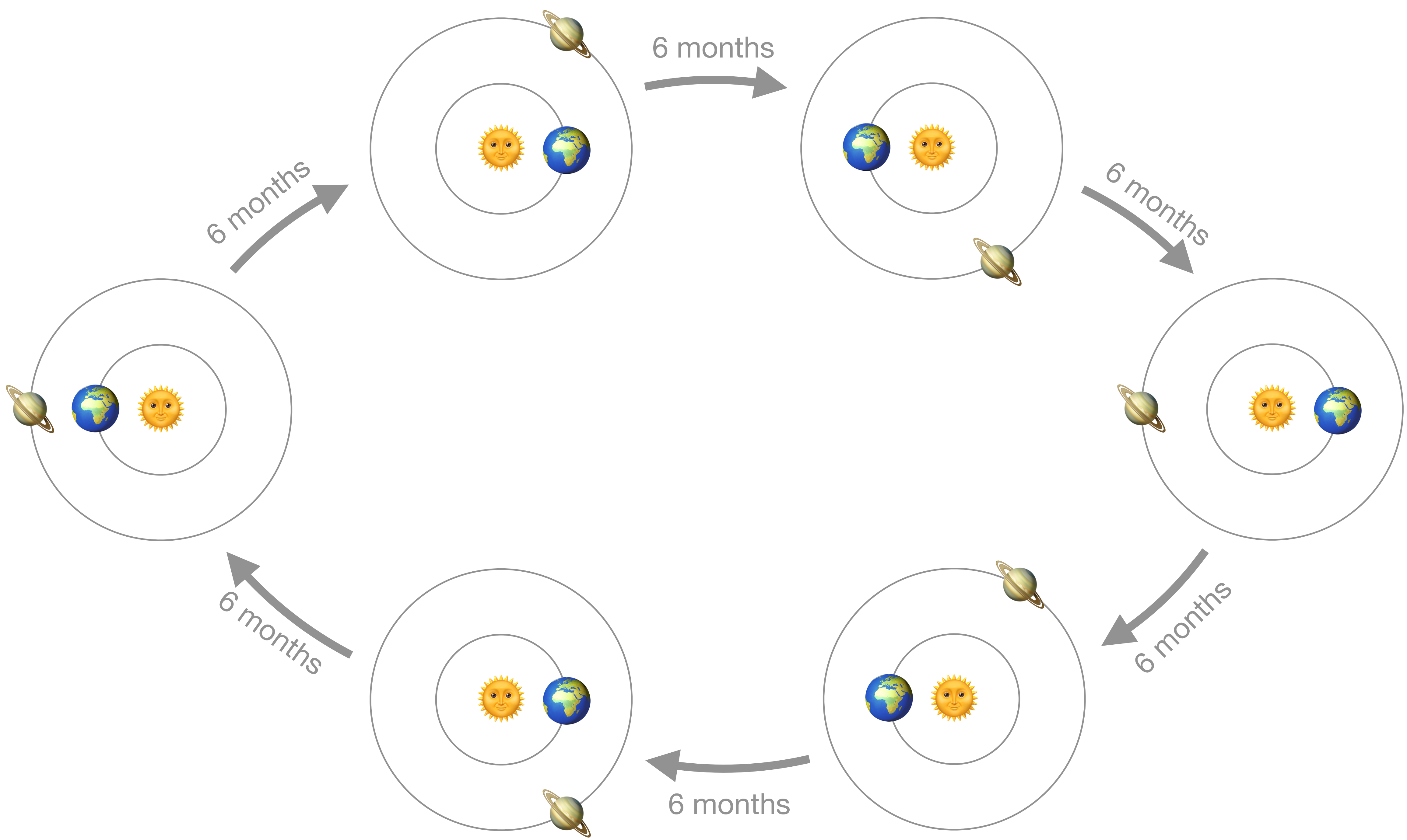
Evolution in time



Evolution in time

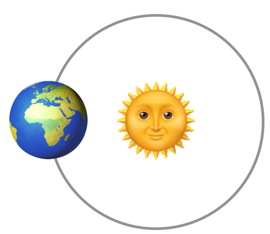


Evolution in time

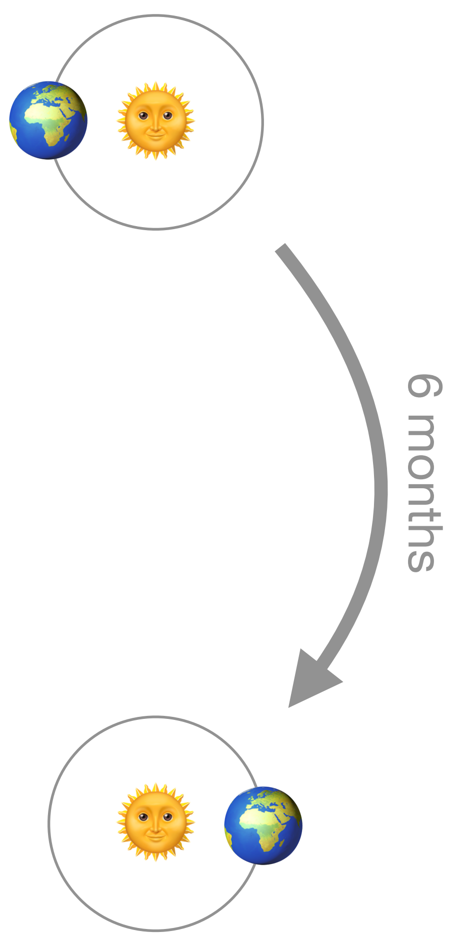


Decomposing the system

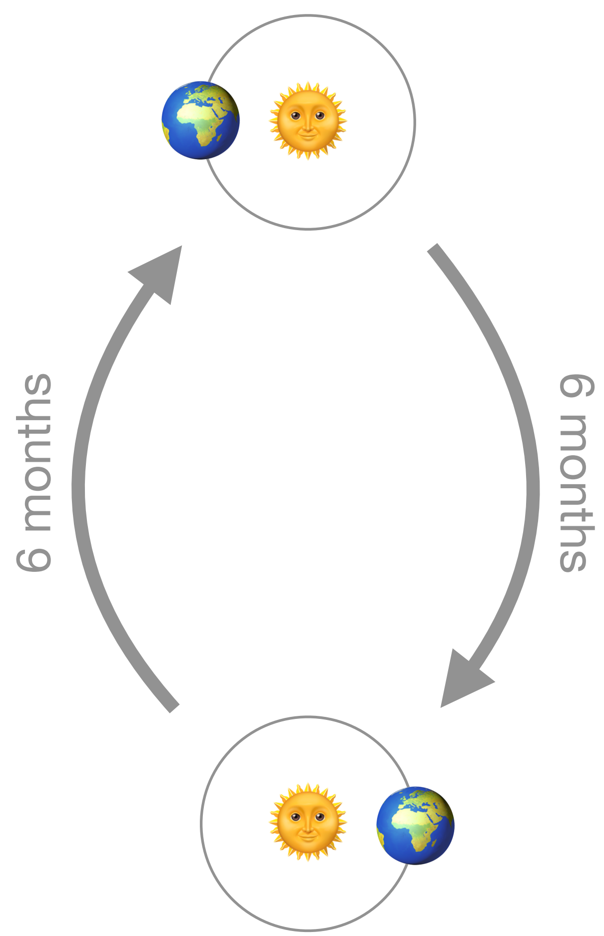
Decomposing the system



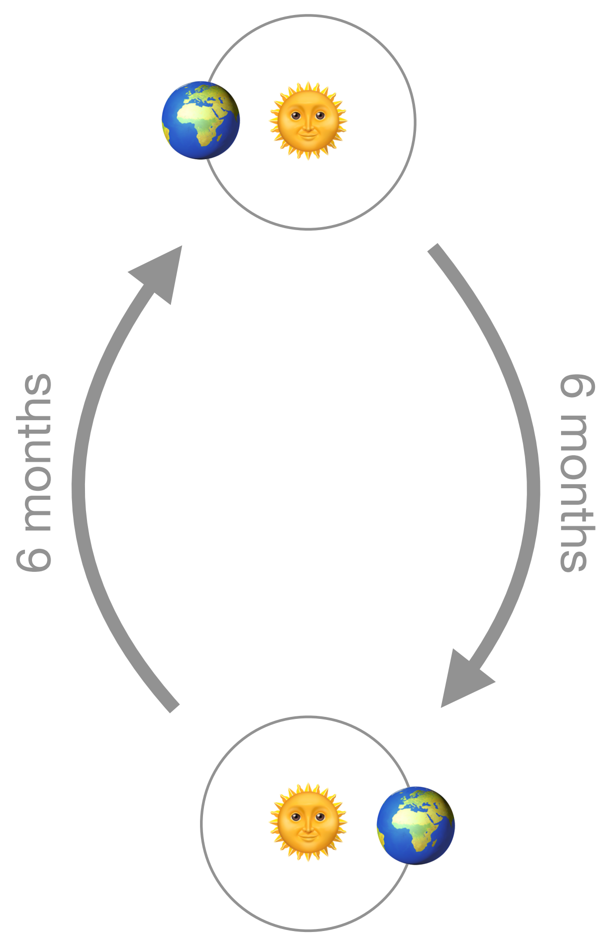
Decomposing the system

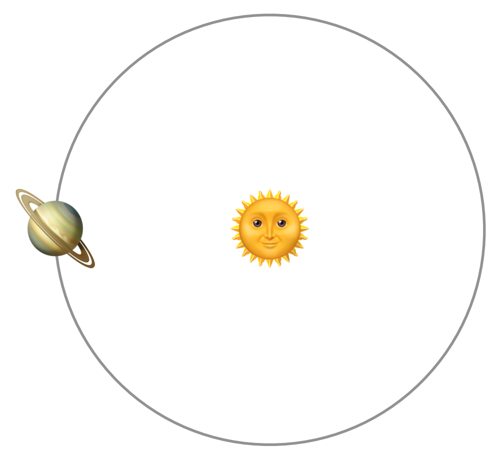


Decomposing the system

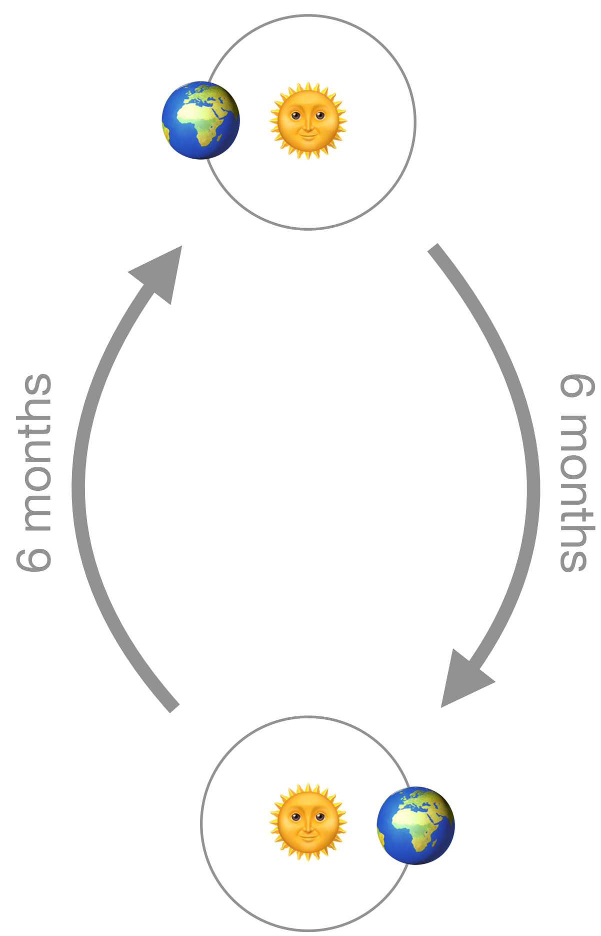


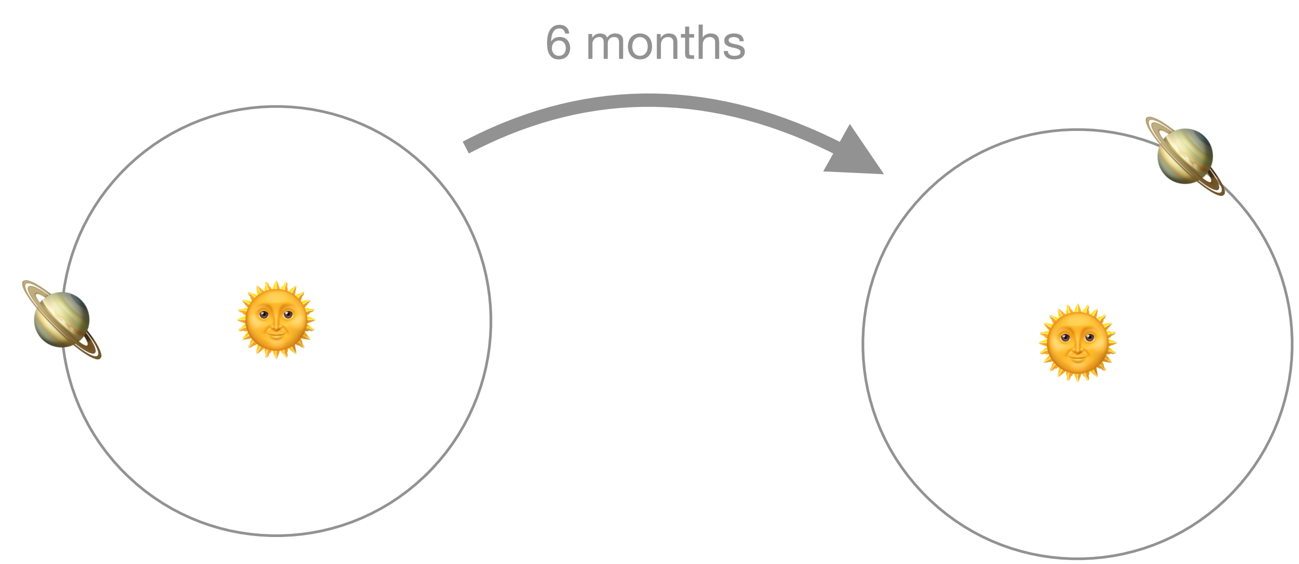
Decomposing the system



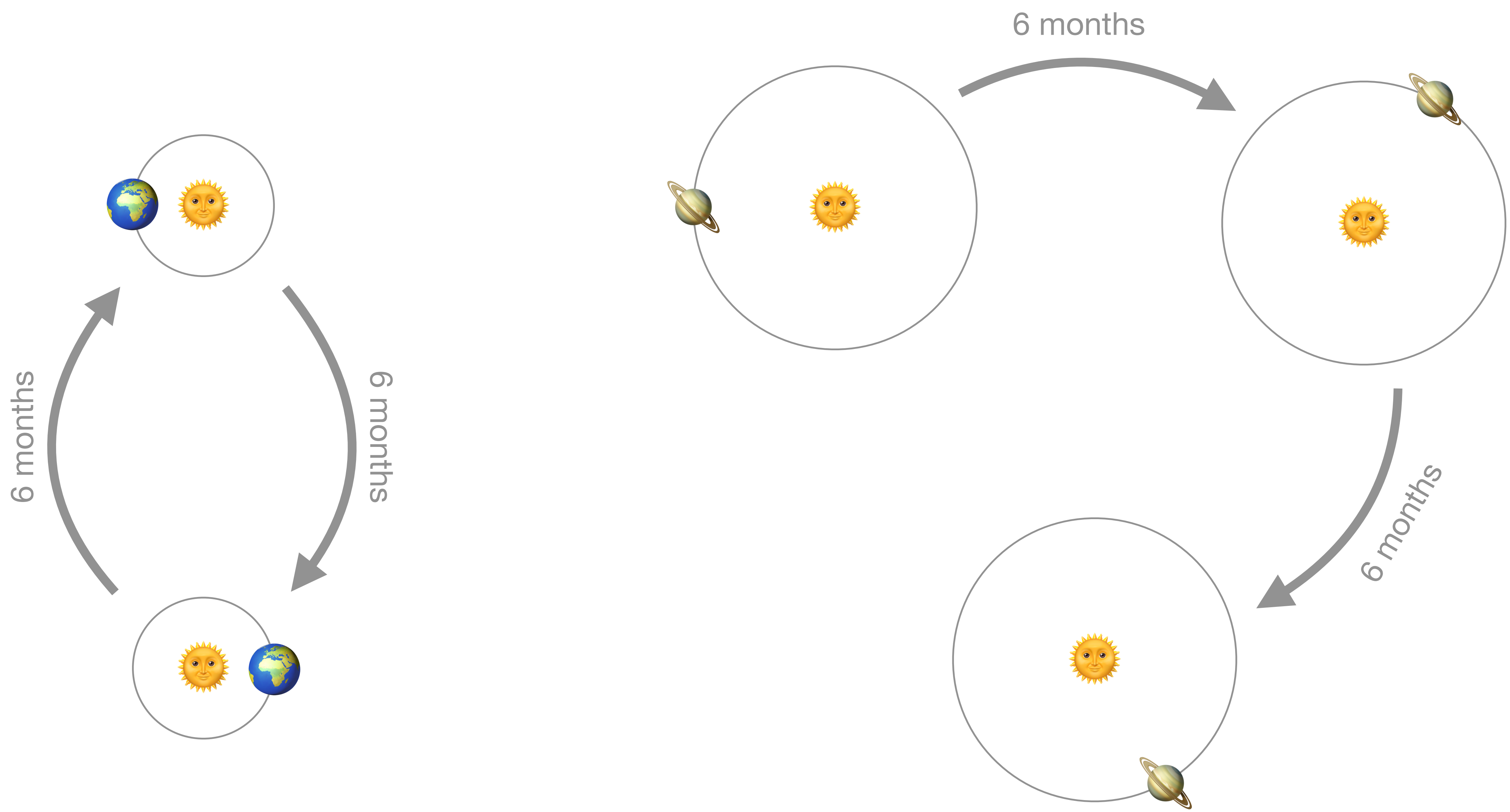


Decomposing the system

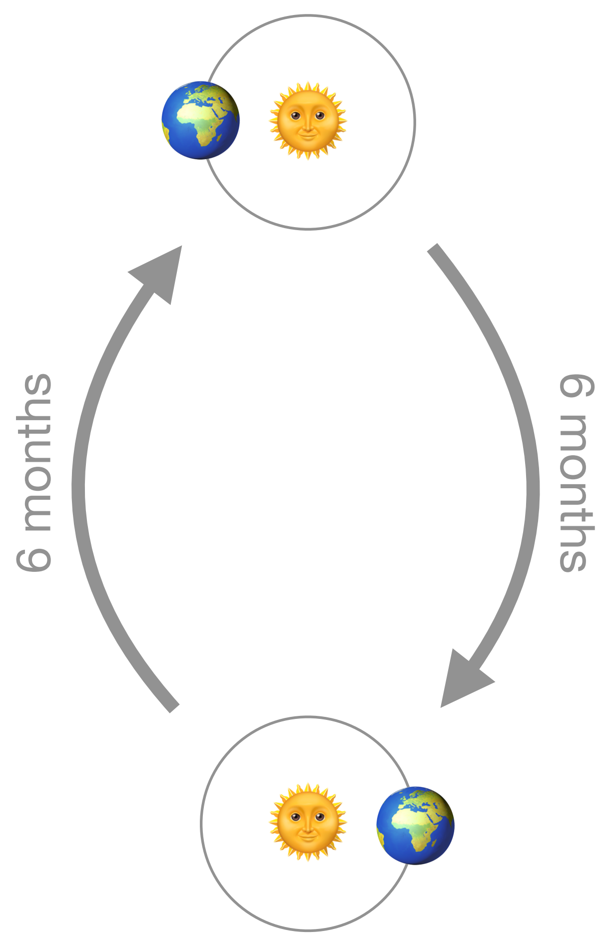


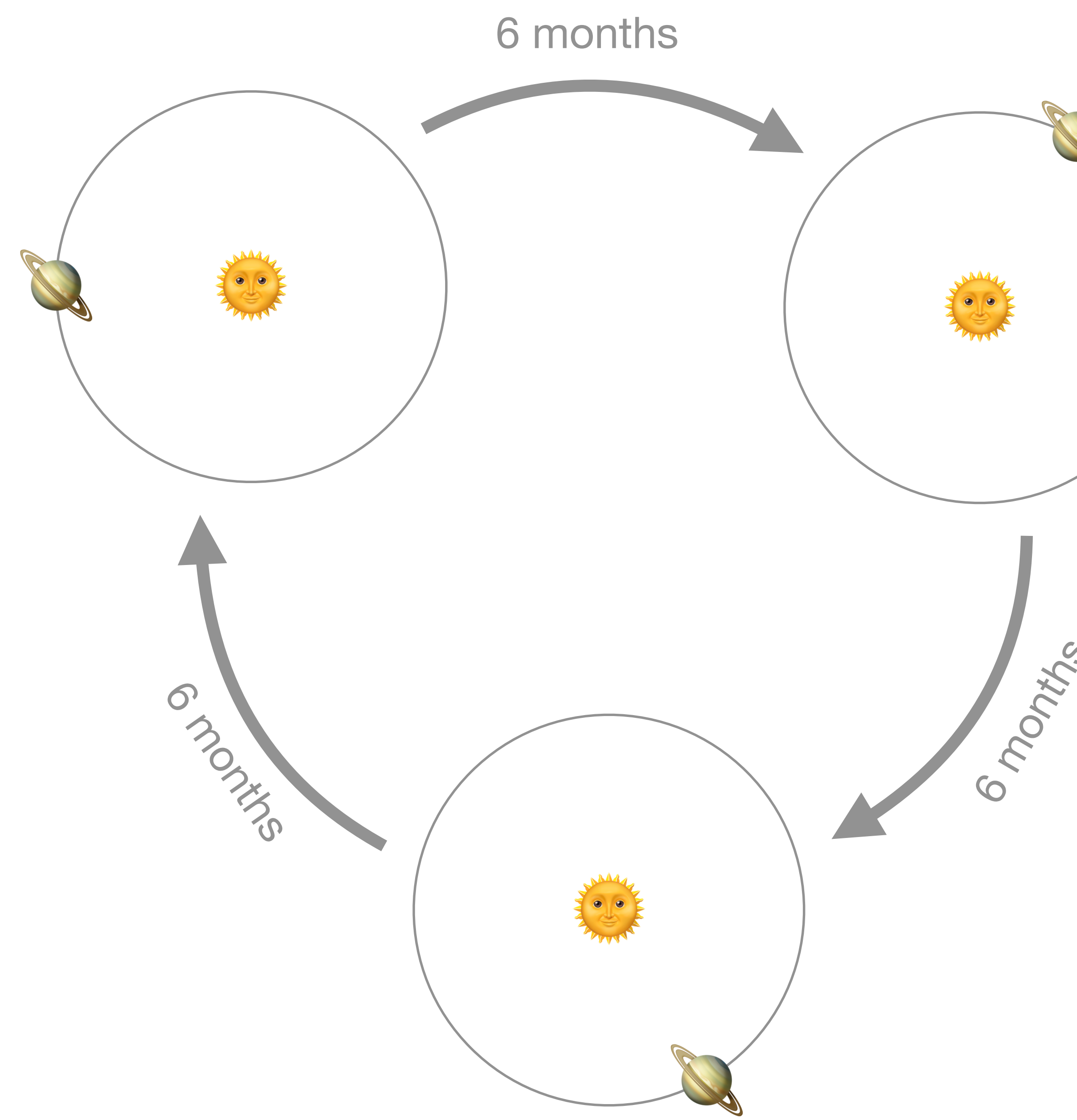


Decomposing the system

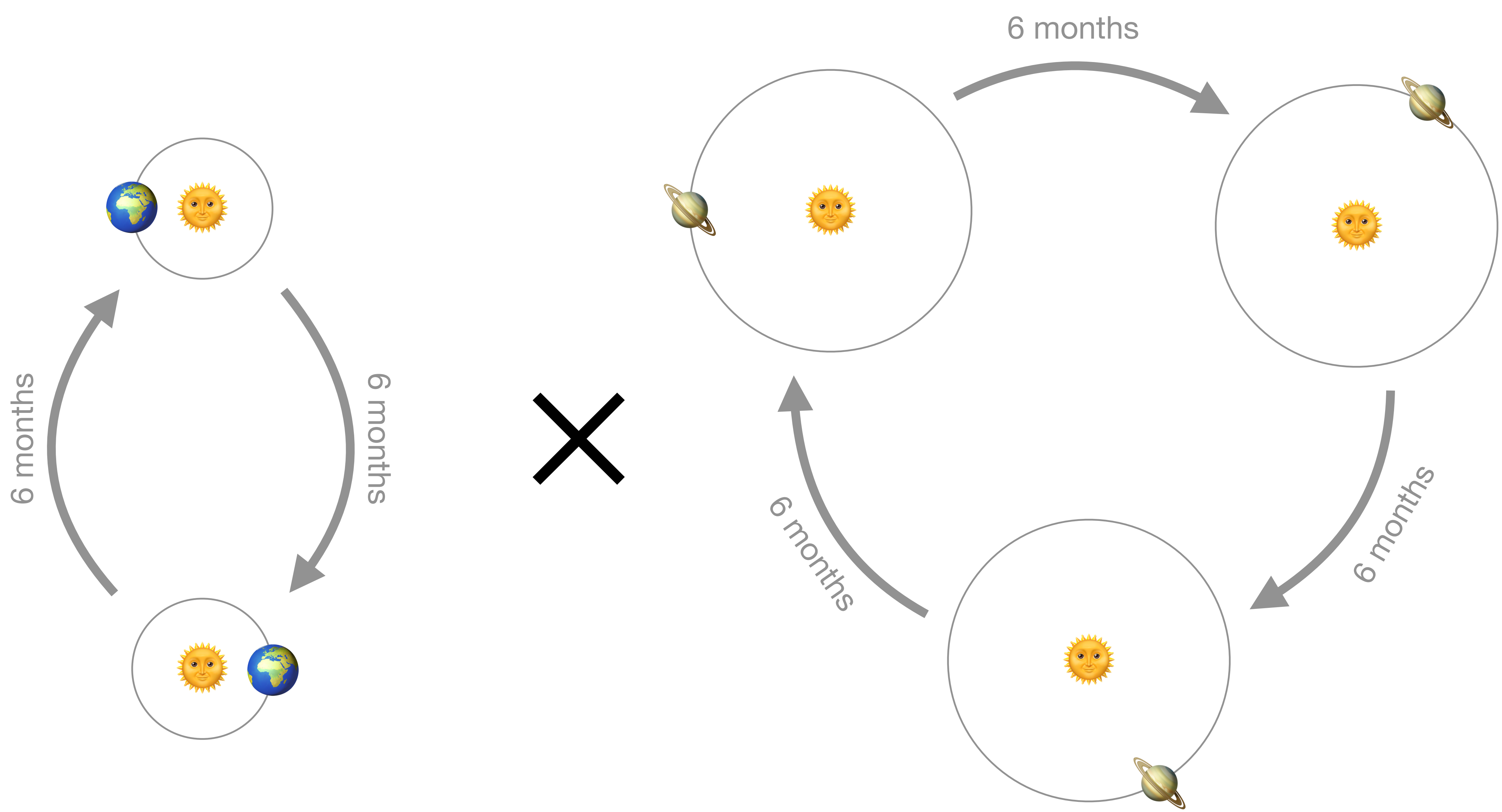


Decomposing the system



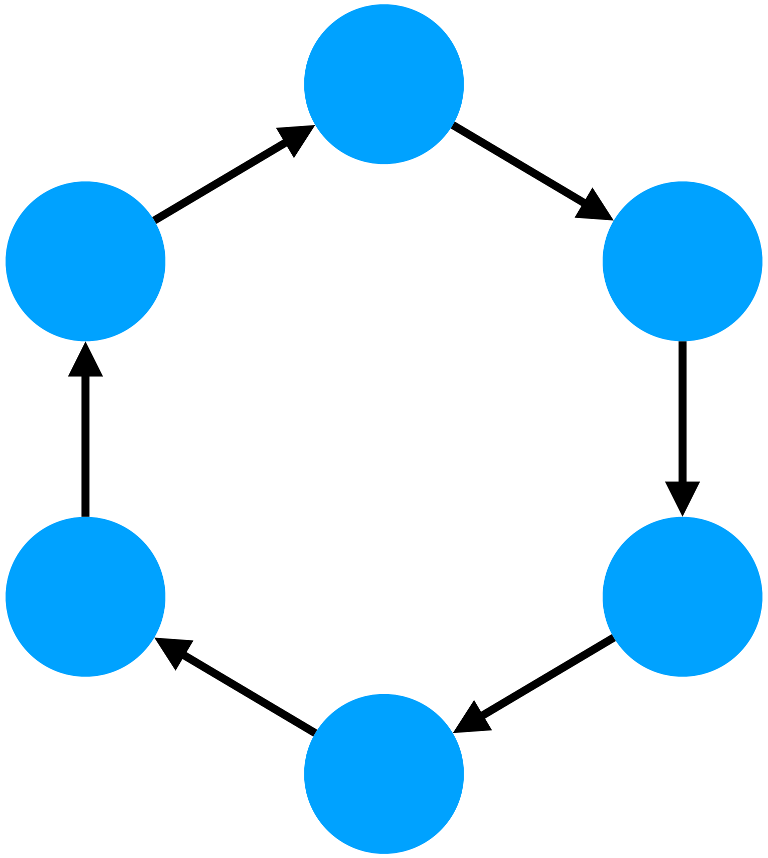


Decomposing the system

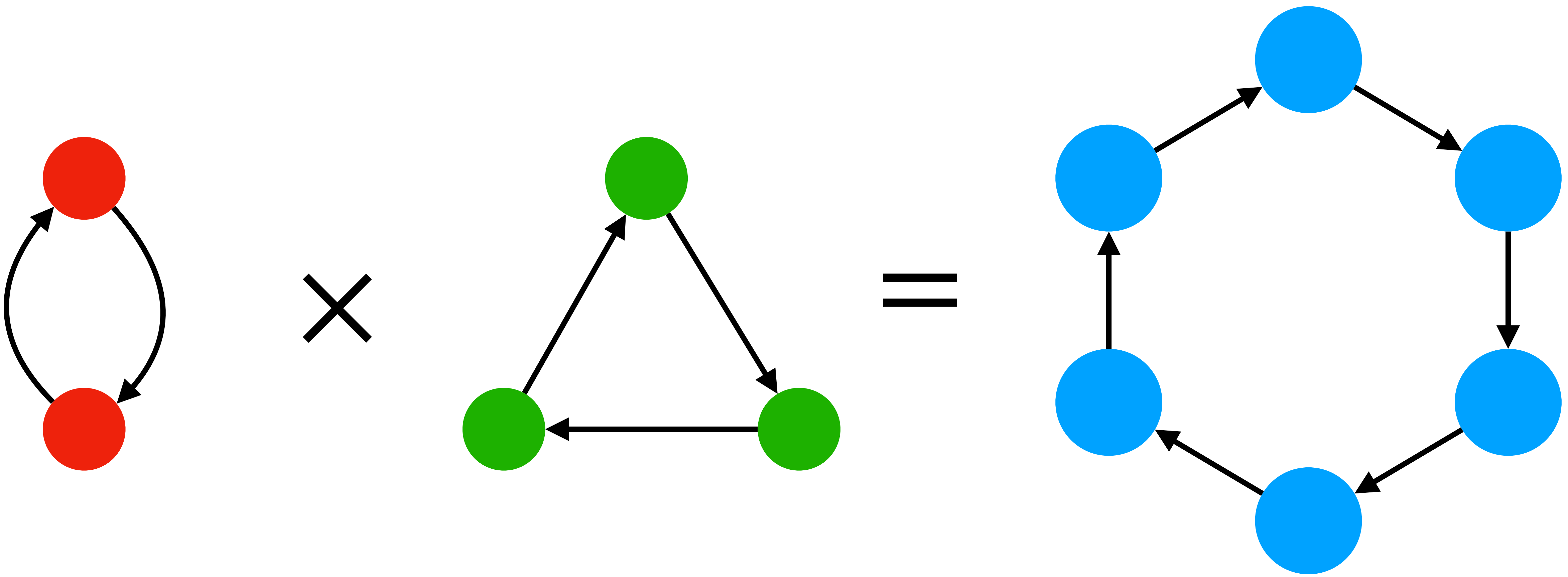


What if our instruments are less sophisticated?

Abstract evolution of the system

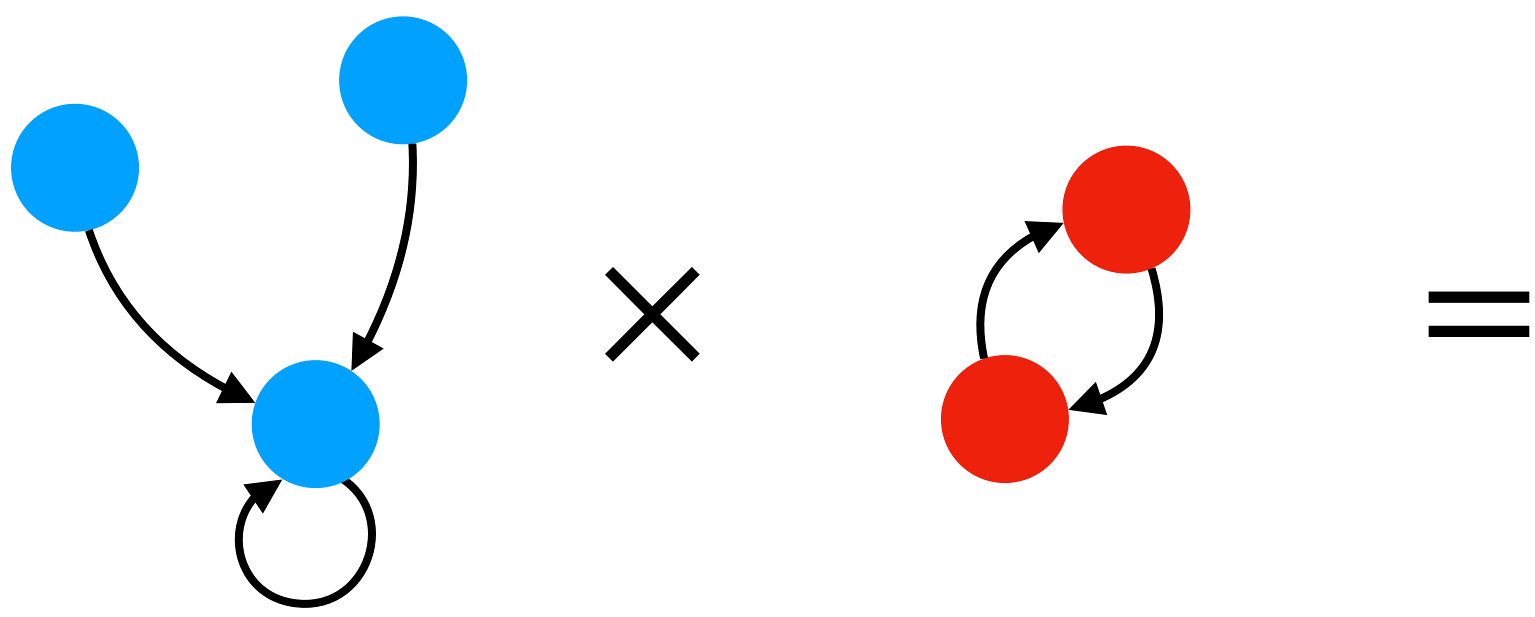


Abstract evolution of the system

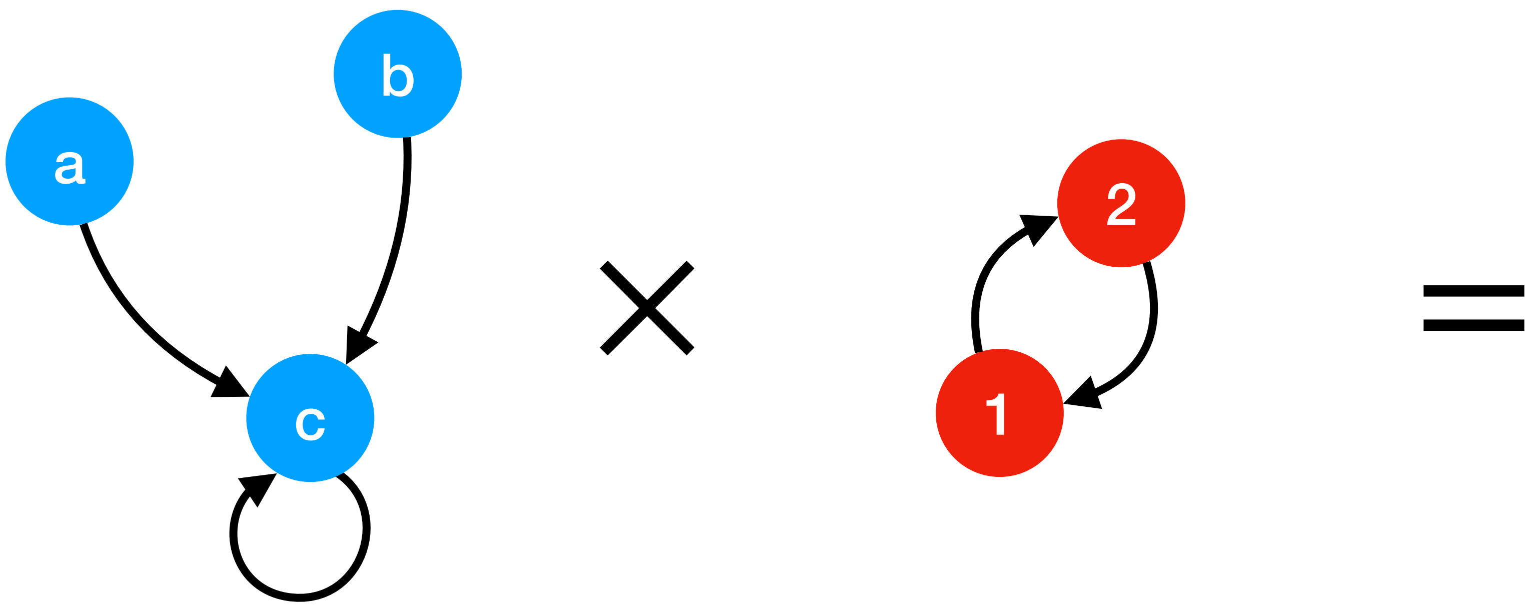


Product of dynamical systems

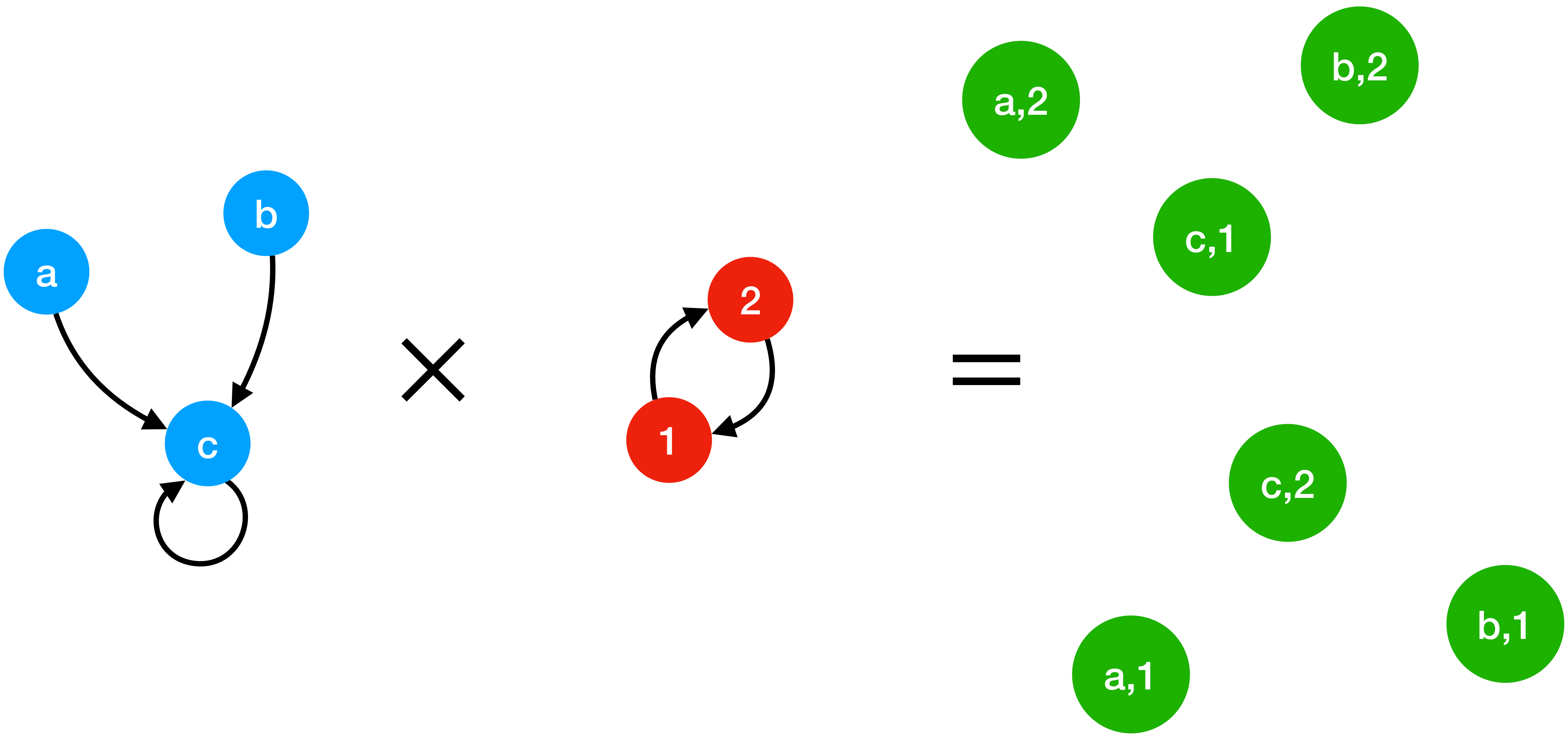
Product of systems



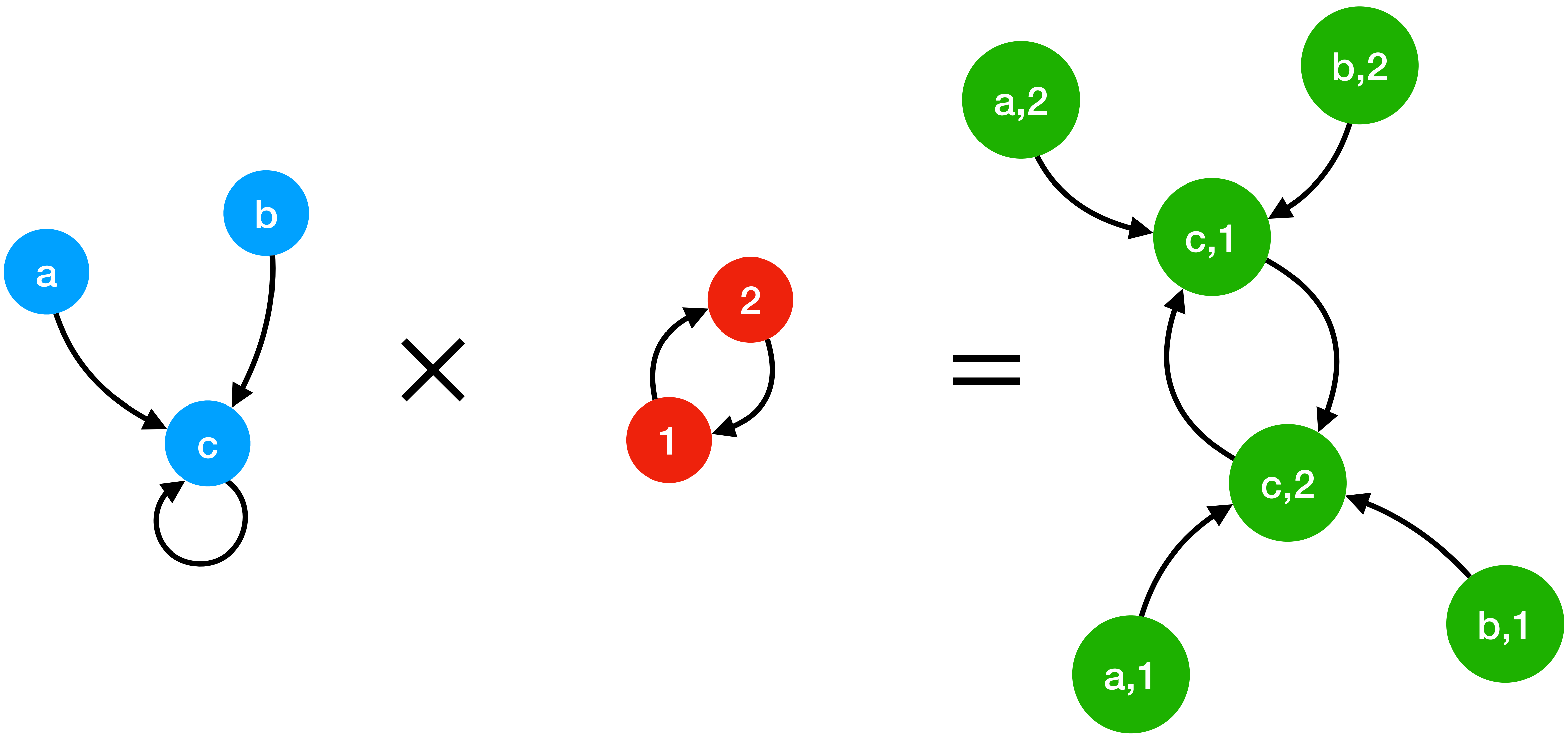
Give temporary names to the states



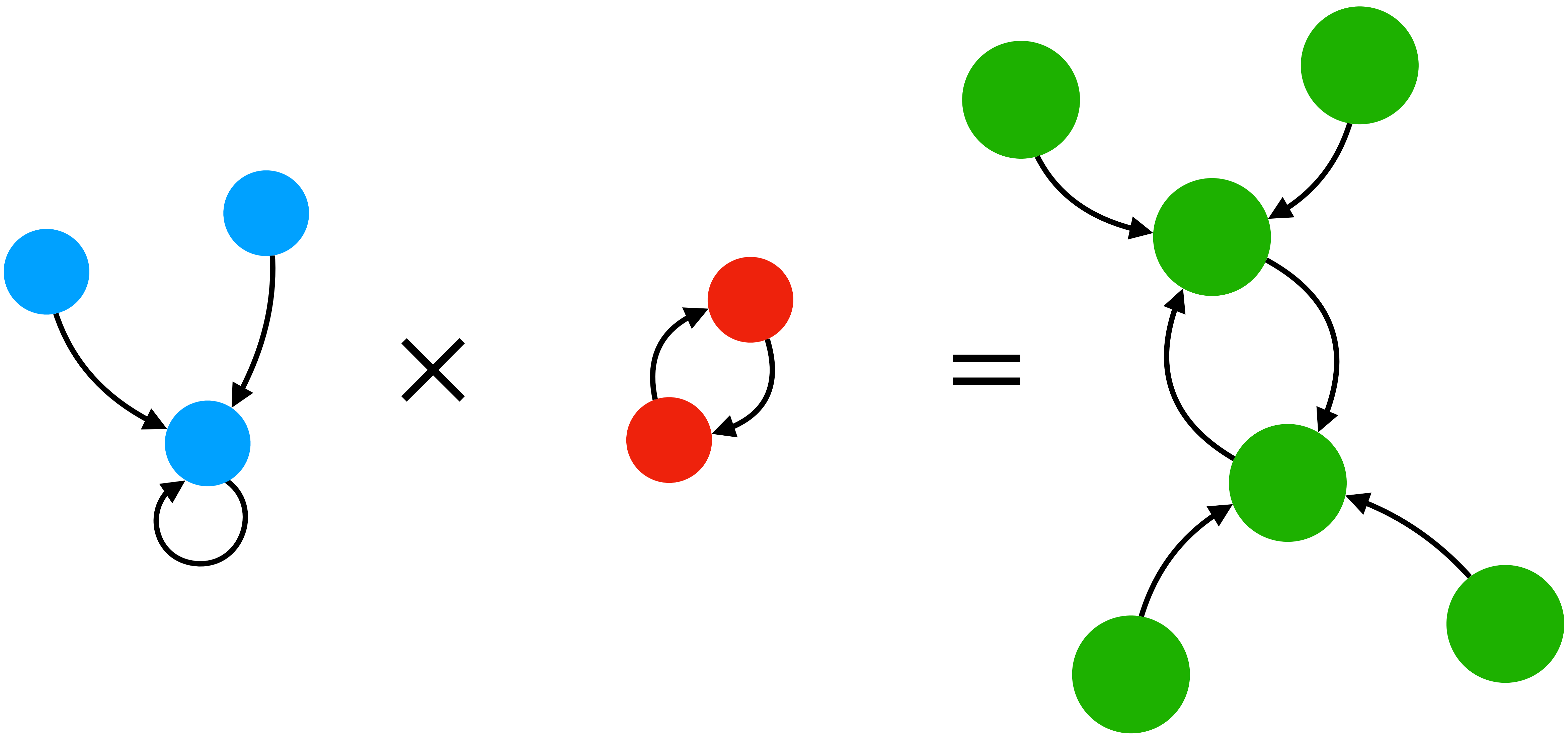
Compute the Cartesian product



Add the arcs between states

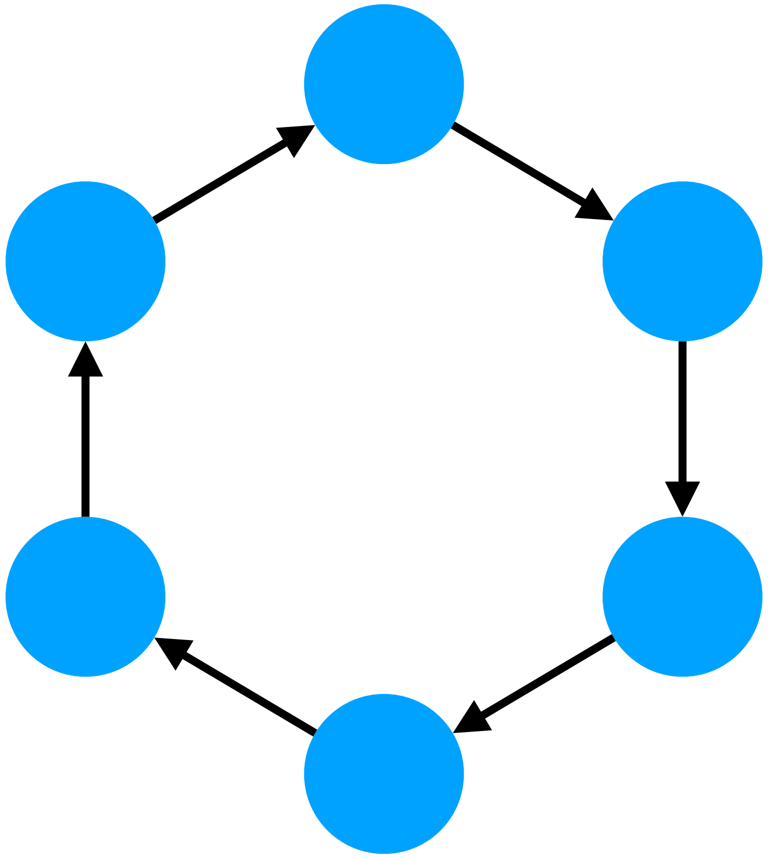


Forget the names once again

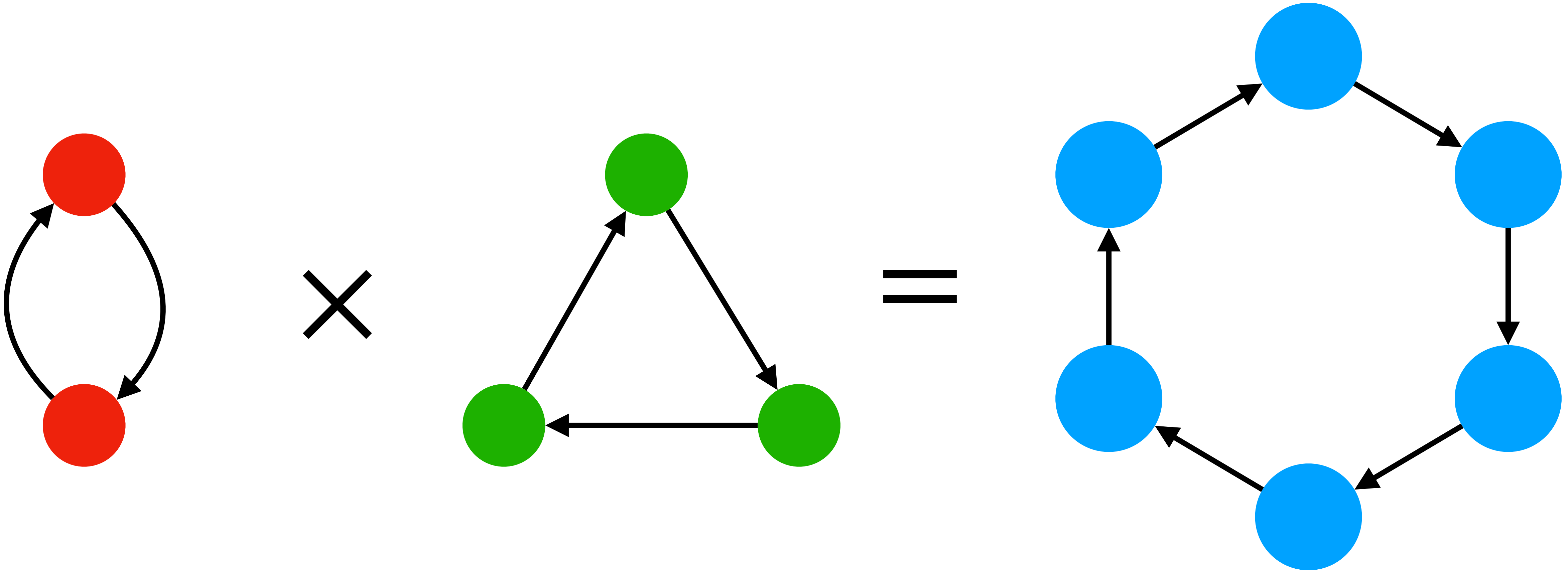


Back to our planetary system

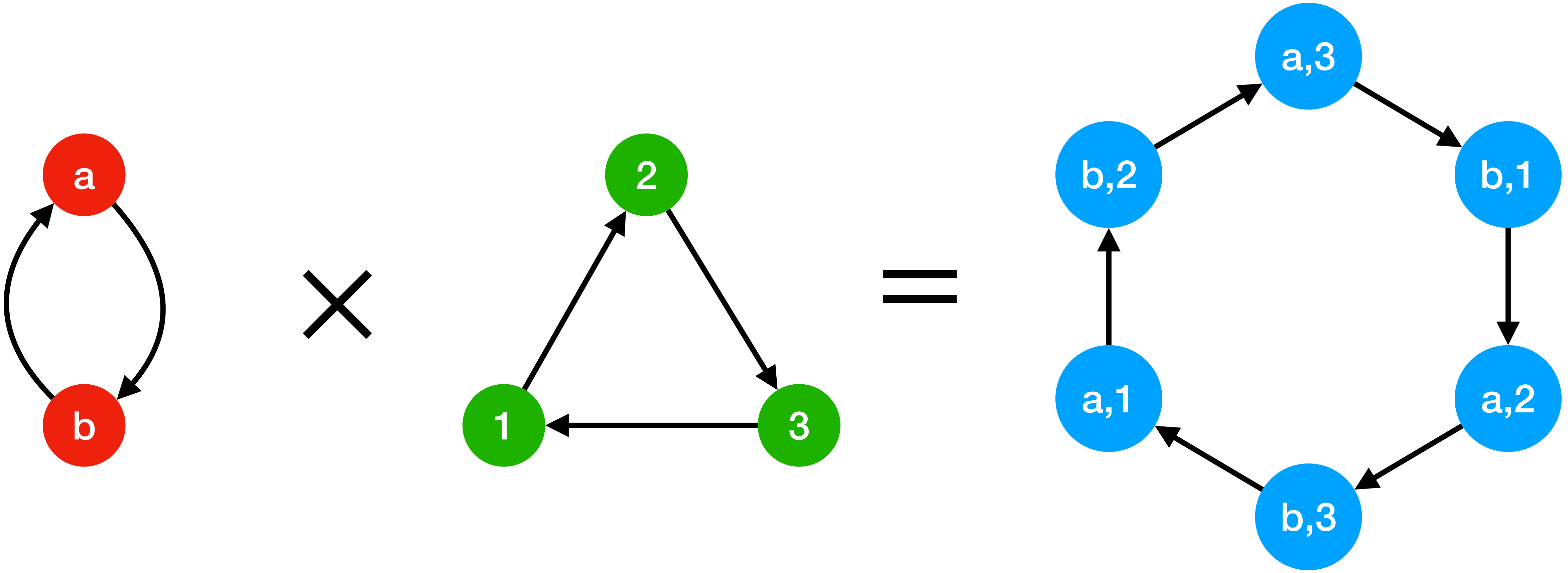
Decomposition



Decomposition

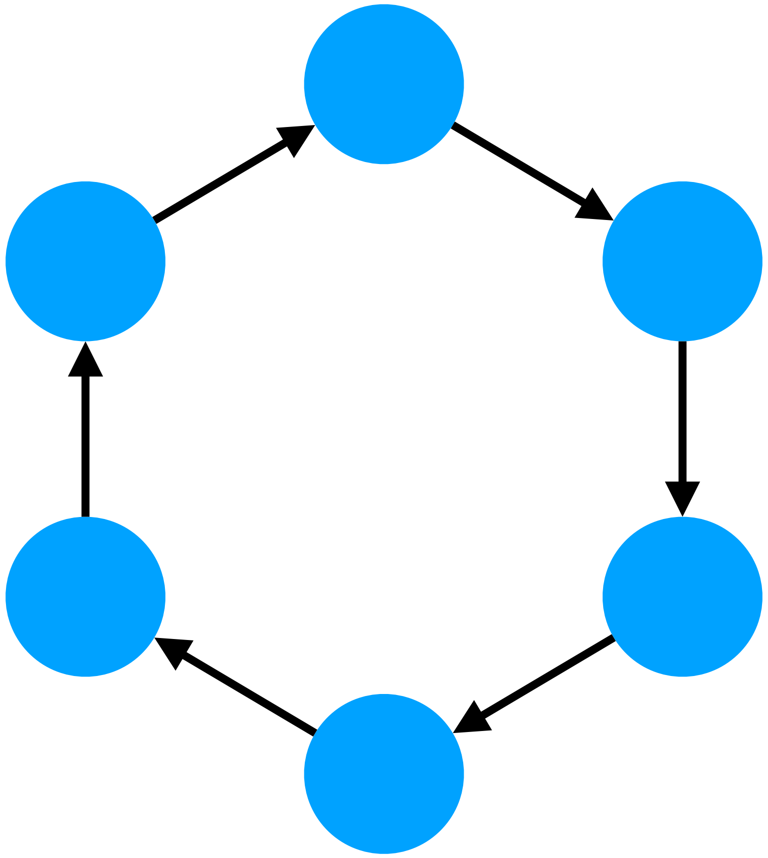


Decomposition

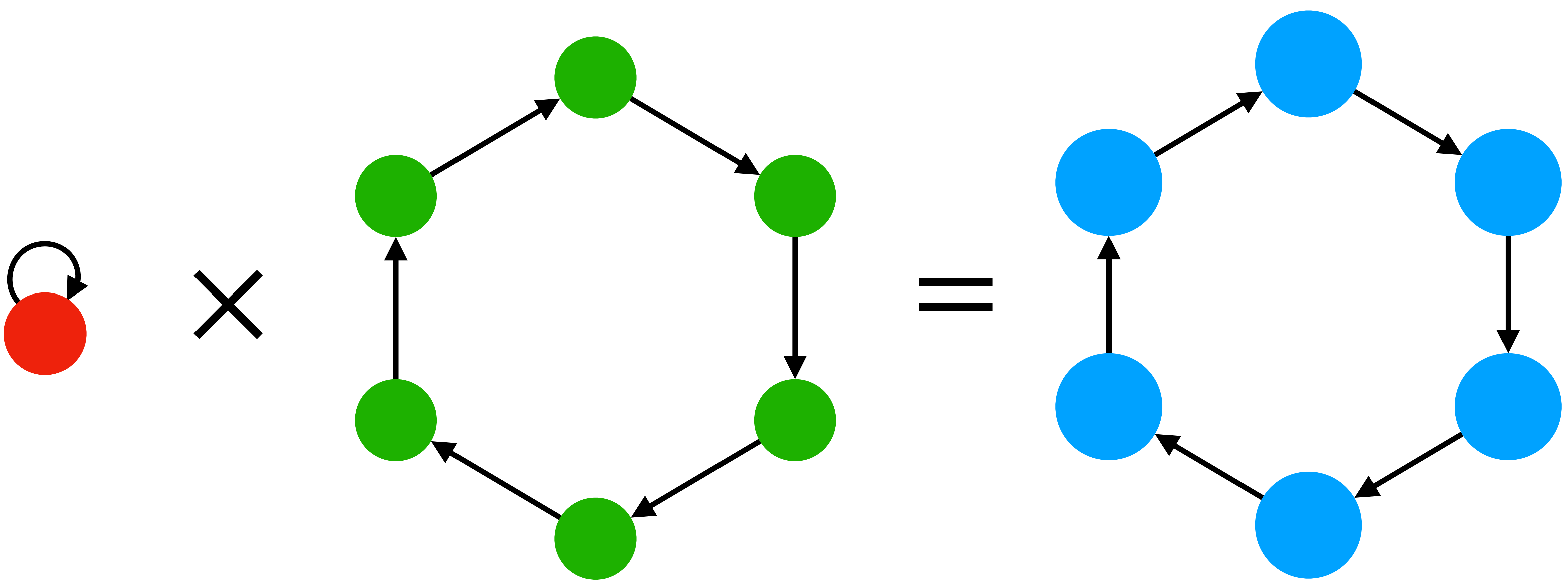


Any other decomposition?

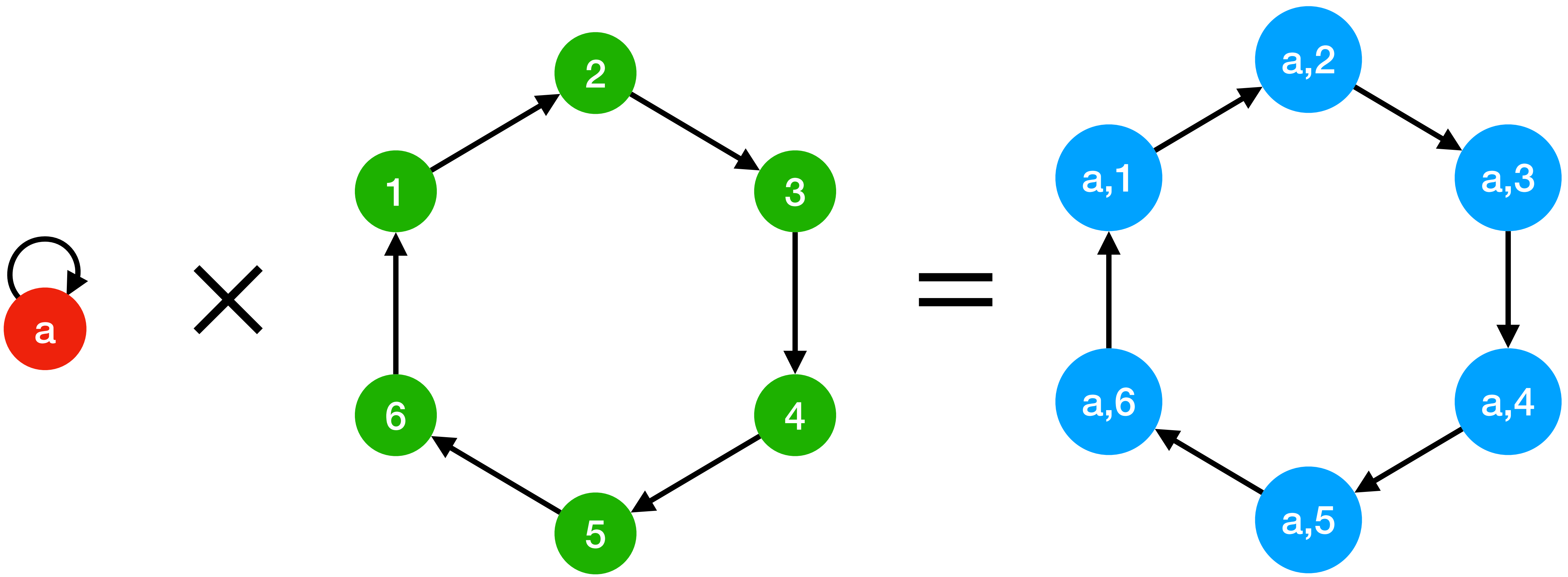
Another decomposition

$=$ 

Another decomposition

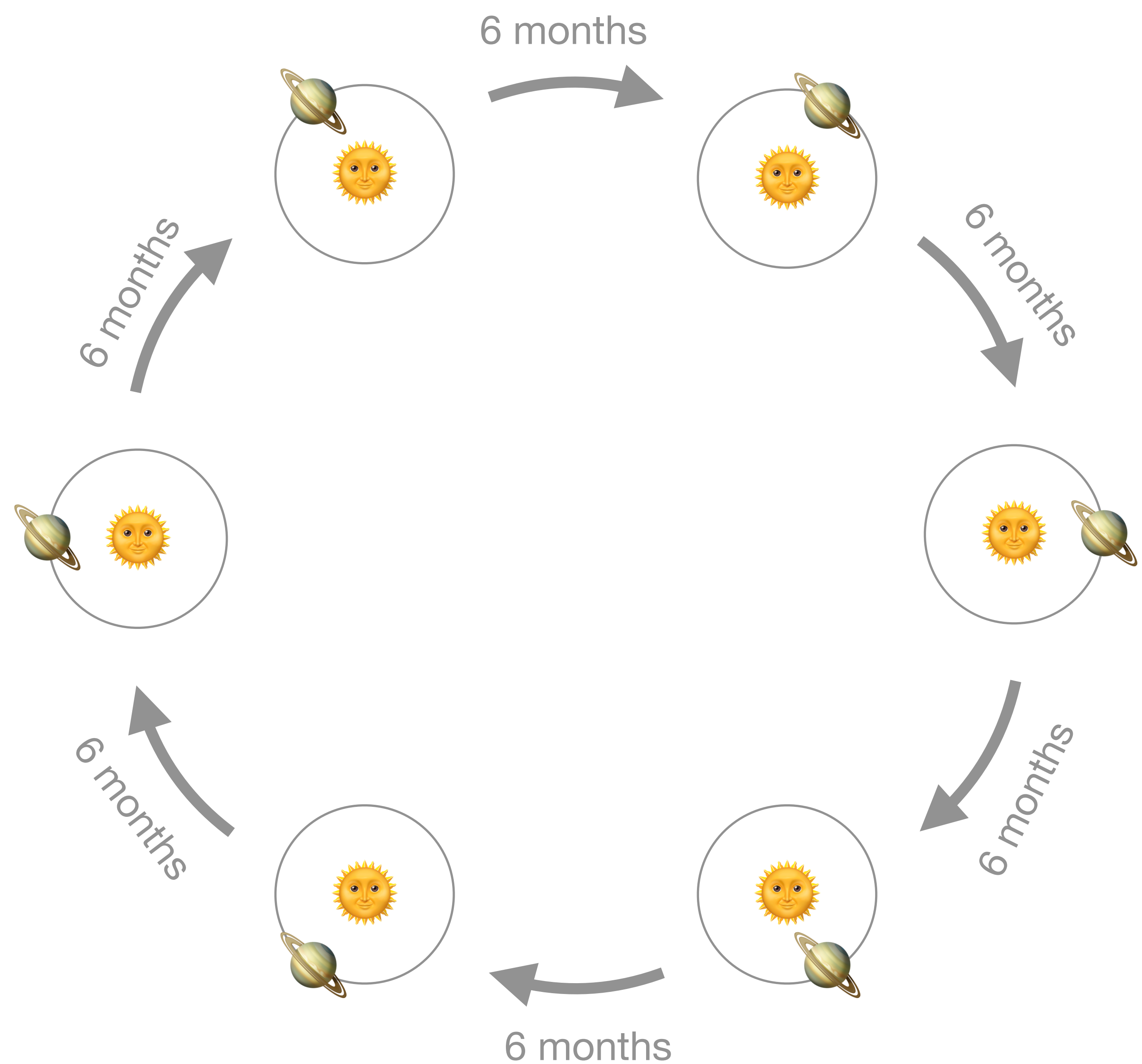


Another decomposition

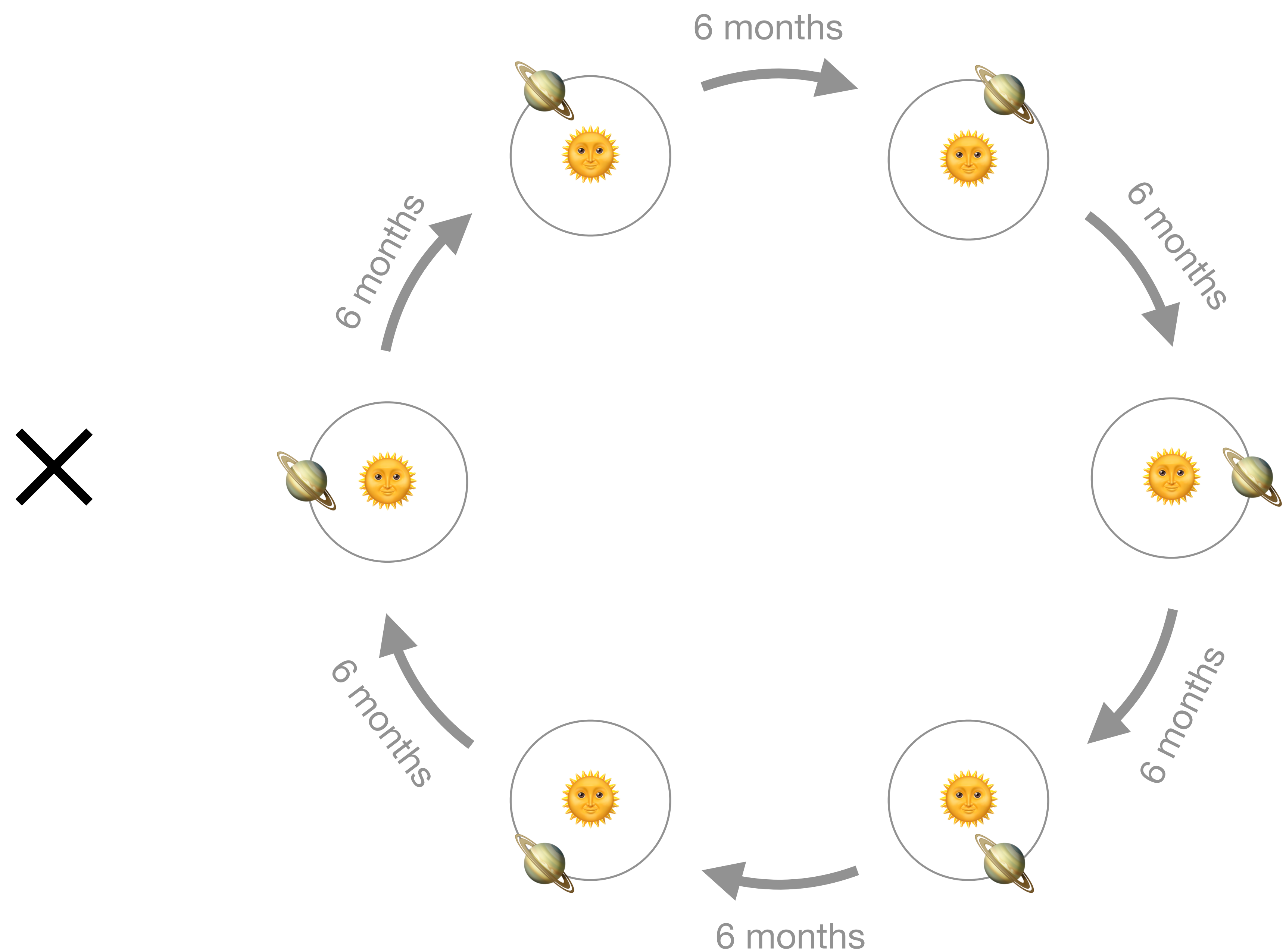


More concretely…

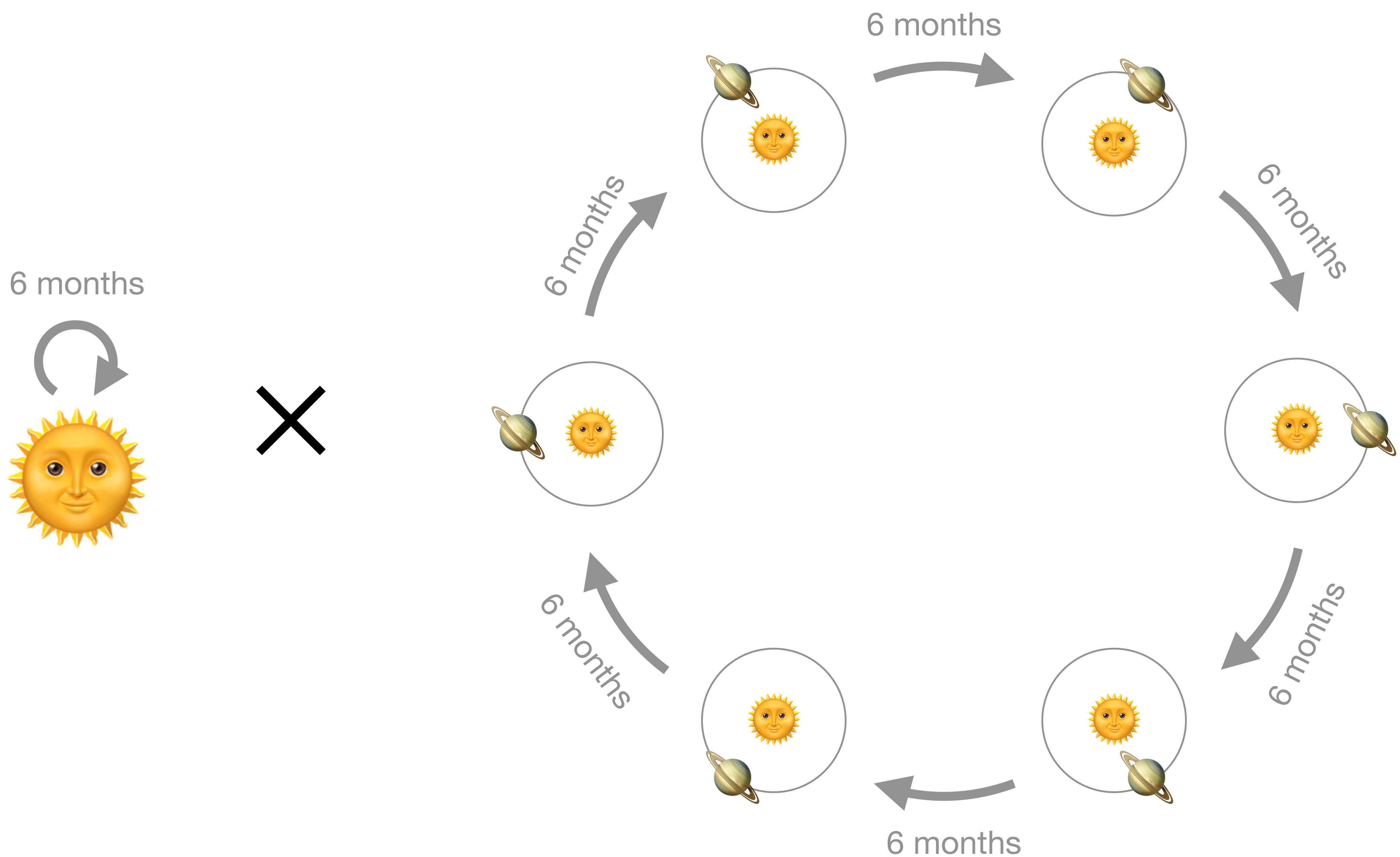
More concretely…



More concretely…

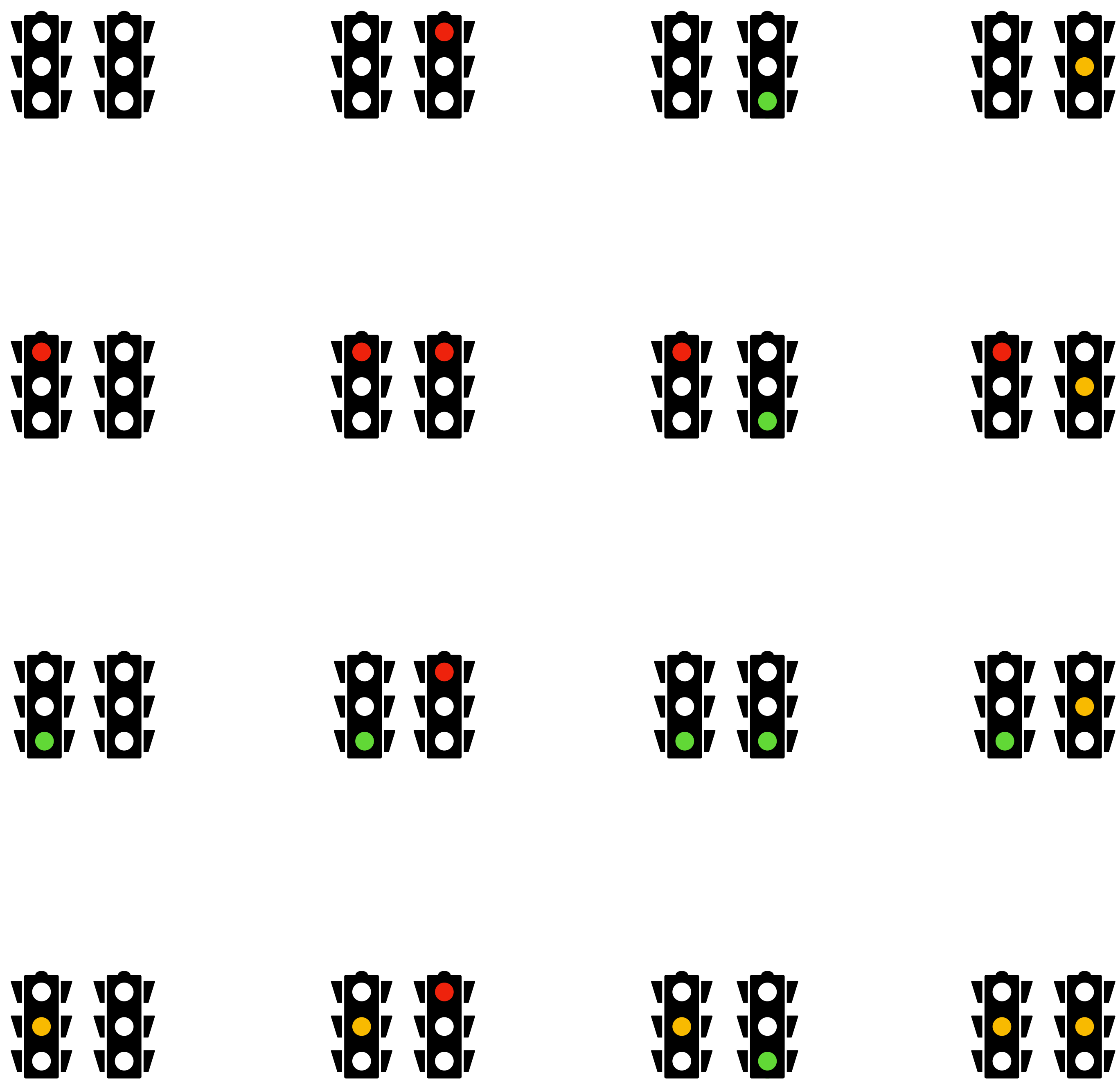


More concretely…

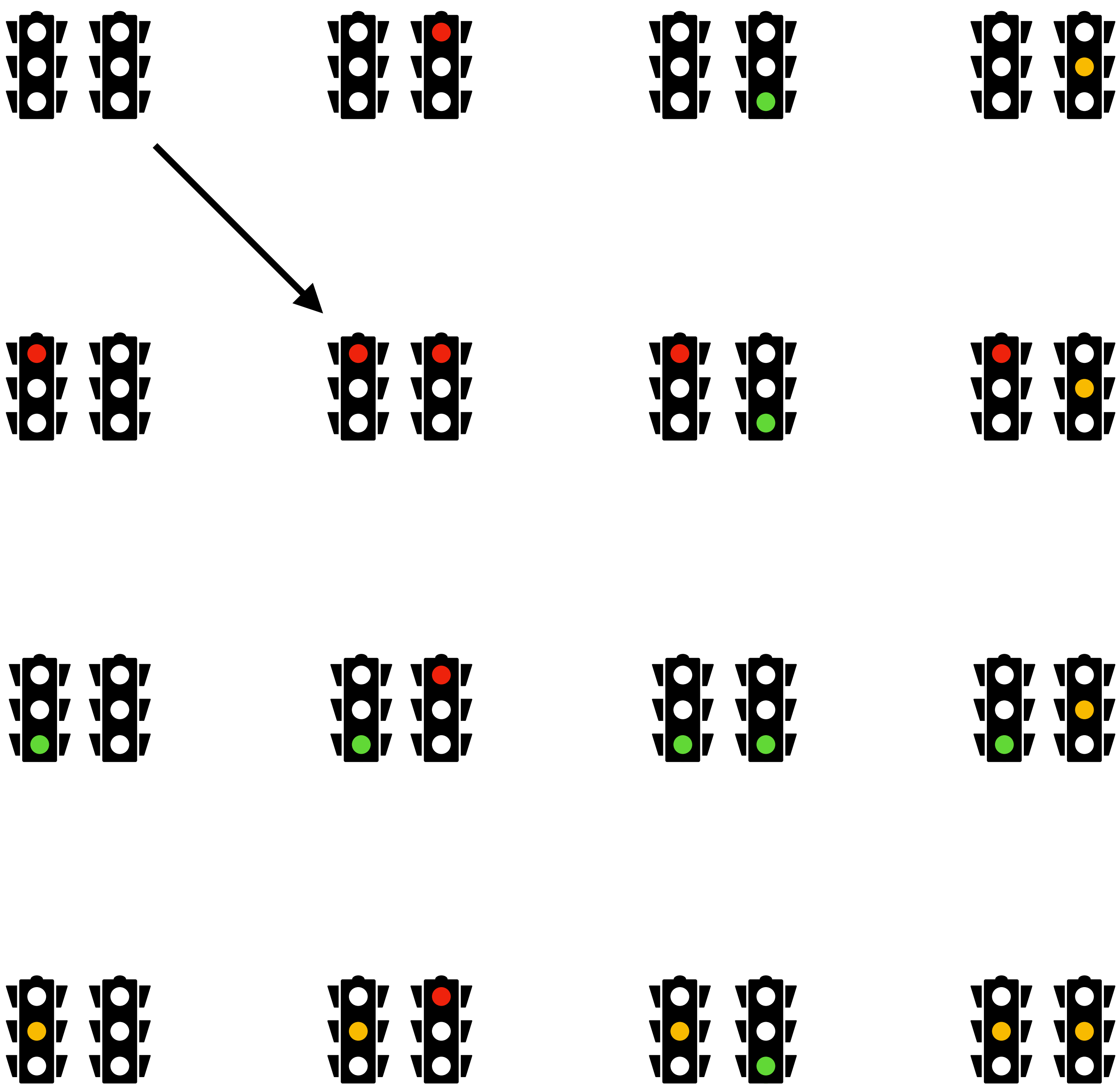


Untangling complex systems

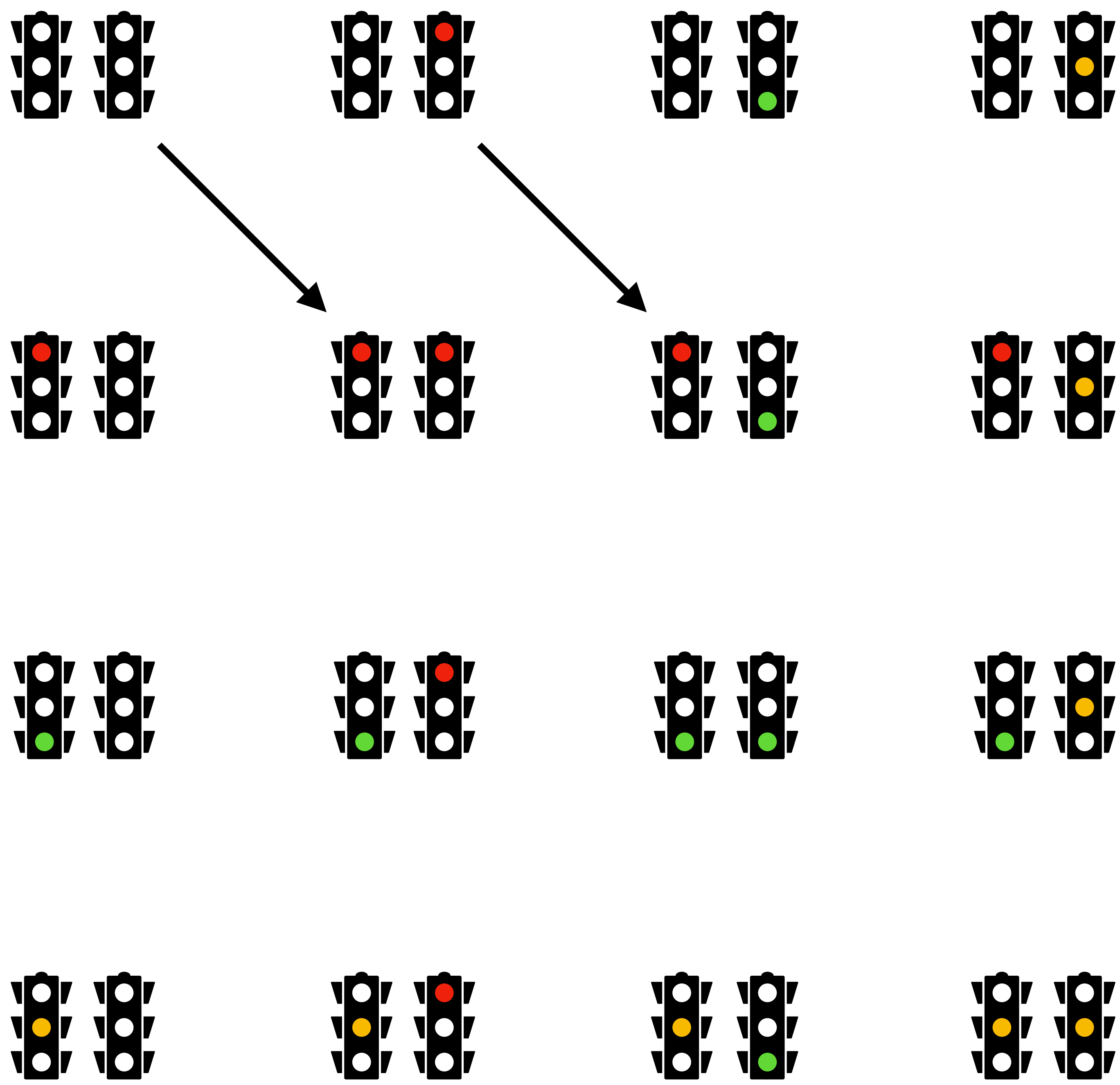
Traffic lights at a crossroads



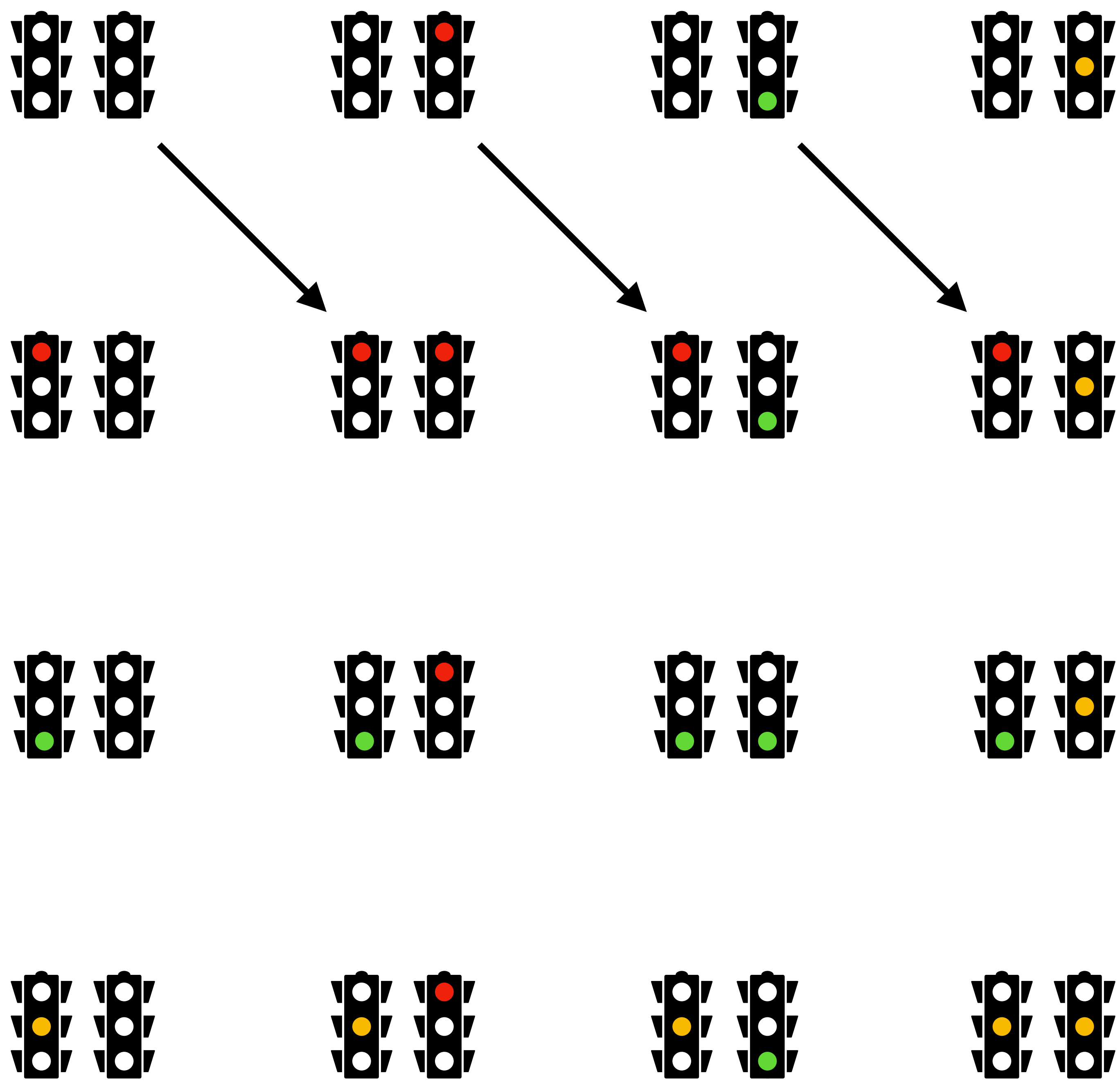
Traffic lights at a crossroads



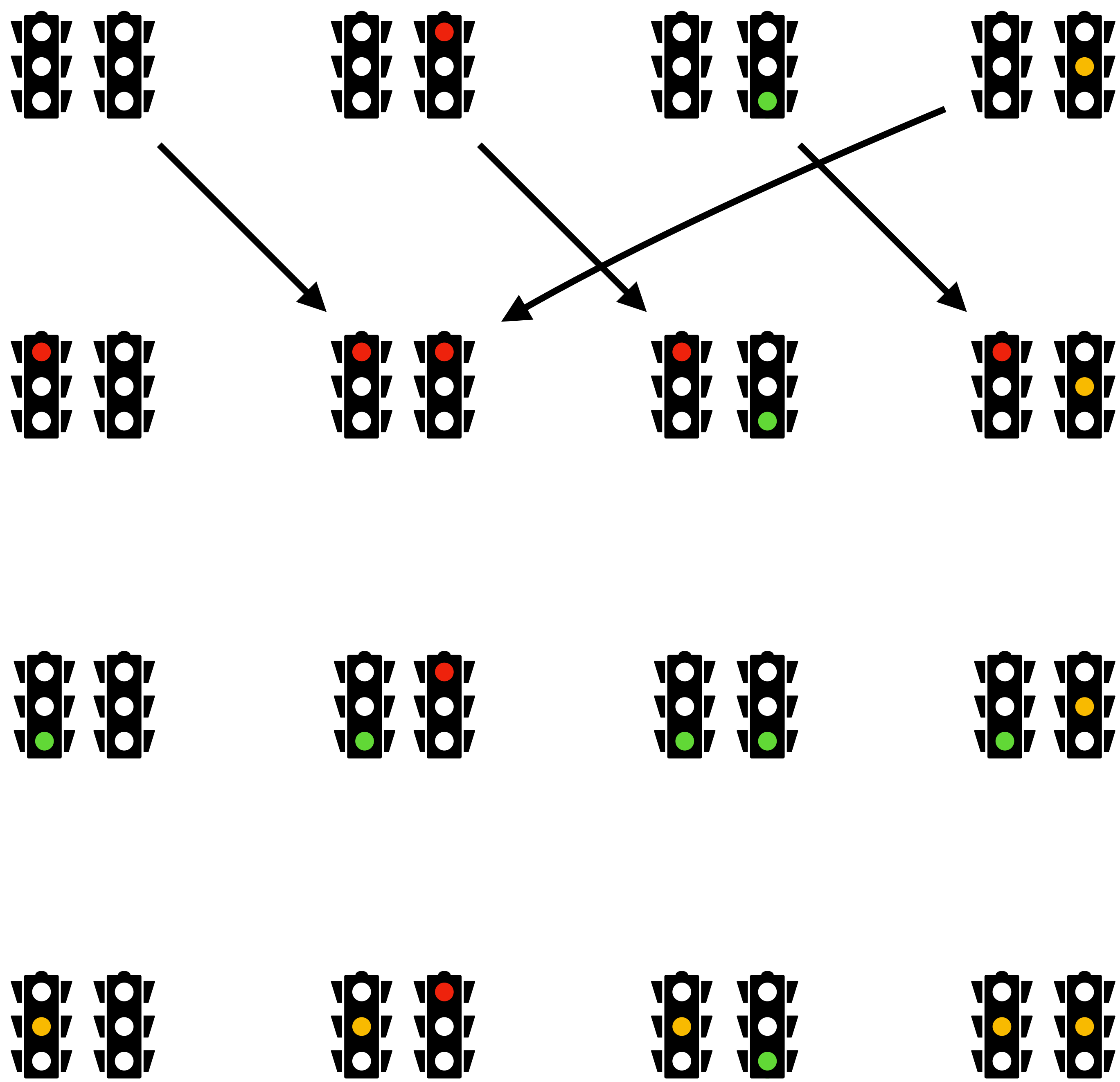
Traffic lights at a crossroads



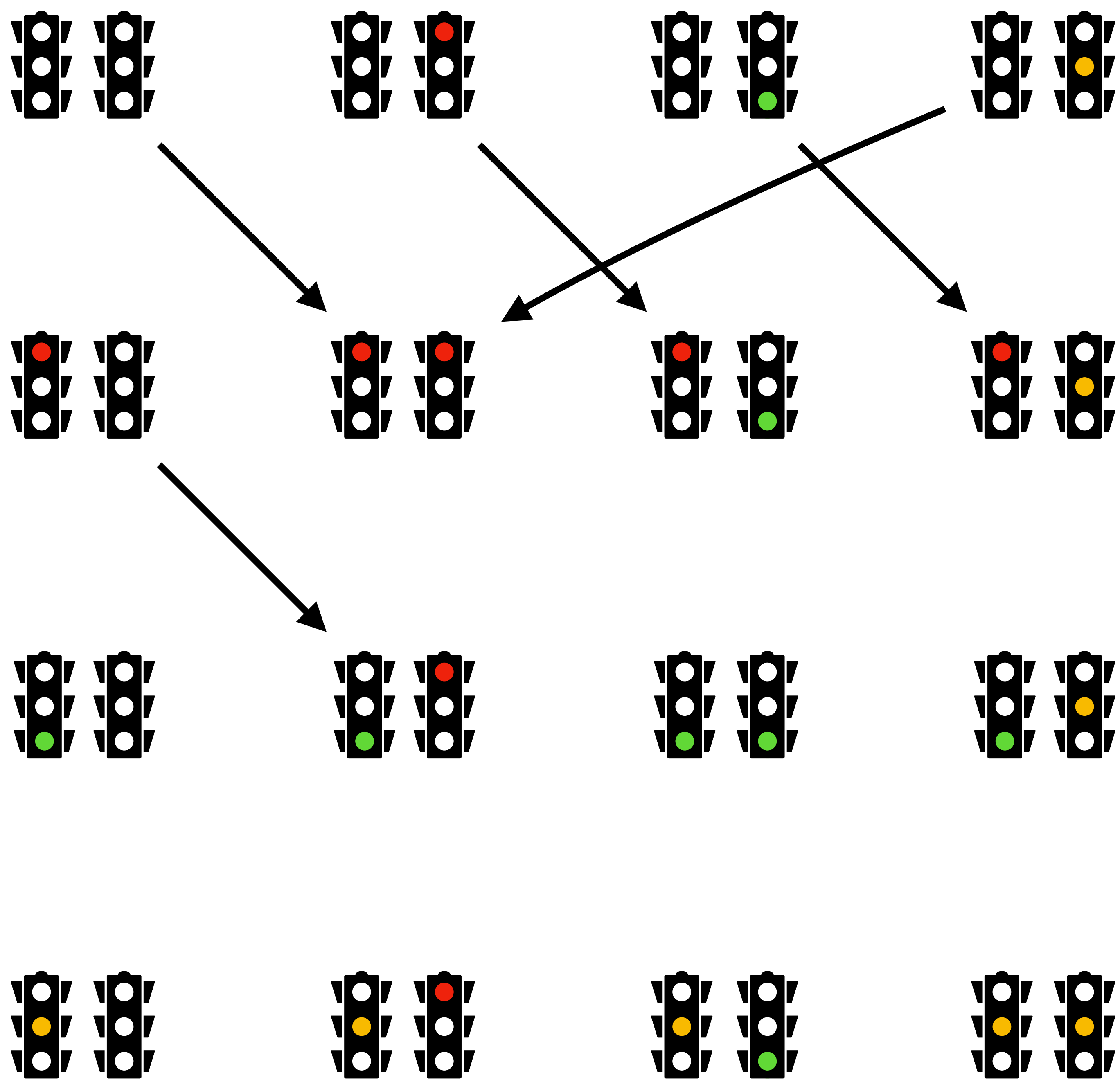
Traffic lights at a crossroads



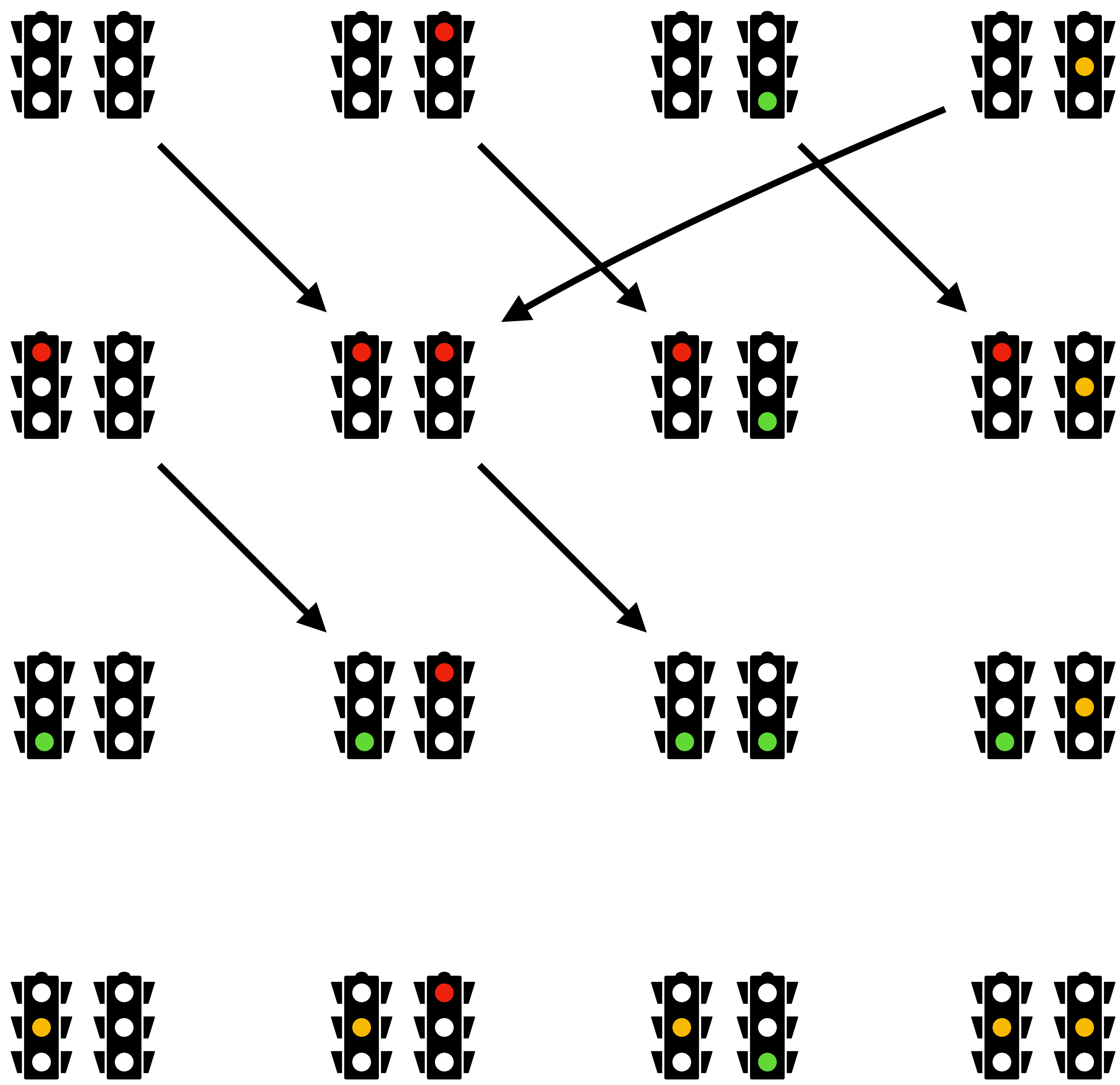
Traffic lights at a crossroads



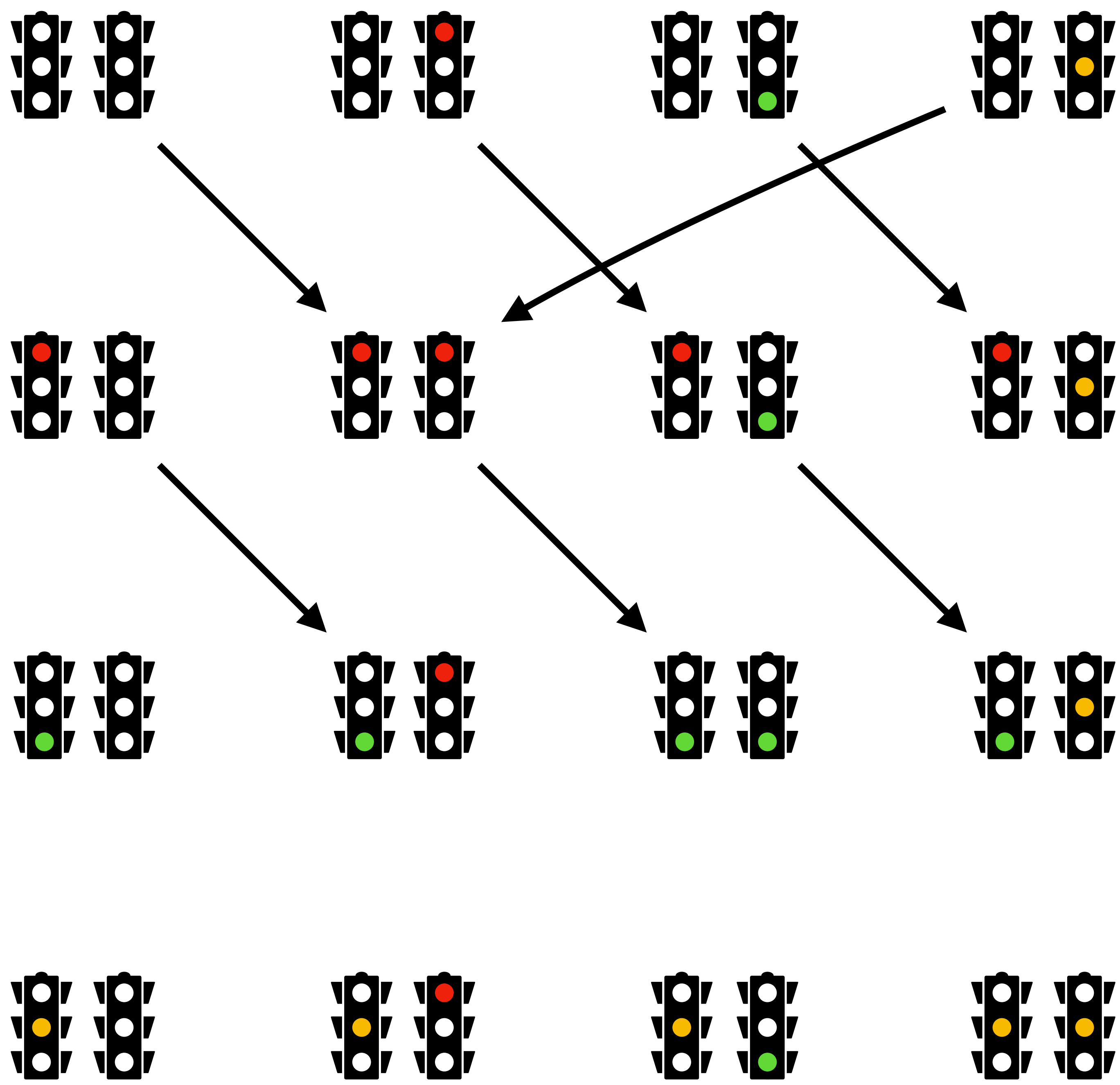
Traffic lights at a crossroads



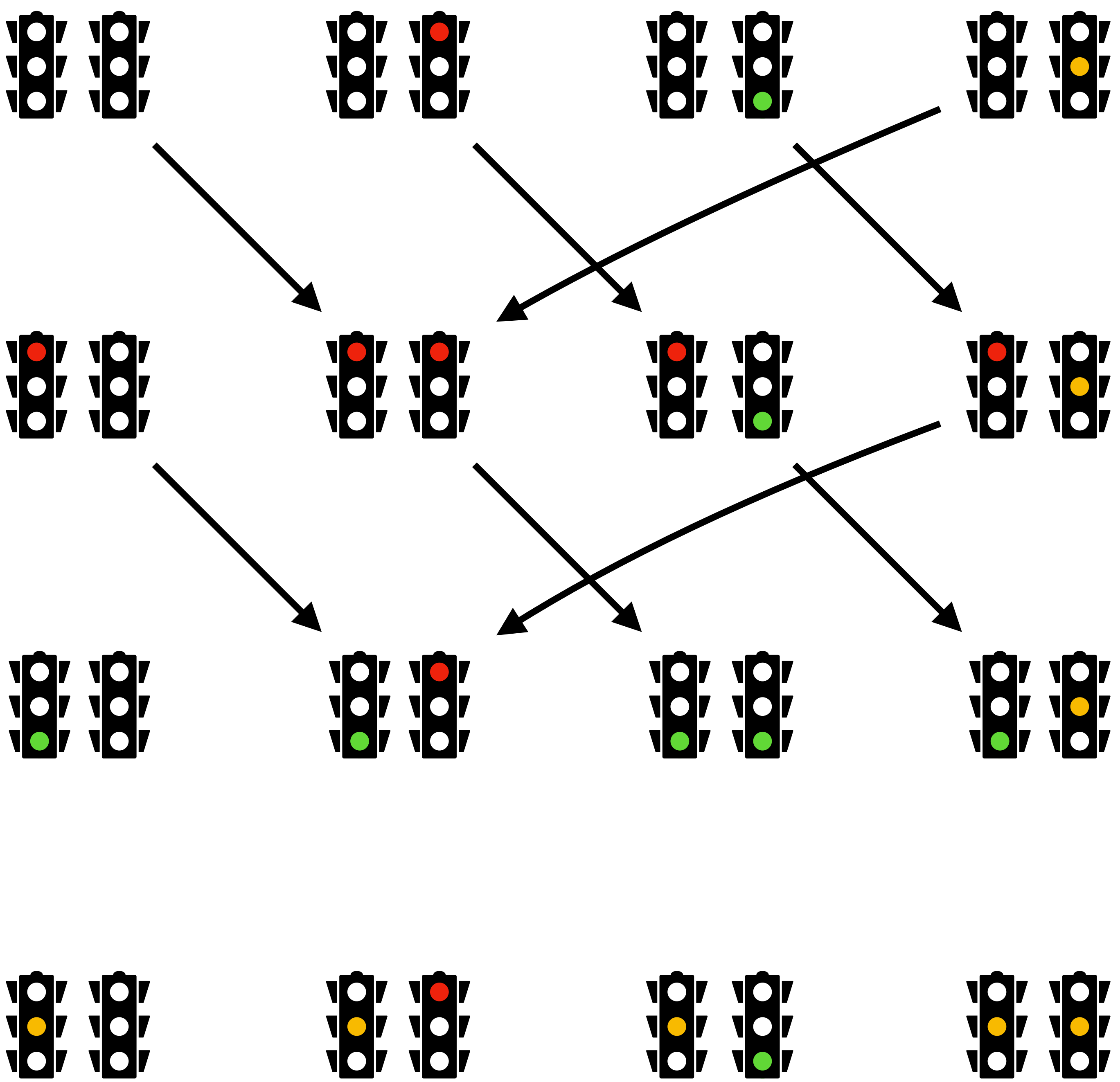
Traffic lights at a crossroads



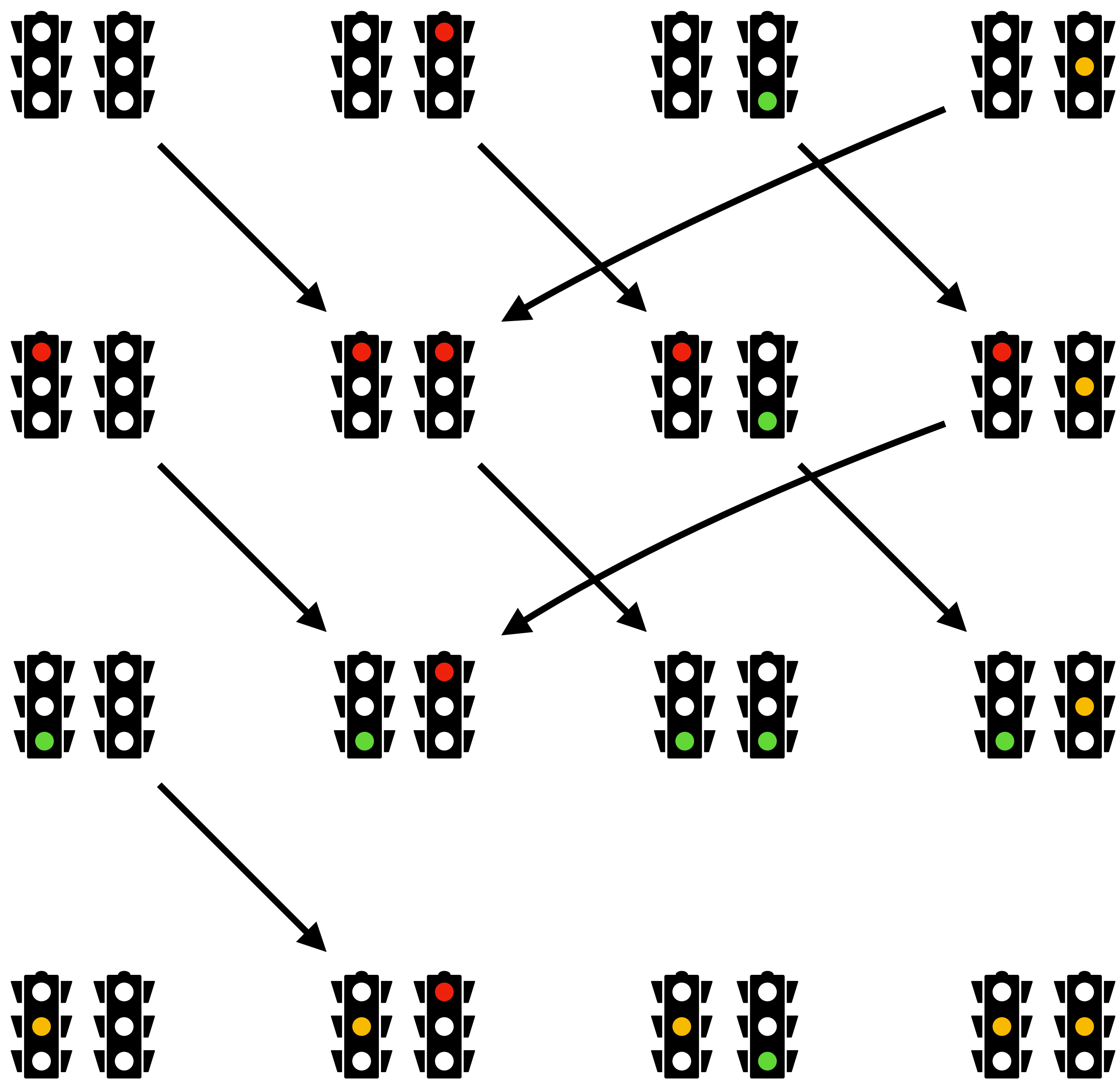
Traffic lights at a crossroads



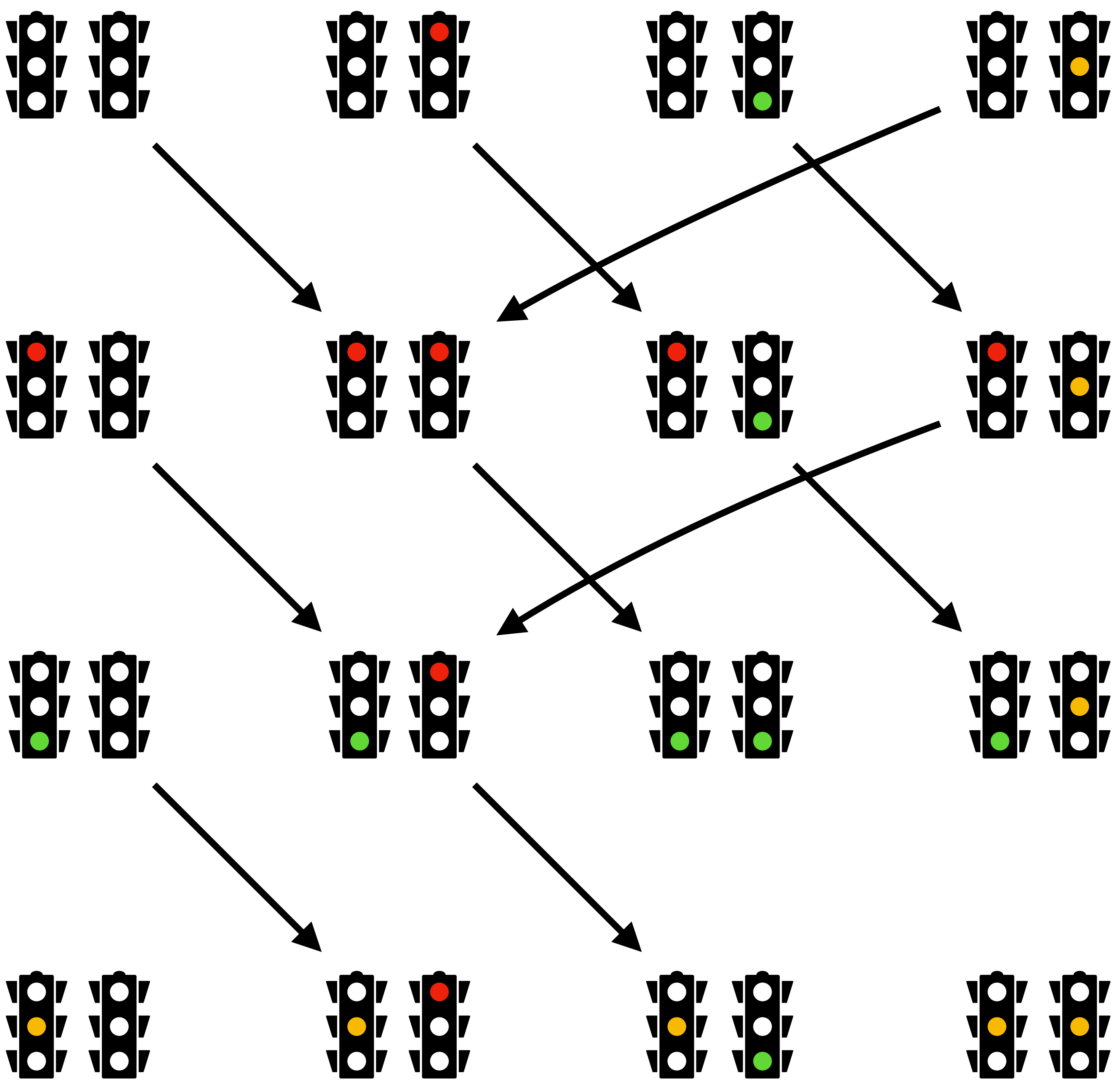
Traffic lights at a crossroads



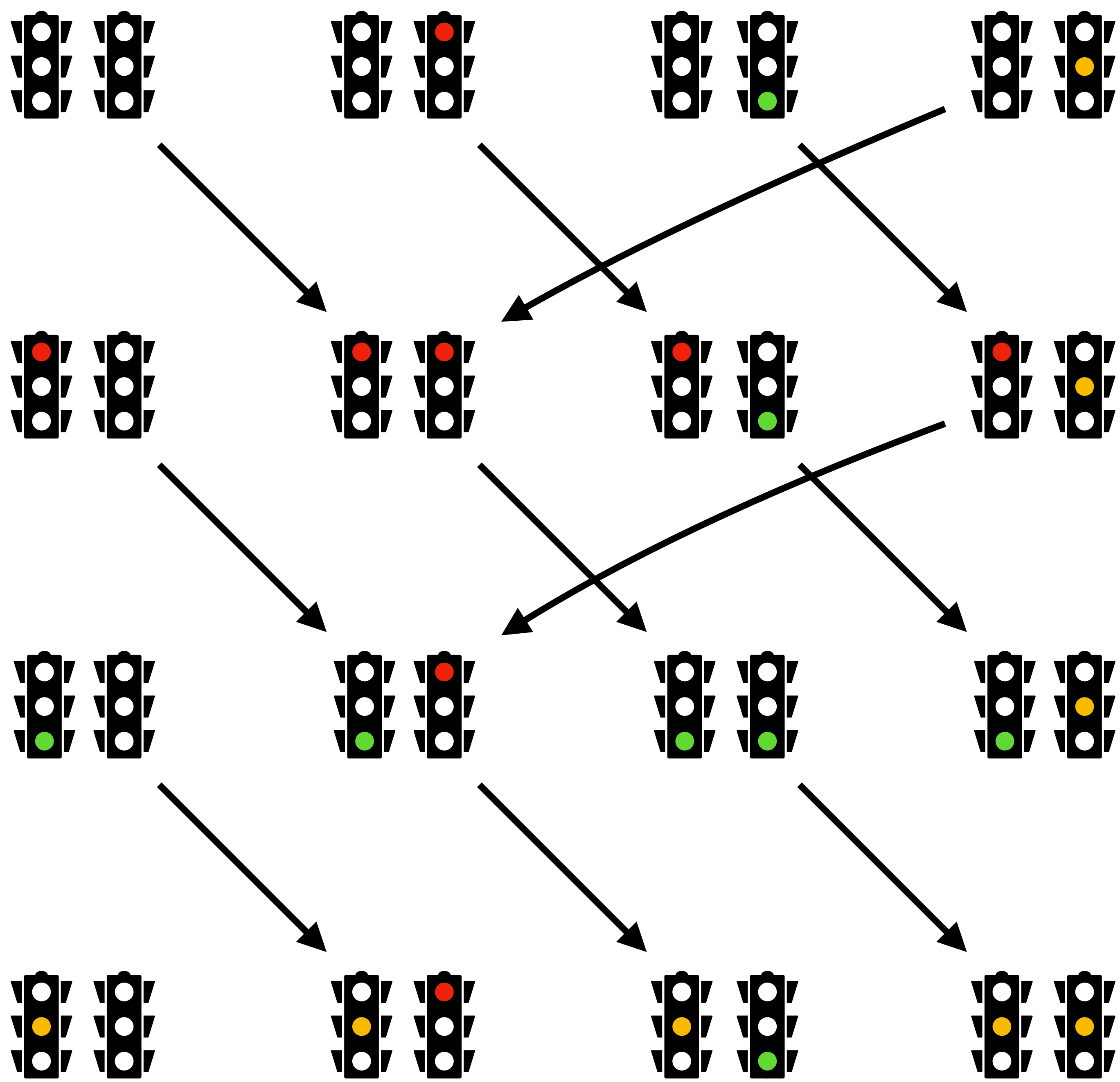
Traffic lights at a crossroads



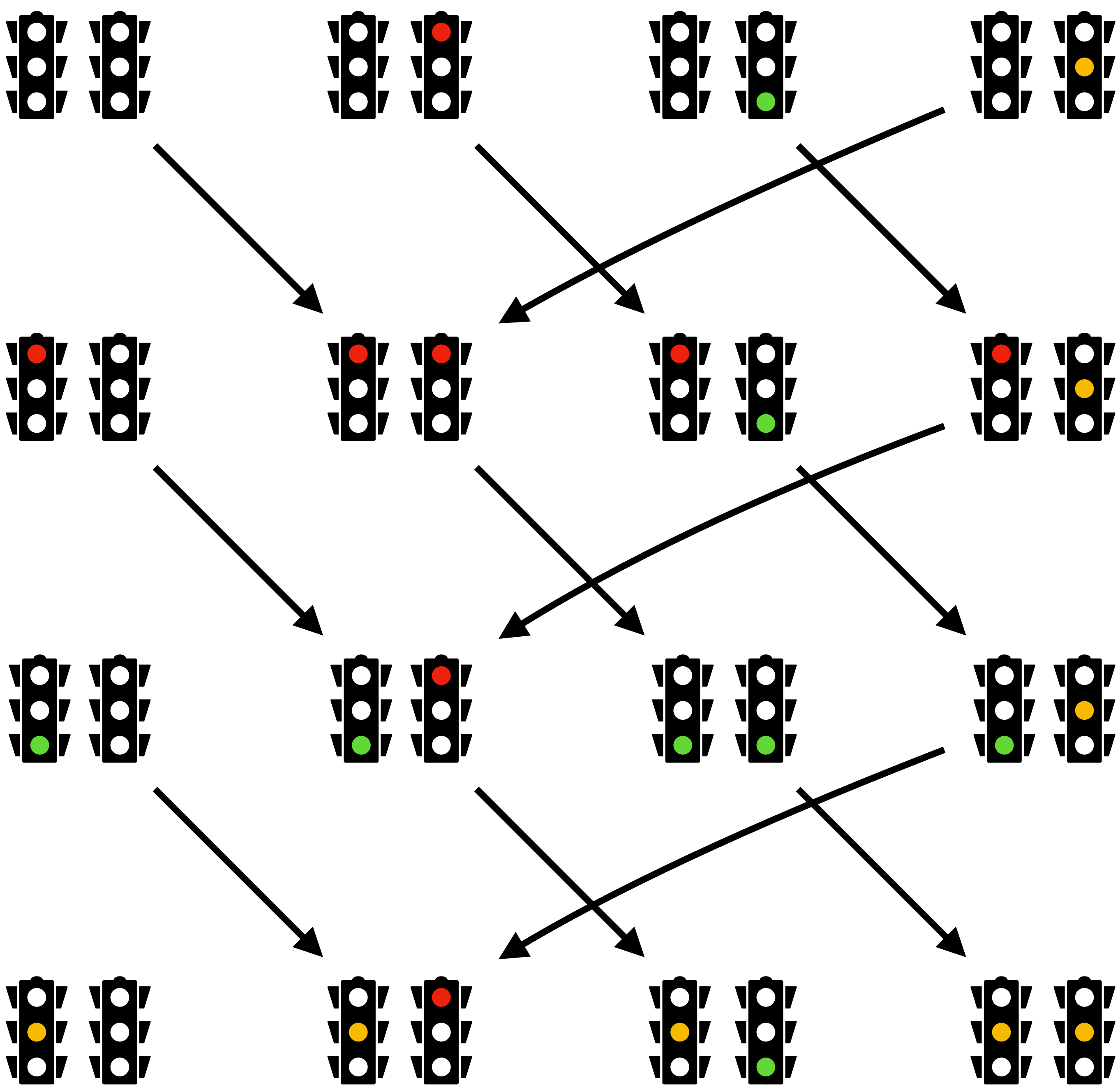
Traffic lights at a crossroads



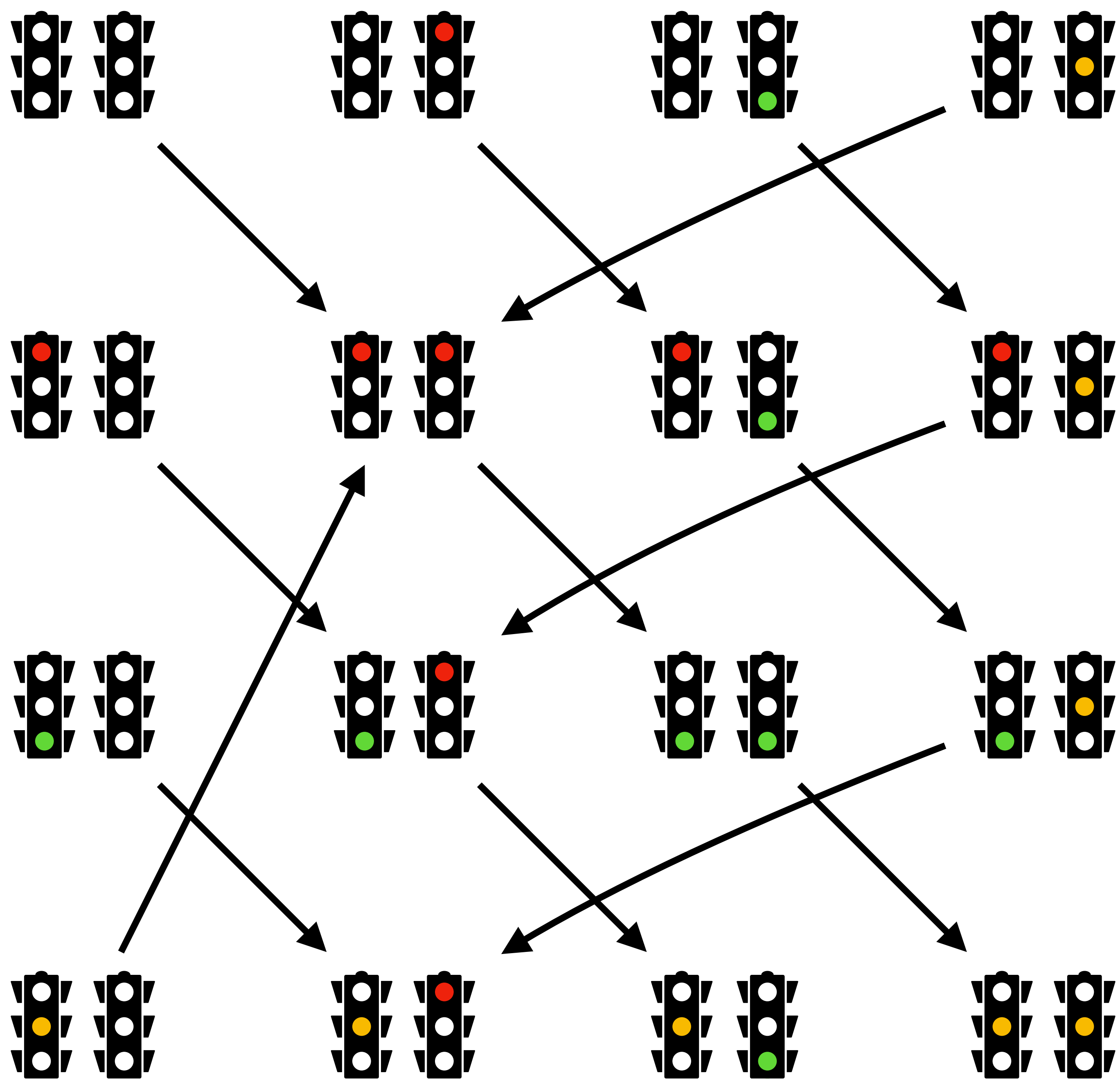
Traffic lights at a crossroads



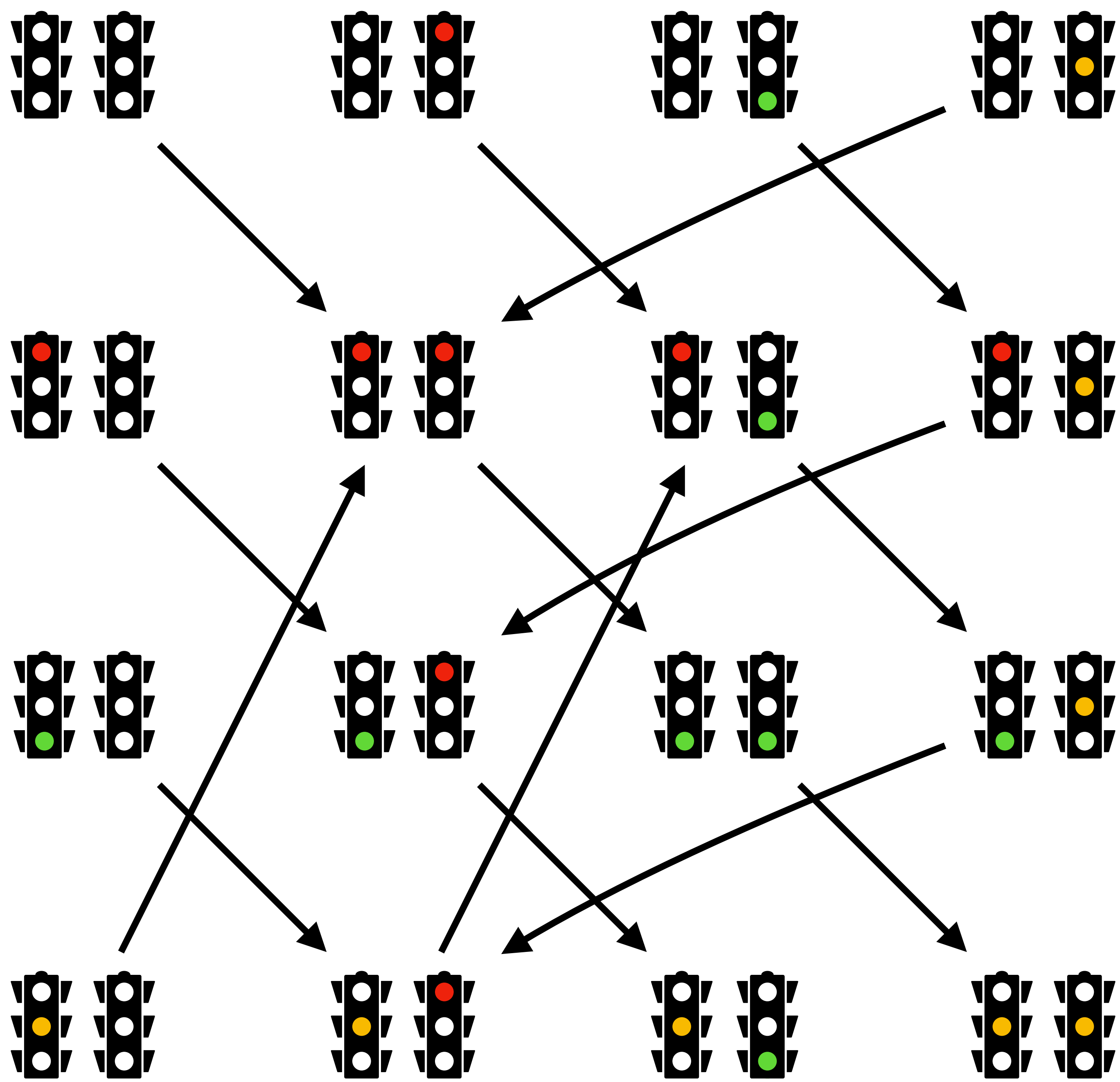
Traffic lights at a crossroads



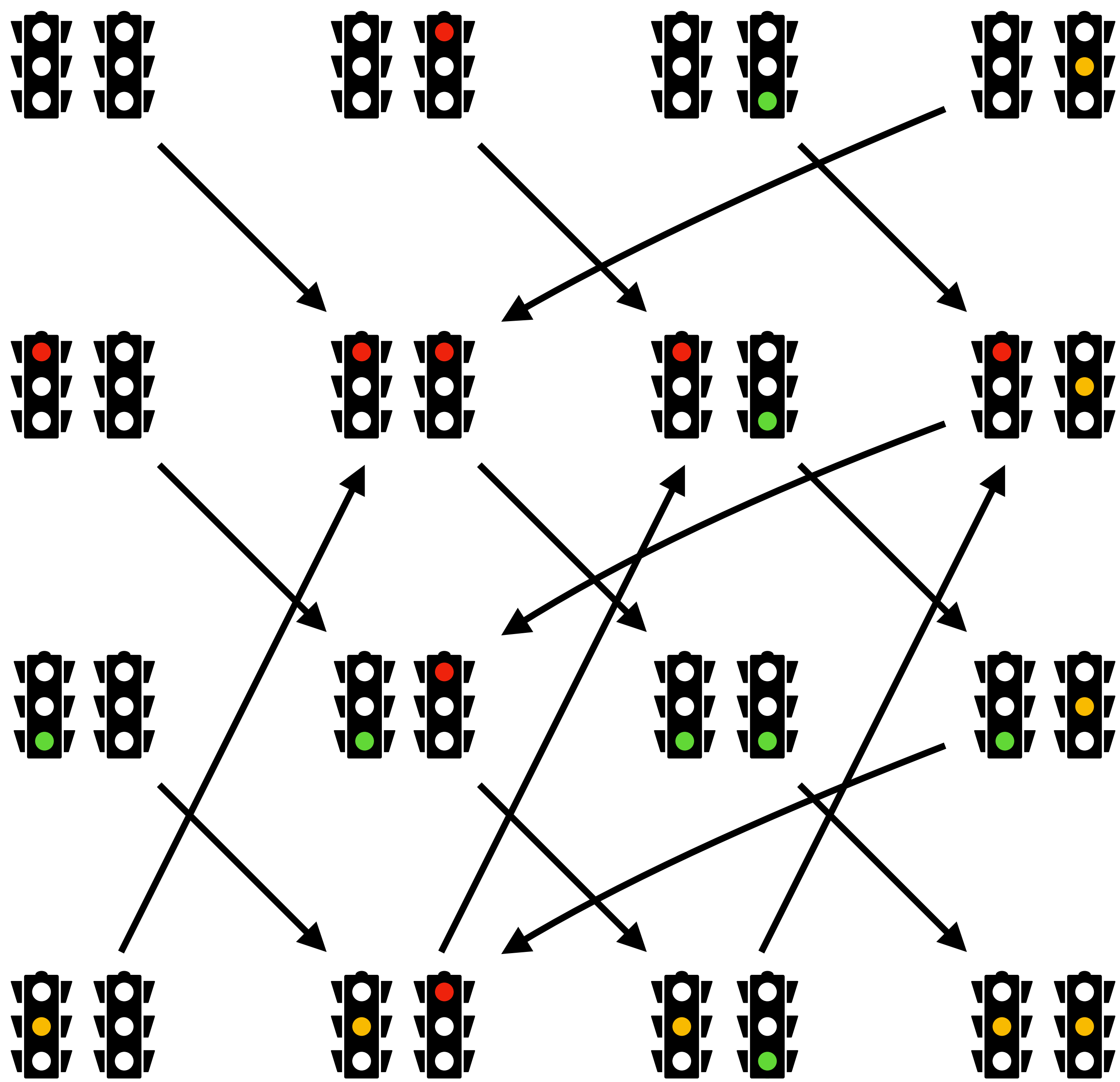
Traffic lights at a crossroads



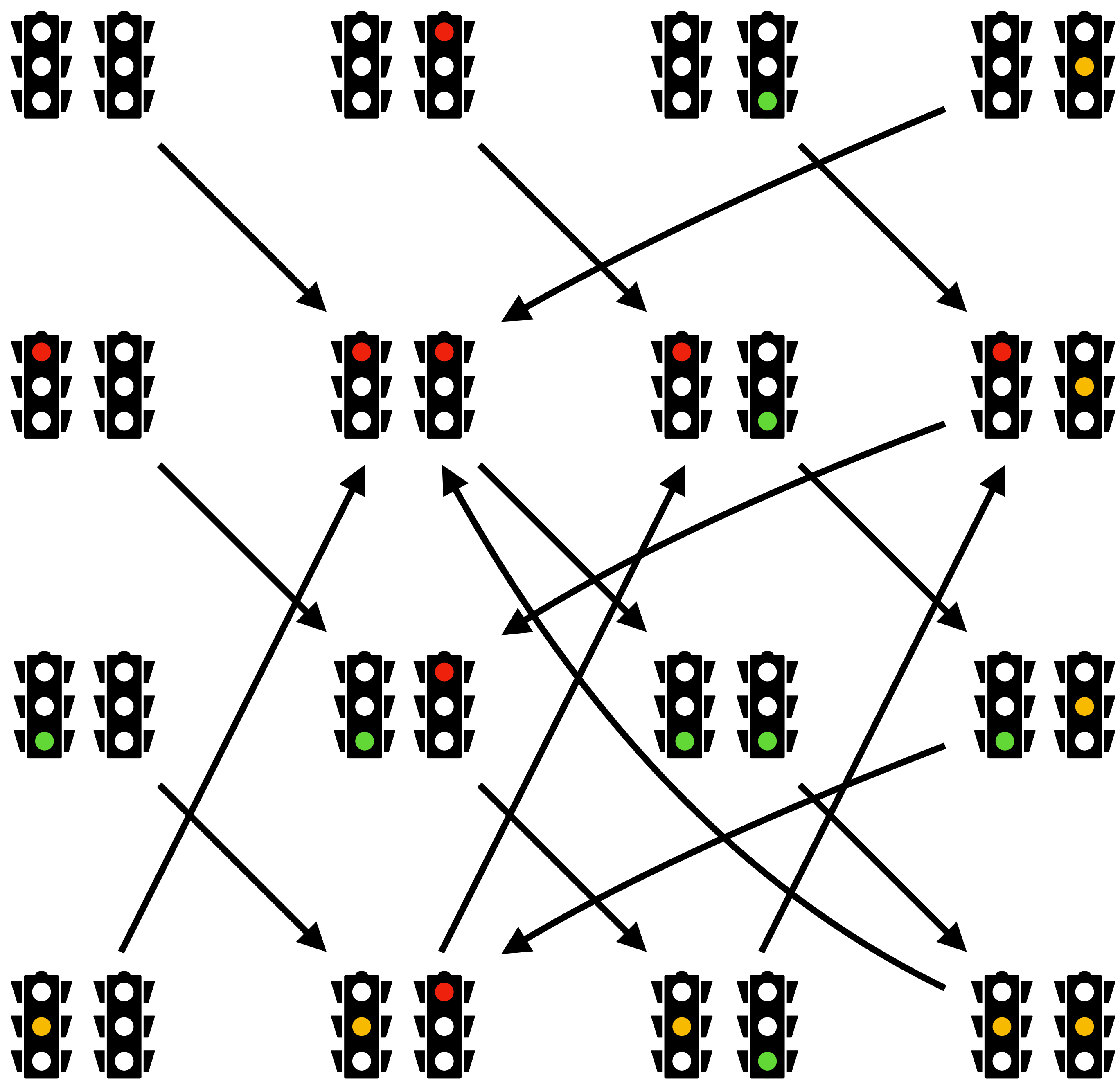
Traffic lights at a crossroads



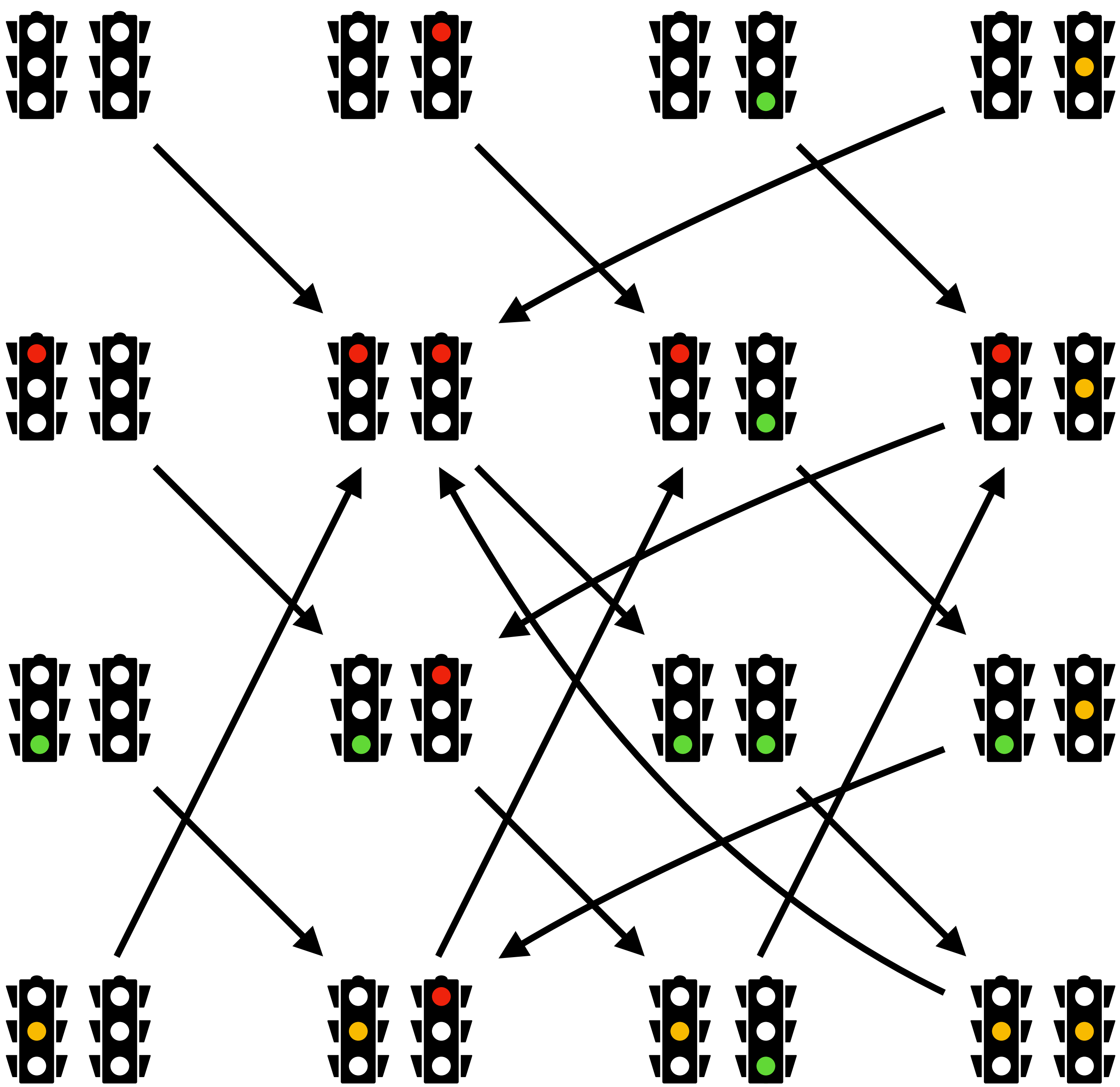
Traffic lights at a crossroads



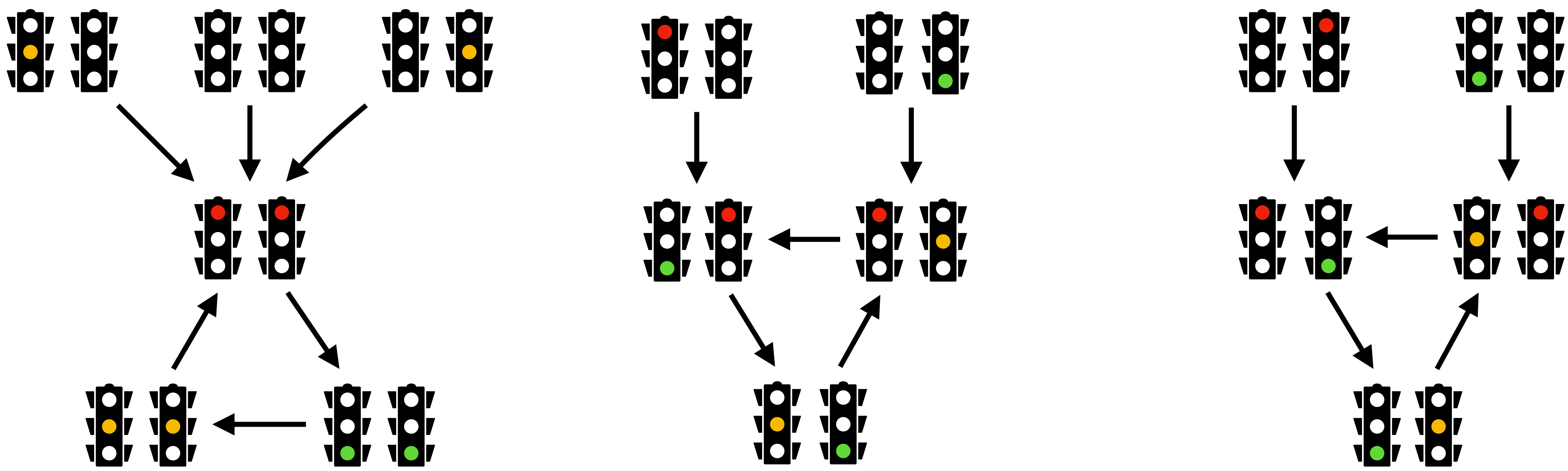
Traffic lights at a crossroads



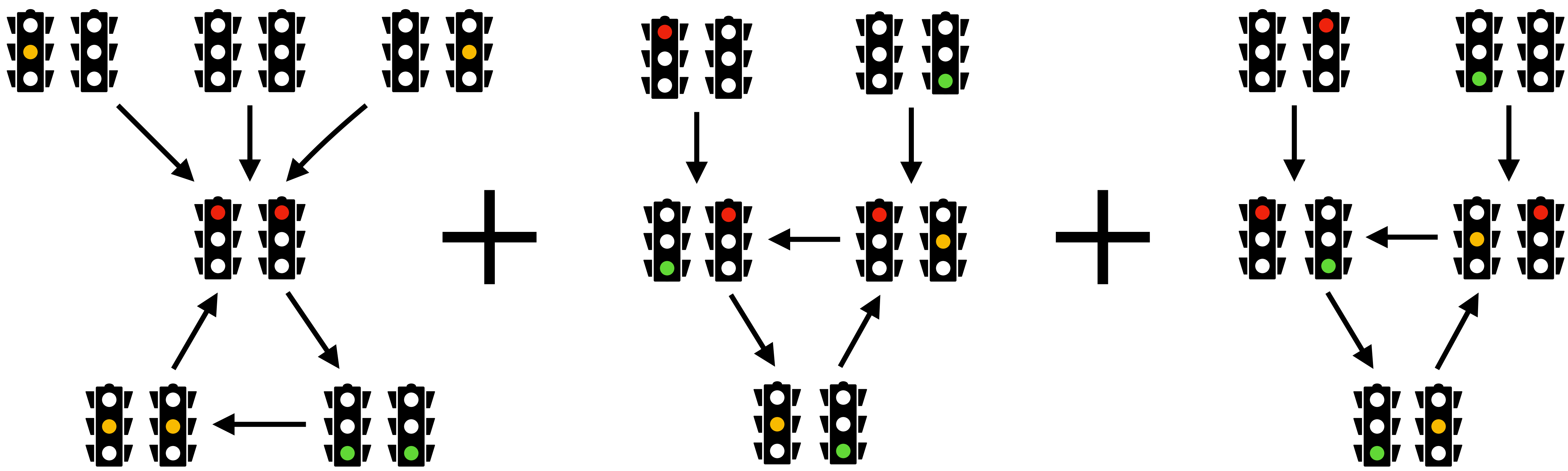
Traffic lights at a crossroads



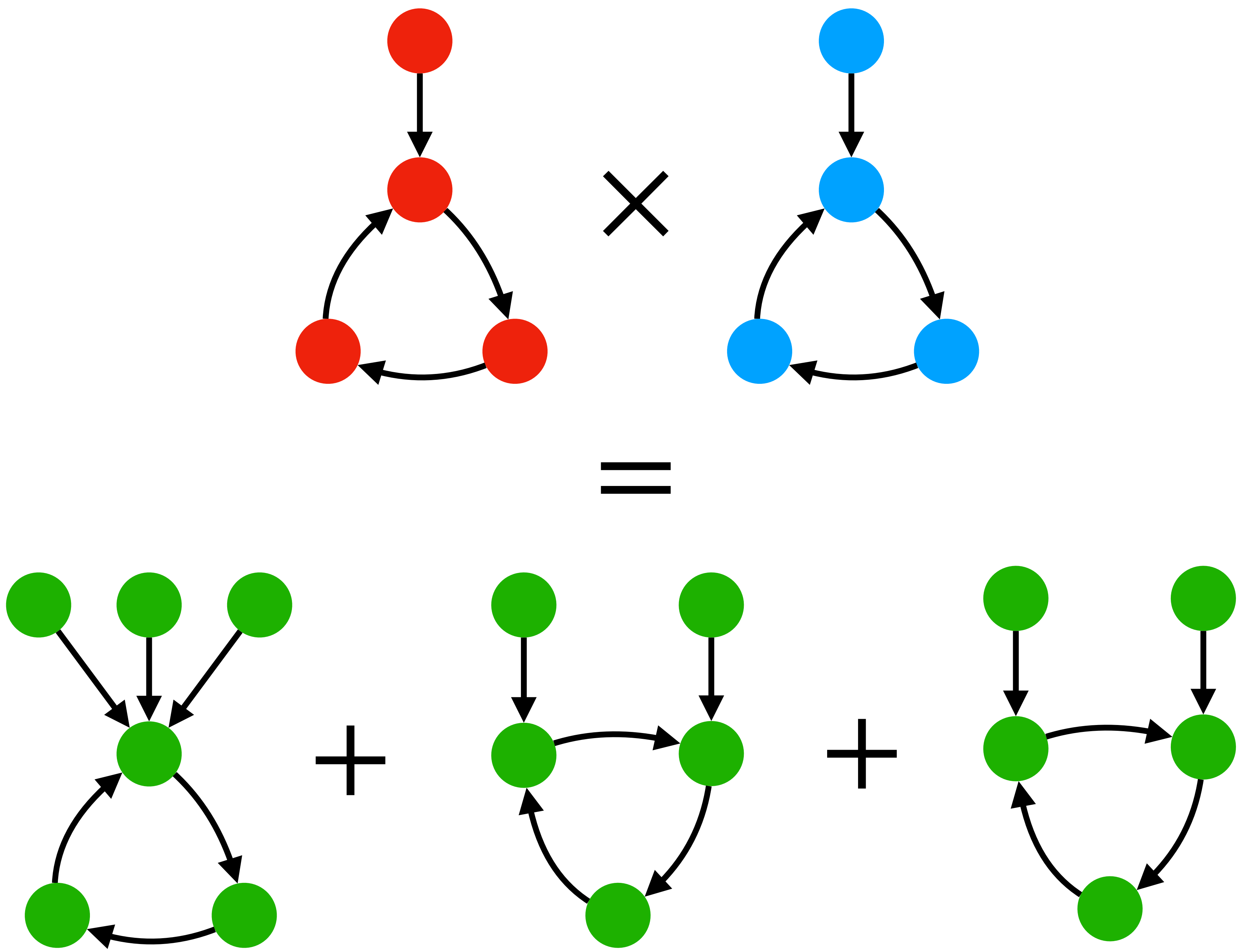
Traffic lights at a crossroads



Traffic lights at a crossroads



More abstractly…



$+$ and $\times$ behave as with nonnegative integers (a commutative semiring)

$+$ and $\times$ behave as with nonnegative integers (a commutative semiring)

- Commutative: $X + Y = Y + X$ and $X \times Y = Y \times X$

$+$ and $\times$ behave as with nonnegative integers (a commutative semiring)

- Commutative: $X + Y = Y + X$ and $X \times Y = Y \times X$
- Associative: $X + (Y + Z) = (Y + X) + Z$ and $X \times (Y \times Z) = (Y \times X) \times Z$

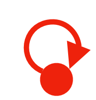
$+$ and $\times$ behave as with nonnegative integers (a commutative semiring)

- Commutative: $X + Y = Y + X$ and $X \times Y = Y \times X$
- Associative: $X + (Y + Z) = (Y + X) + Z$ and $X \times (Y \times Z) = (Y \times X) \times Z$
- Neutral elements: $\varnothing + X = X$ and  $\times X = X$

$+$ and $\times$ behave as with nonnegative integers (a commutative semiring)

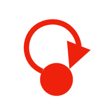
- Commutative: $X + Y = Y + X$ and $X \times Y = Y \times X$

- Associative: $X + (Y + Z) = (Y + X) + Z$ and $X \times (Y \times Z) = (Y \times X) \times Z$

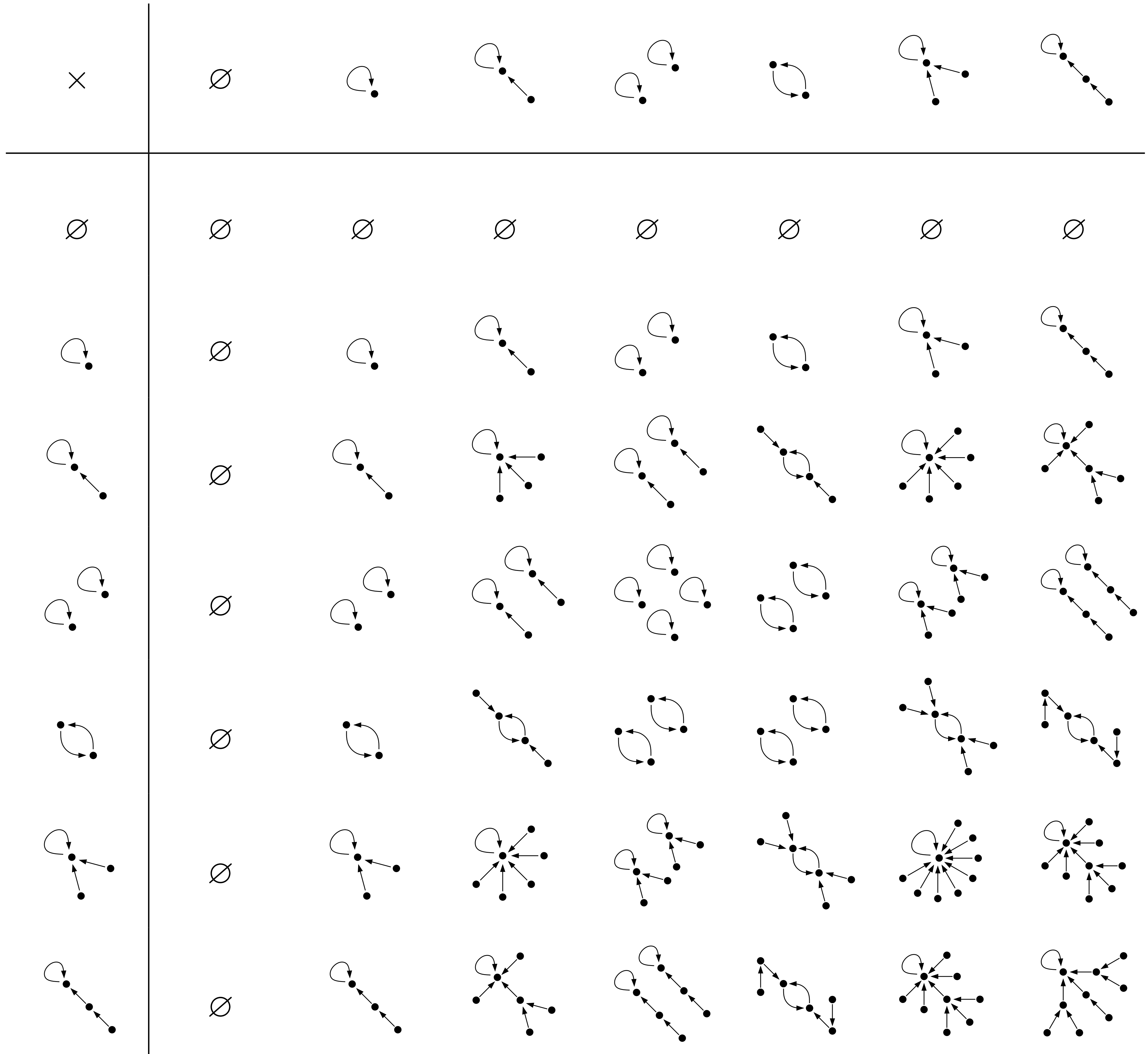
- Neutral elements: $\varnothing + X = X$ and  $\times X = X$

- Distributive: $X \times (Y + Z) = X \times Y + X \times Z$

$+$ and $\times$ behave as with nonnegative integers (a commutative semiring)

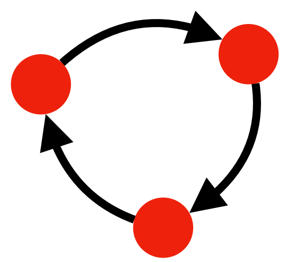
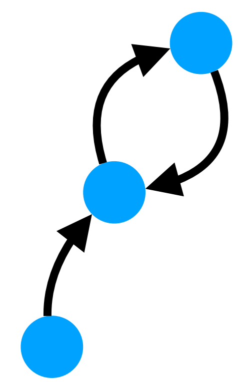
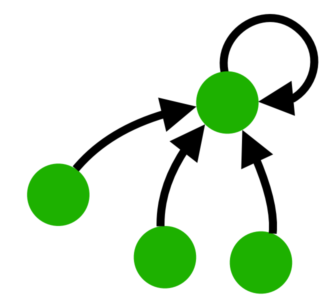
- Commutative: $X + Y = Y + X$ and $X \times Y = Y \times X$
- Associative: $X + (Y + Z) = (Y + X) + Z$ and $X \times (Y \times Z) = (Y \times X) \times Z$
- Neutral elements: $\varnothing + X = X$ and  $\times X = X$
- Distributive: $X \times (Y + Z) = X \times Y + X \times Z$
- Multiplication by zero: $\varnothing \times X = \varnothing$

Multiplication table

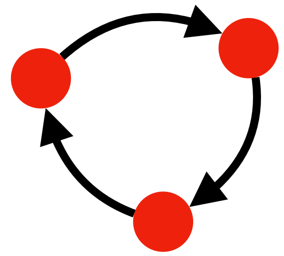
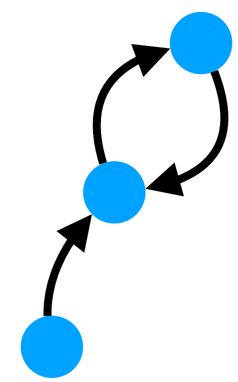
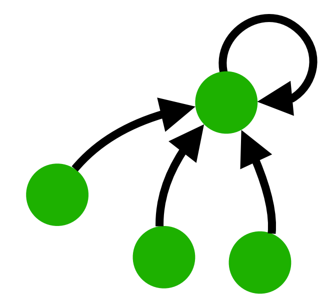


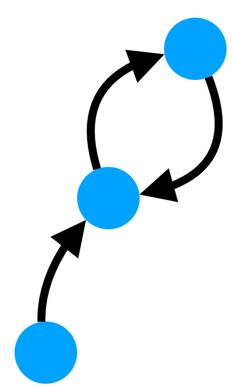
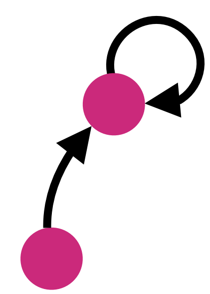
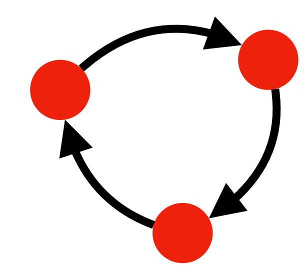
Equations for decomposing systems

Eqns over dynamical systems

 $X + Y^2 =$  $Z +$ 

Eqns over dynamical systems

 $X + Y^2 =$  $Z +$ 

$X =$  $Y =$  $Z =$ 

⚠️ Bad news about solving equations ⚠️

⚠️ Bad news about solving equations ⚠️

- There is no algorithm at all for solving general equations! (By reduction from Hilbert's 10th problem)

⚠️ Bad news about solving equations ⚠️

- There is no algorithm at all for solving general equations! (By reduction from Hilbert's 10th problem)
- Equations without variables on one side have an algorithm, but even linear ones of this form are hard to solve (**NP**-complete):

$$A_1X_1 + A_2X_2 + \cdots + A_nX_n = B$$

⚠️ Bad news about solving equations ⚠️

- There is no algorithm at all for solving general equations! (By reduction from Hilbert's 10th problem)

- Equations without variables on one side have an algorithm, but even linear ones of this form are hard to solve (**NP**-complete):

$$A_1X_1 + A_2X_2 + \cdots + A_nX_n = B$$

- We are still unsure if equations in one single variable, like $AX = B$, can be solved efficiently (conjecture: no)

Identifying basic building blocks

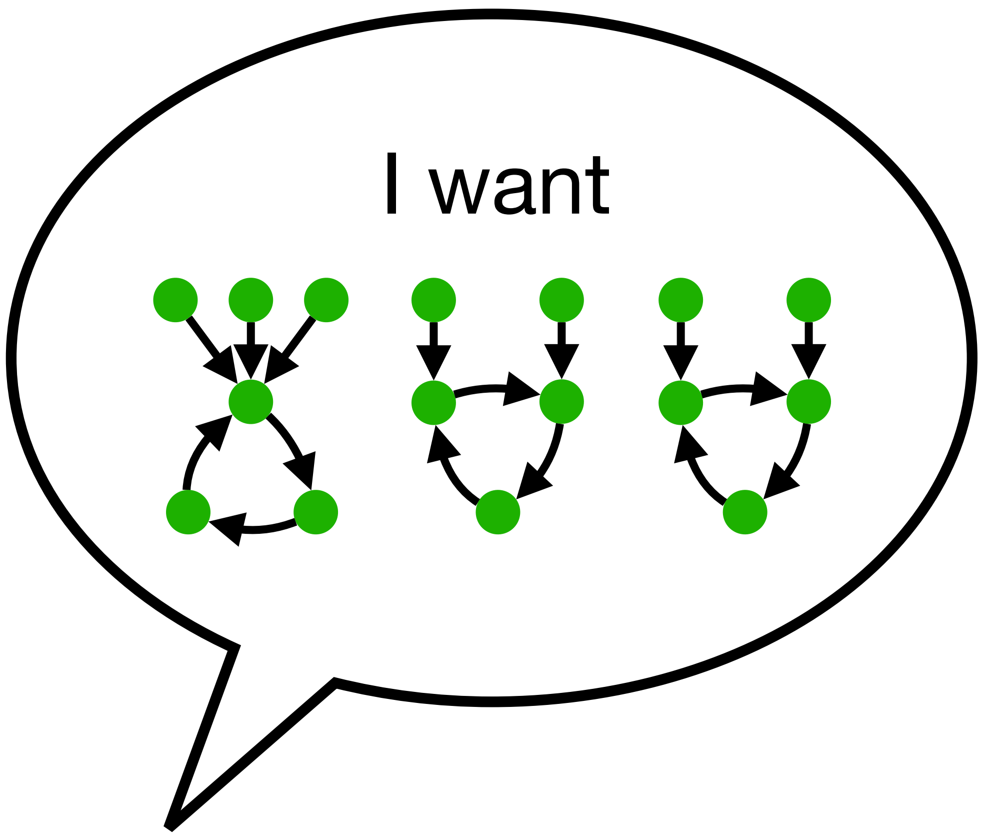
Scenario

DynaSys Inc.

Scenario

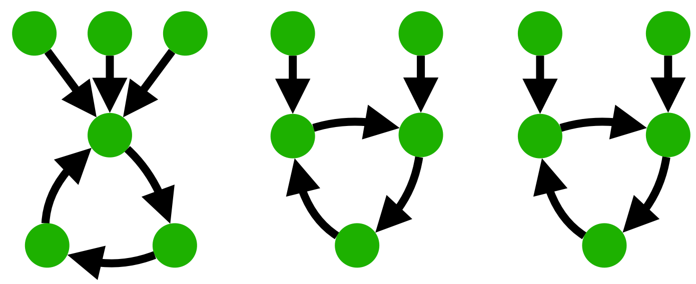
DynaSys Inc.

Scenario




DynaSys Inc.

Scenario



$$= X \times Y + Z$$

DynaSys Inc.

Scenario

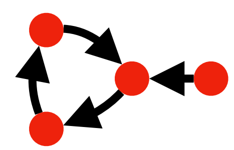
$$X =$$

$$Y =$$

$$Z =$$


DynaSys Inc.

Scenario

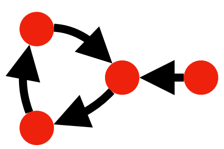
$X =$ 

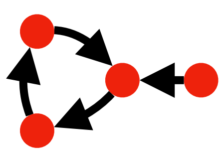
$Y =$

$Z =$

DynaSys Inc.

Scenario

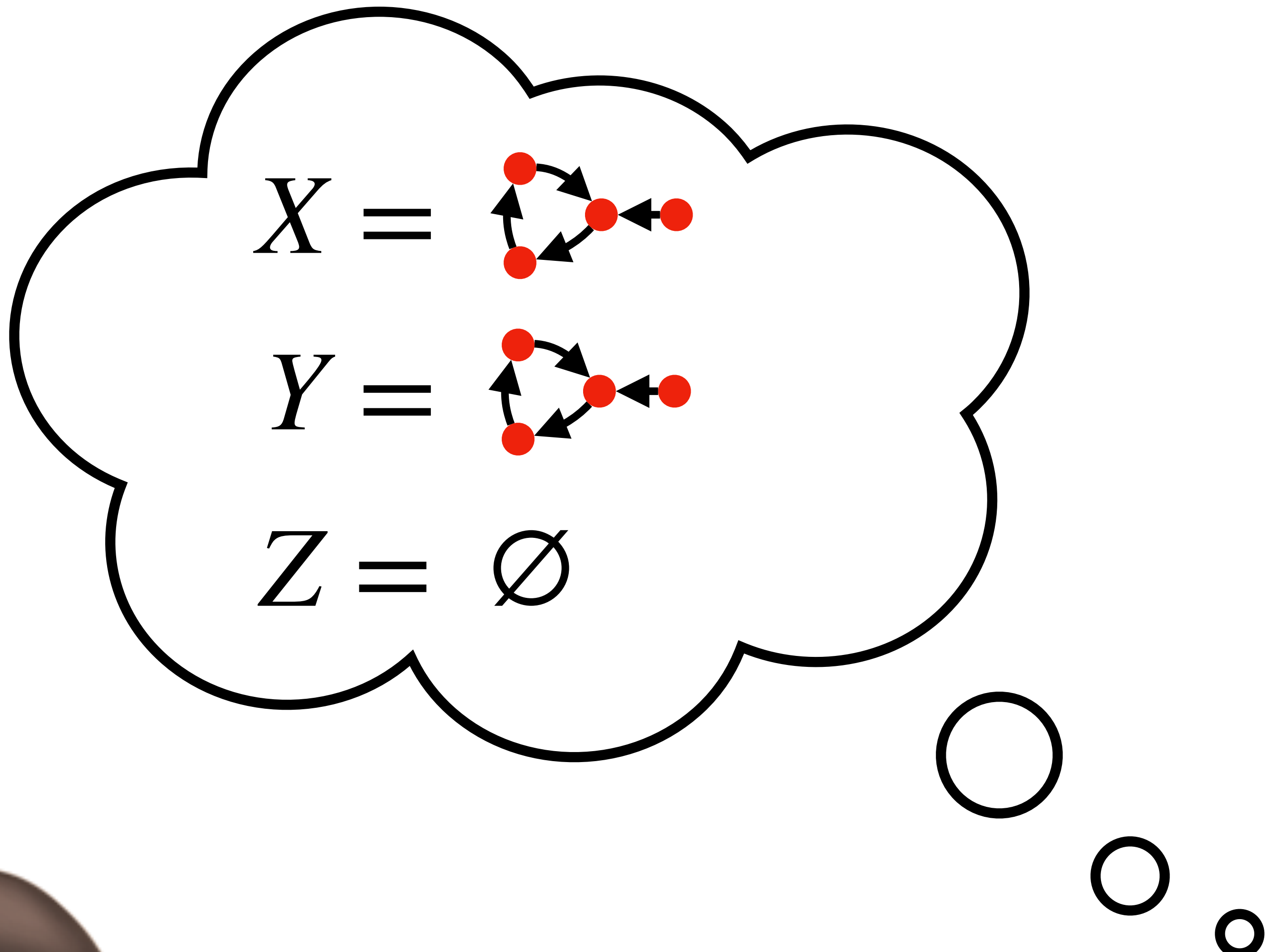
$X =$ 

$Y =$ 

$Z =$

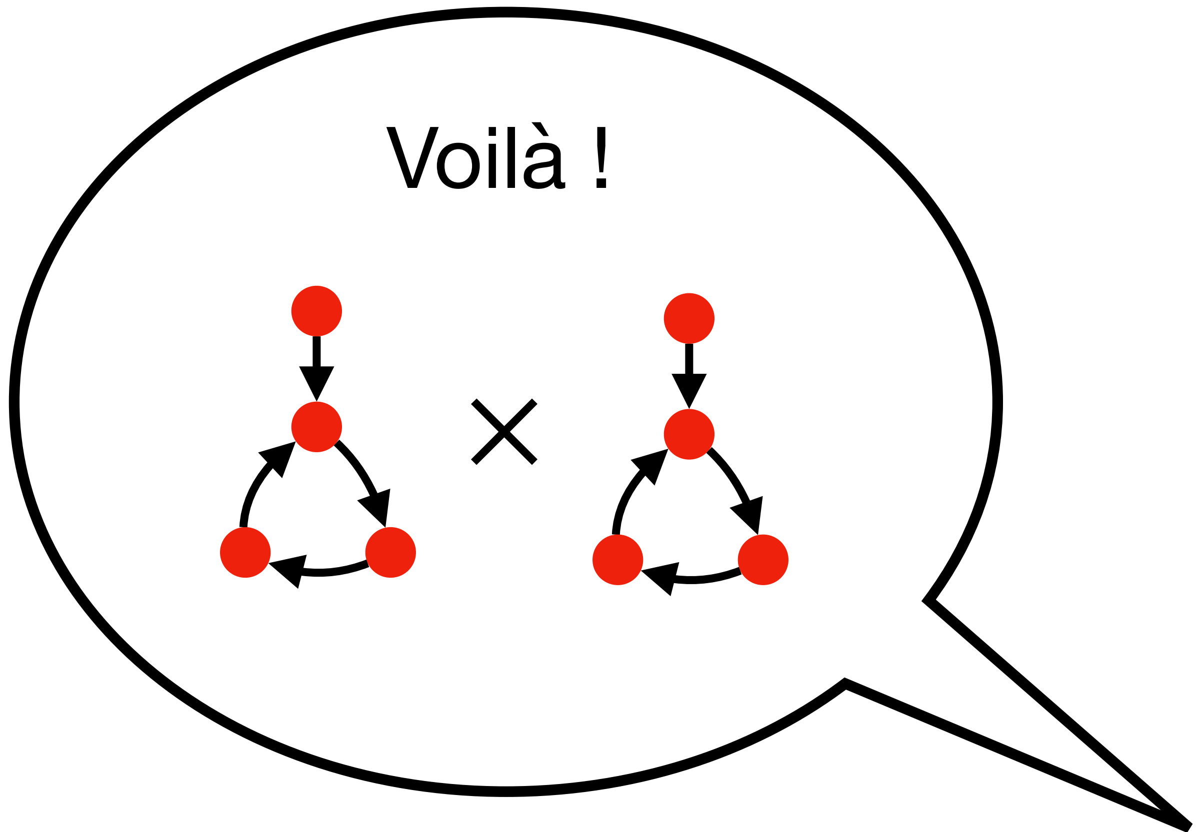
DynaSys Inc.

Scenario



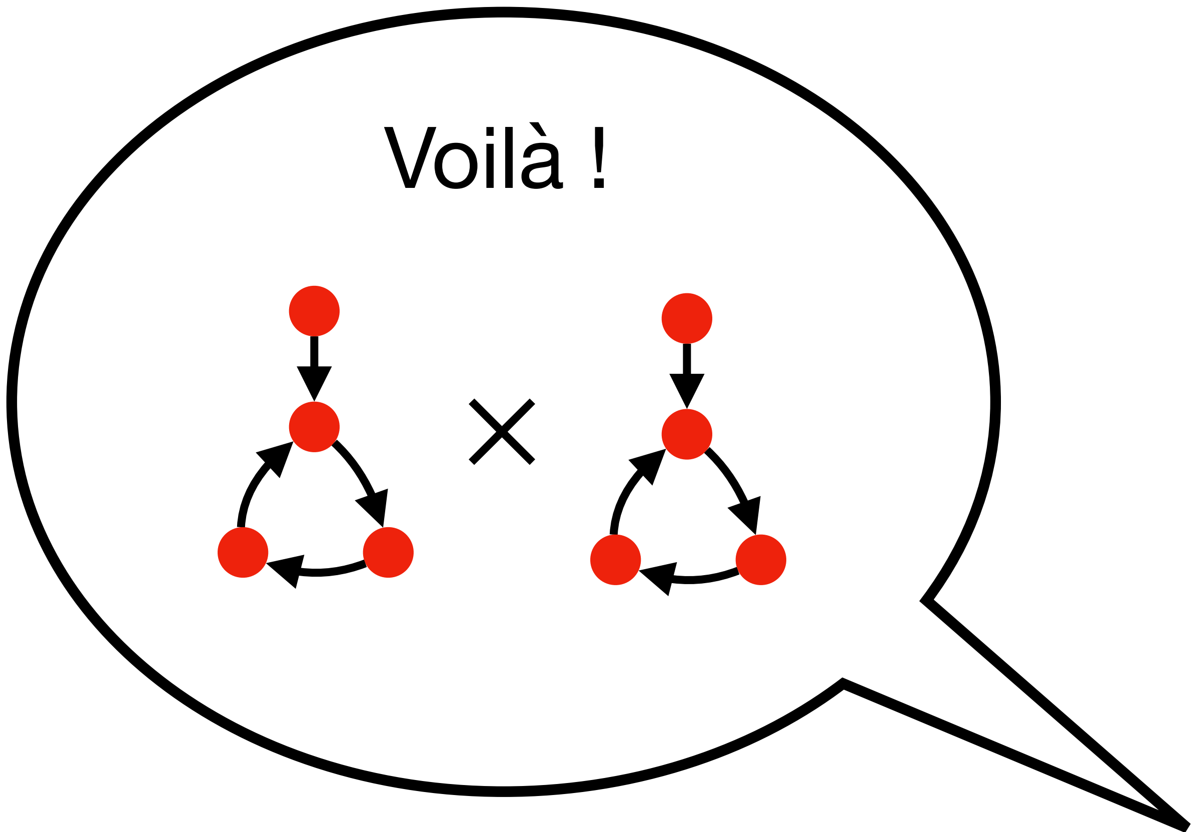
DynaSys Inc.

Scenario



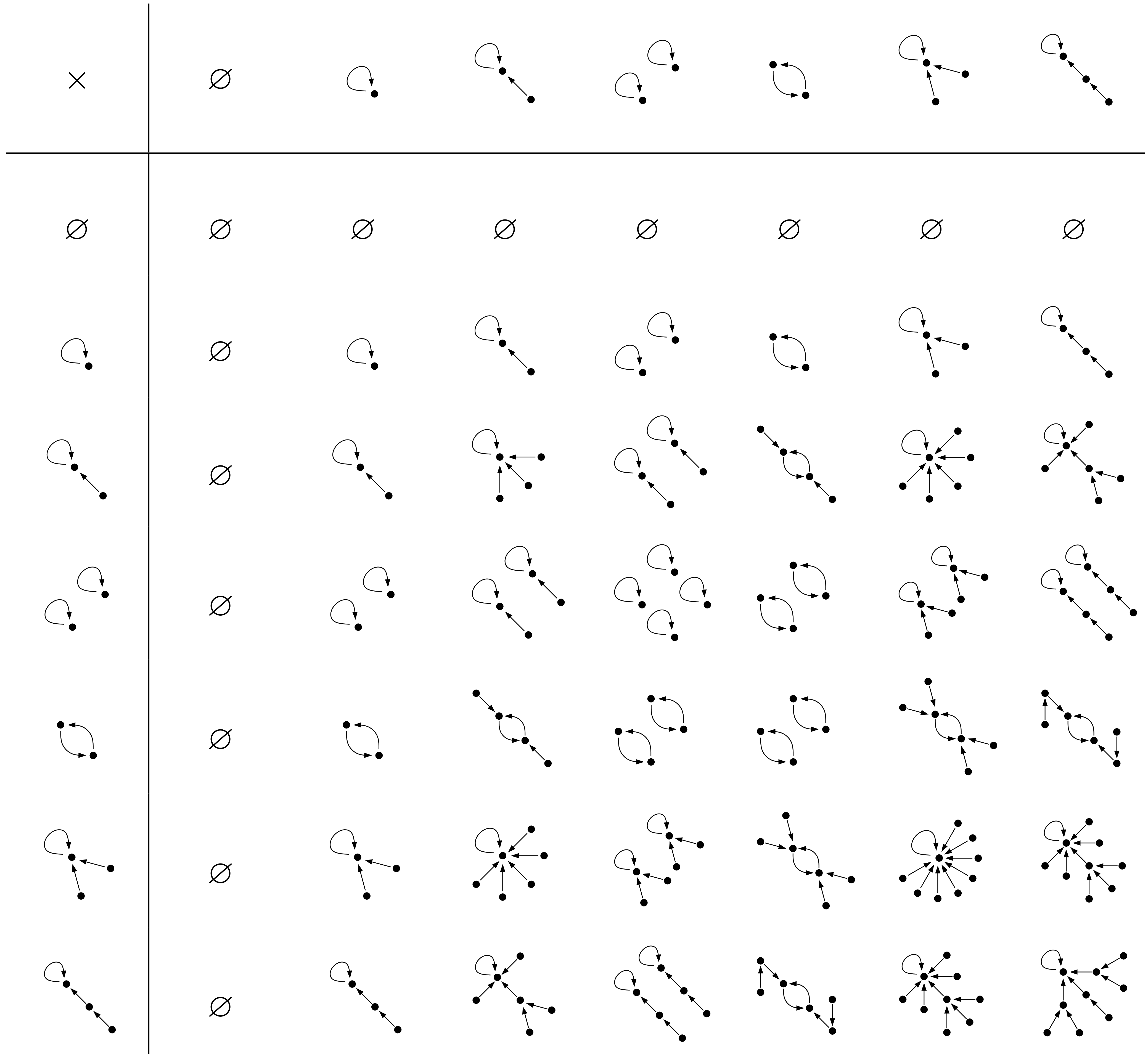
DynaSys Inc.

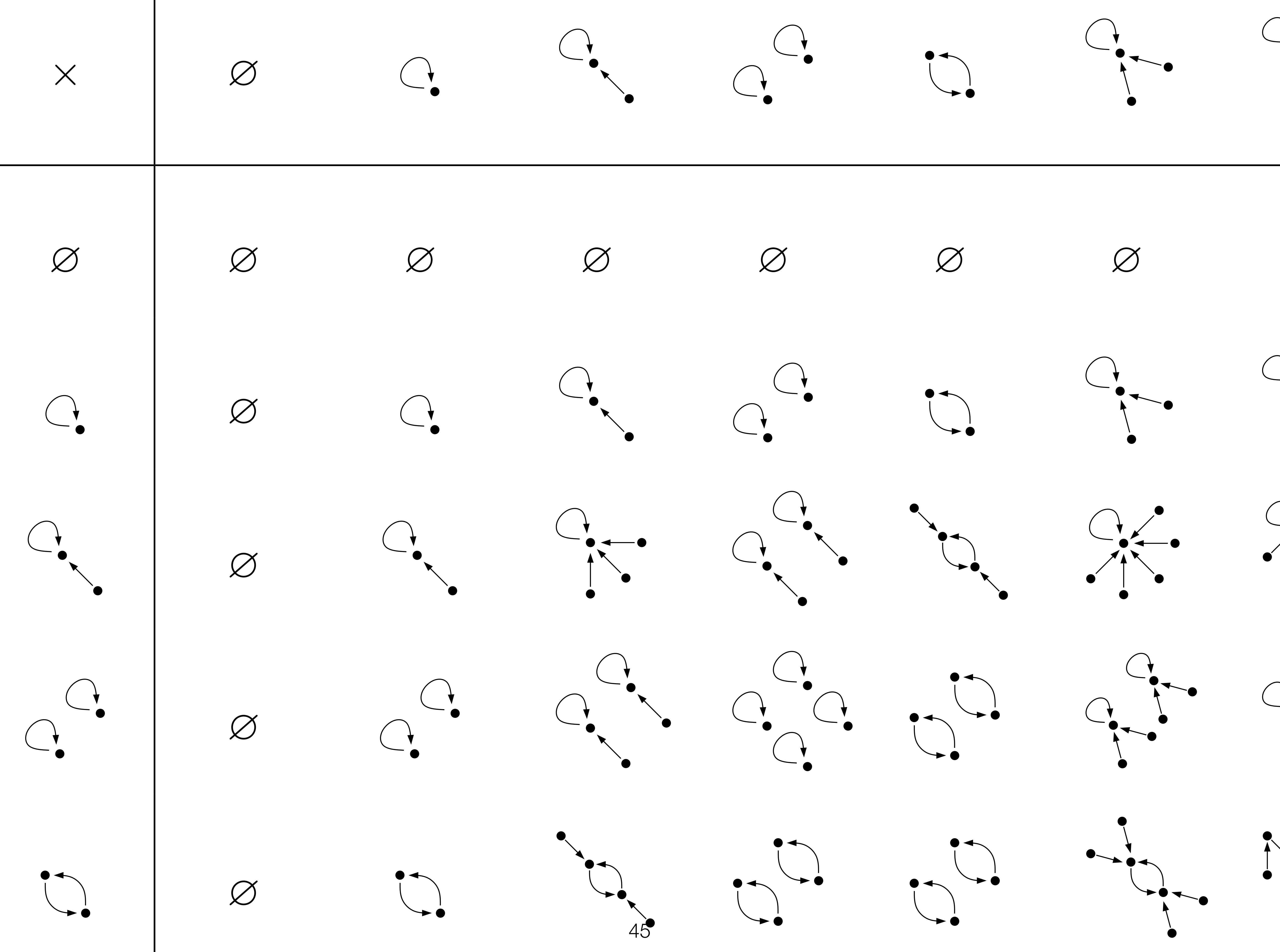
Scenario

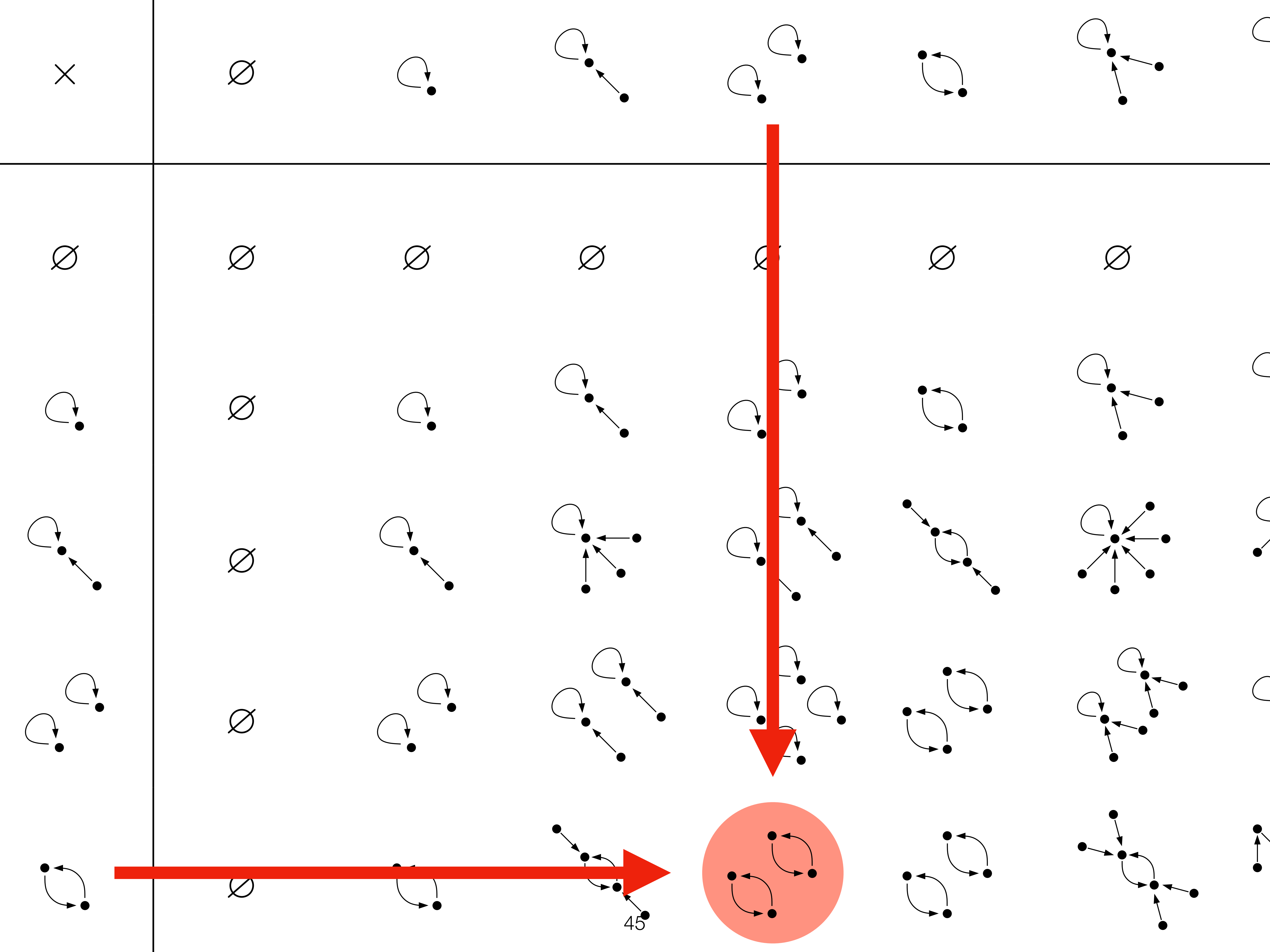


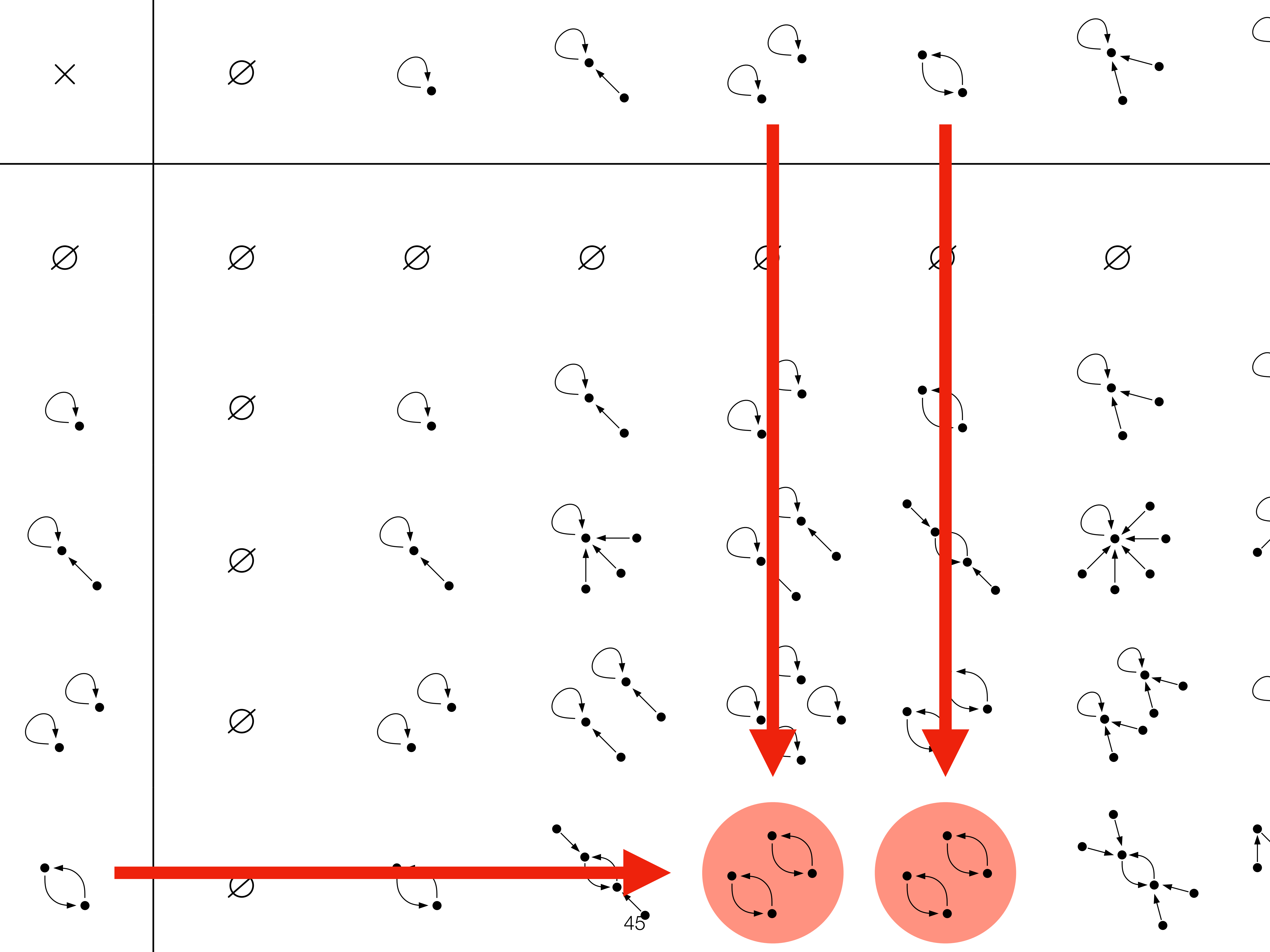
DynaSys Inc.

No unique factorisation into irreducibles!

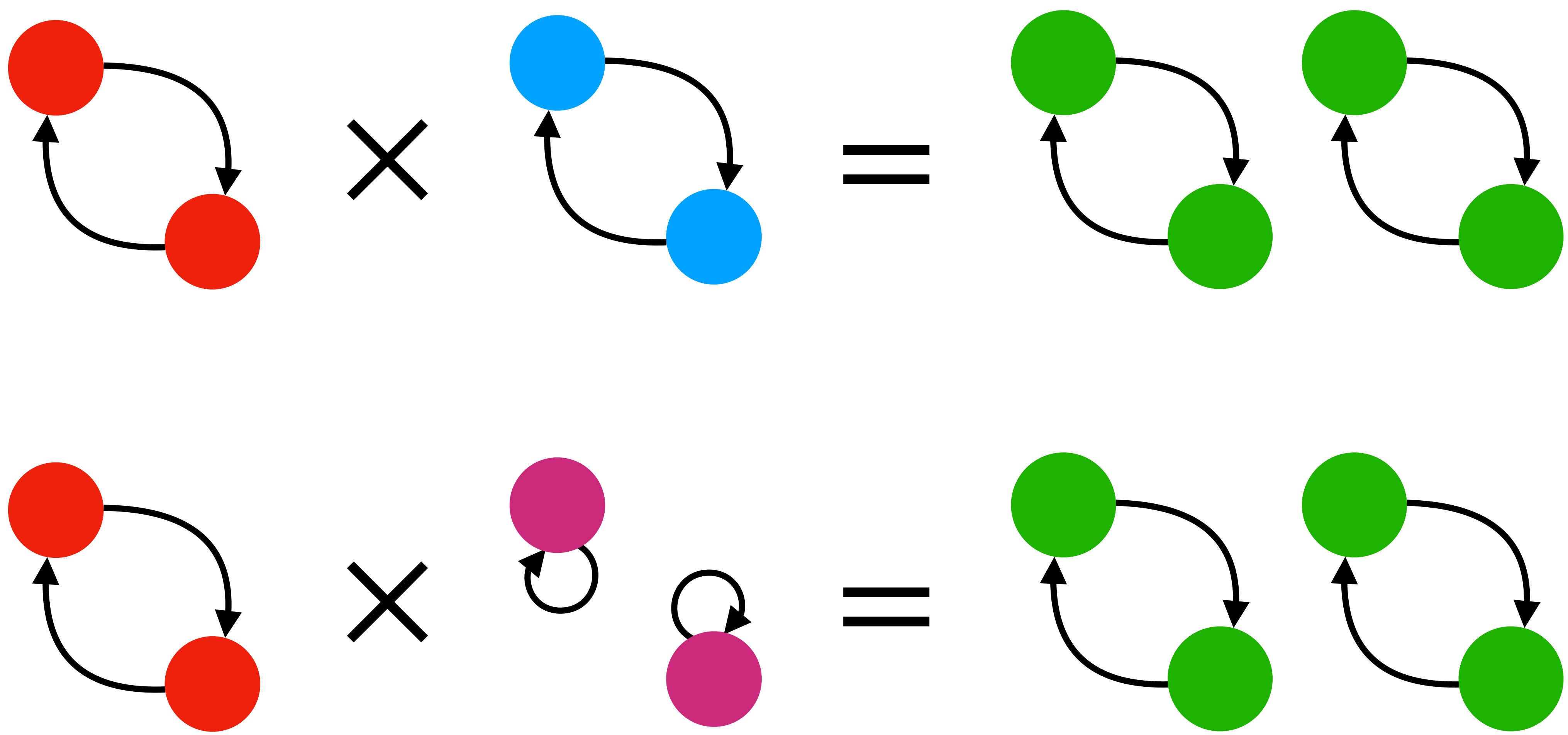




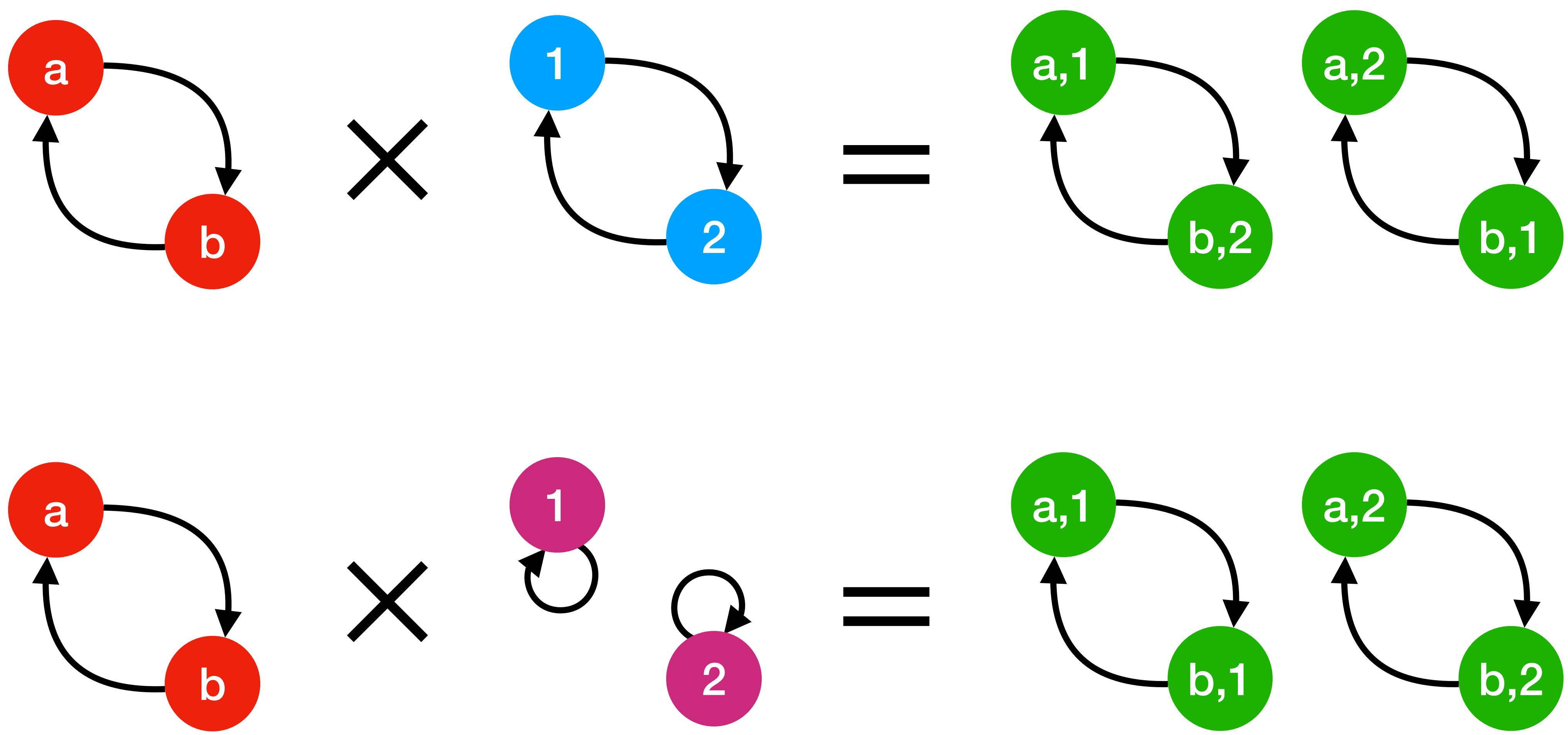




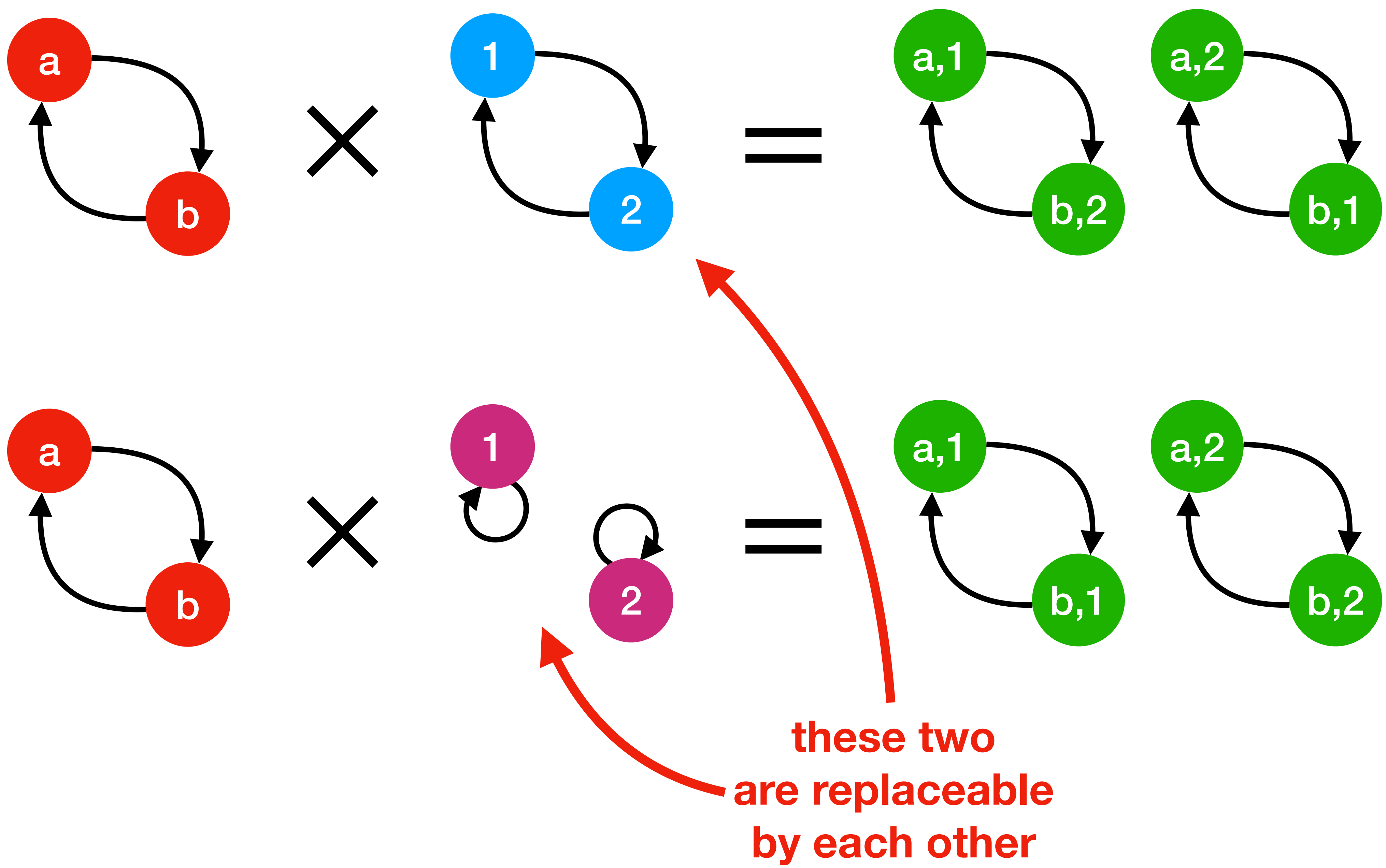
Multiple factorisations



Multiple factorisations



Multiple factorisations



What do we need in our warehouse?

What do we need in our warehouse?

- Decompose our system A as a product of smaller, irreducible systems $A = A_1 \times A_2 \times \cdots \times A_n$

What do we need in our warehouse?

- Decompose our system A as a product of smaller, irreducible systems $A = A_1 \times A_2 \times \cdots \times A_n$

- But in general there are multiple ways of doing that: $A = B_1 \times B_2 \times \cdots \times B_m$

What do we need in our warehouse?

- Decompose our system A as a product of smaller, irreducible systems $A = A_1 \times A_2 \times \cdots \times A_n$
- But in general there are multiple ways of doing that: $A = B_1 \times B_2 \times \cdots \times B_m$
- So $A_1, A_2, \ldots, A_n$ might all be replaceable

Prime systems

Prime systems

- A prime is a system P such that, whenever it appears in a factorisation into irreducibles of $A \times B$, it appears in the factorisation of either A or B

Prime systems

- A prime is a system P such that, whenever it appears in a factorisation into irreducibles of $A \times B$, it appears in the factorisation of either A or B

- In other words, if P divides $A \times B$ then it divides A or B

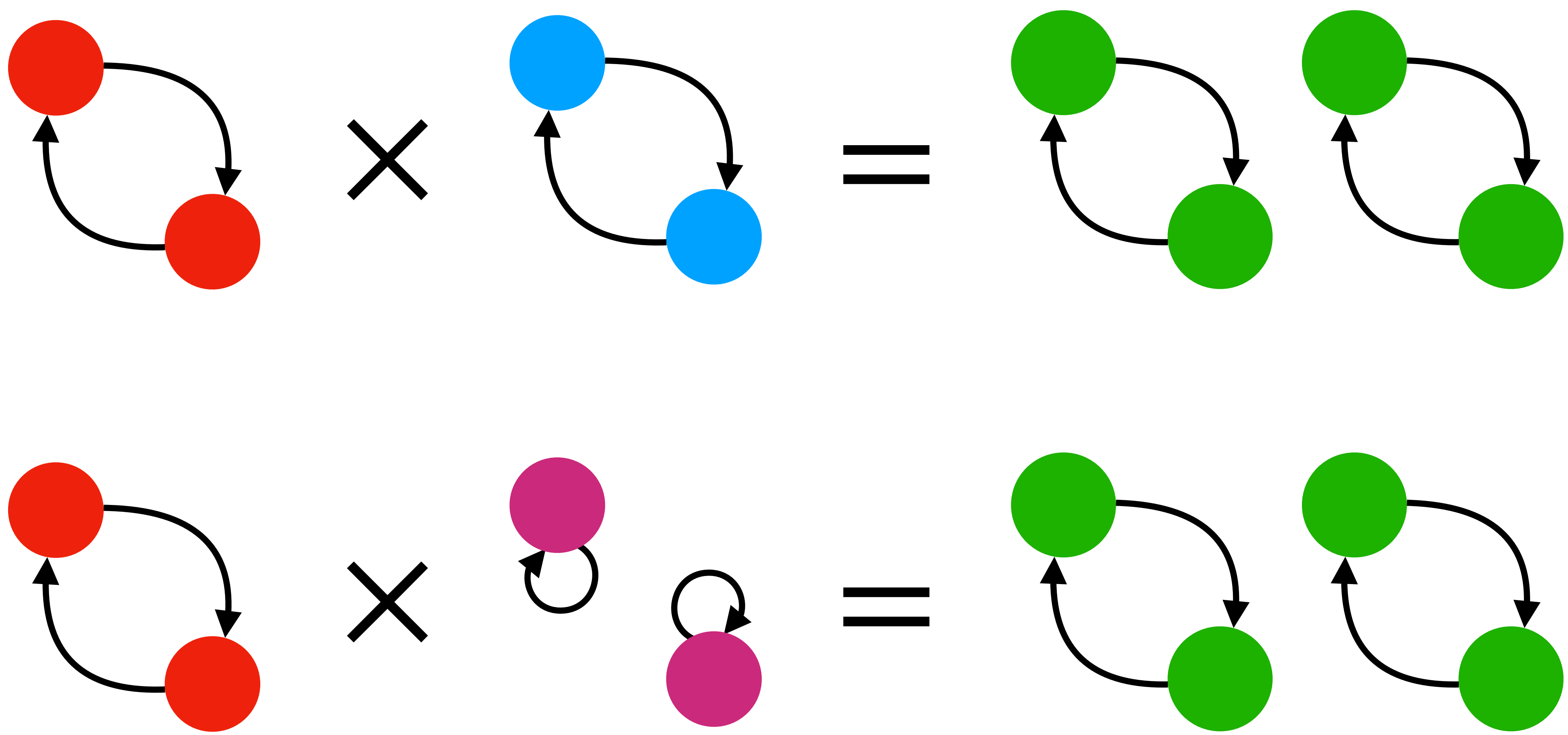
Prime systems

- A prime is a system P such that, whenever it appears in a factorisation into irreducibles of $A \times B$, it appears in the factorisation of either A or B
- In other words, if P divides $A \times B$ then it divides A or B
- If a prime appears in one factorisation of a system, then it appears in all the others as well (it is irreplaceable)

Prime systems

- A prime is a system P such that, whenever it appears in a factorisation into irreducibles of $A \times B$, it appears in the factorisation of either A or B
- In other words, if P divides $A \times B$ then it divides A or B
- If a prime appears in one factorisation of a system, then it appears in all the others as well (it is irreplaceable)
- So we want at least all primes in our warehouse (even though that's not enough…)

Primes are not sufficient



Finding primes

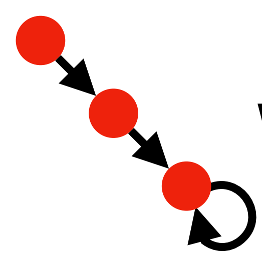
Finding primes

- We haven't been able to find even a single prime yet!

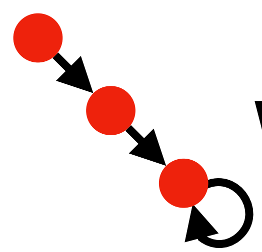
Finding primes

- We haven't been able to find even a single prime yet!
- We have found infinitely many non-primes though

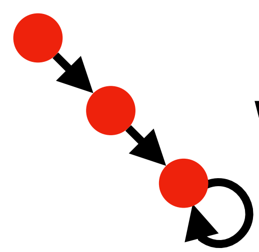
Finding primes

- We haven't been able to find even a single prime yet!
- We have found infinitely many non-primes though
- This guy here?  We don't know if it's prime or not!

Finding primes

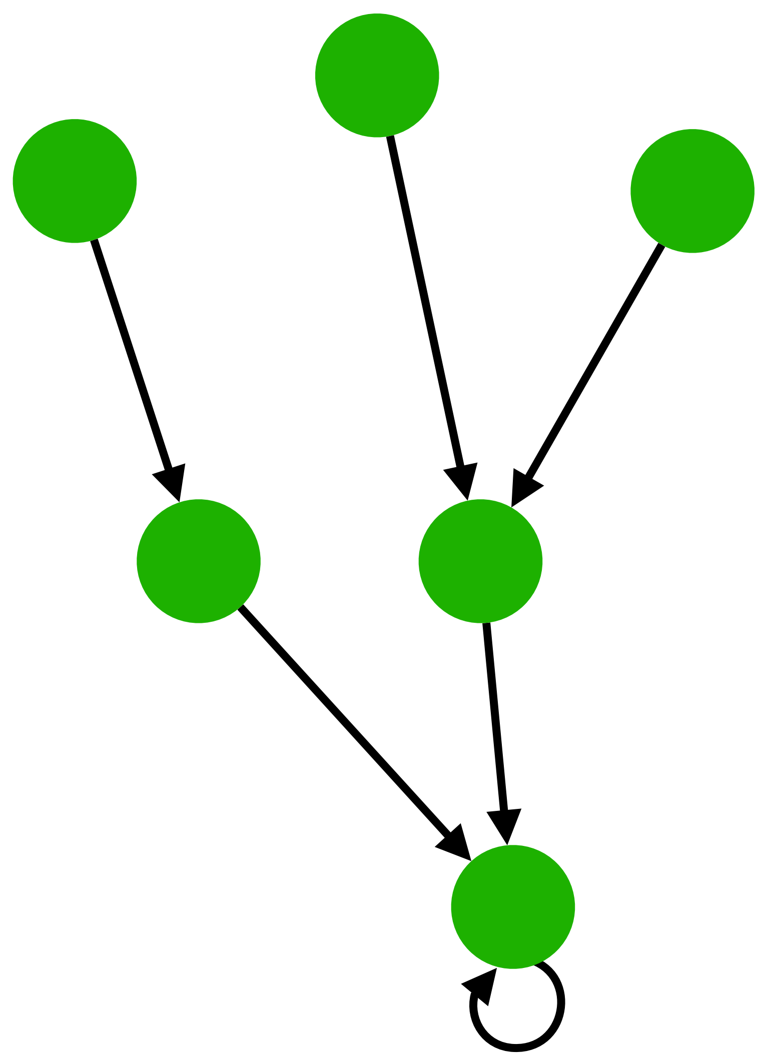
- We haven't been able to find even a single prime yet!
- We have found infinitely many non-primes though
- This guy here?  We don't know if it's prime or not!
- A counterexample to the primality of P is two systems A, B such that P divides $A \times B$ but neither A nor B

Finding primes

- We haven't been able to find even a single prime yet!
- We have found infinitely many non-primes though
- This guy here?  We don't know if it's prime or not!
- A counterexample to the primality of P is two systems A, B such that P divides $A \times B$ but neither A nor B
- Those A and B can be much bigger than P,
  but we don't know how much, so no algorithm yet…

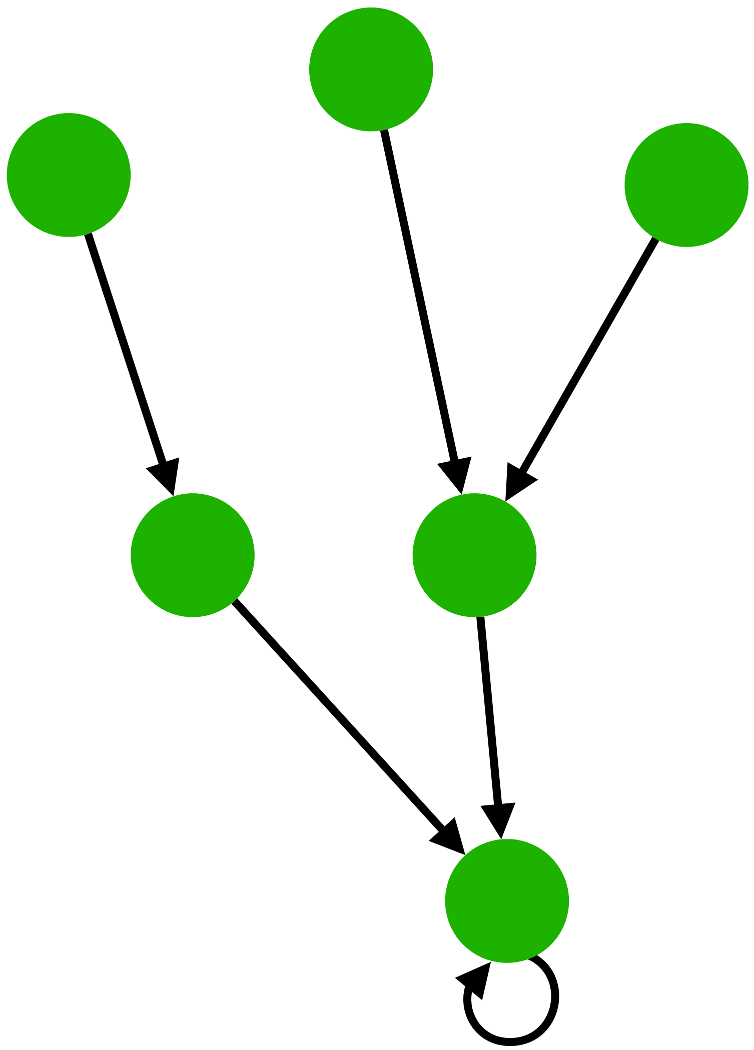
What do primes look like, if they exist at all?

What do primes look like, if they exist at all?



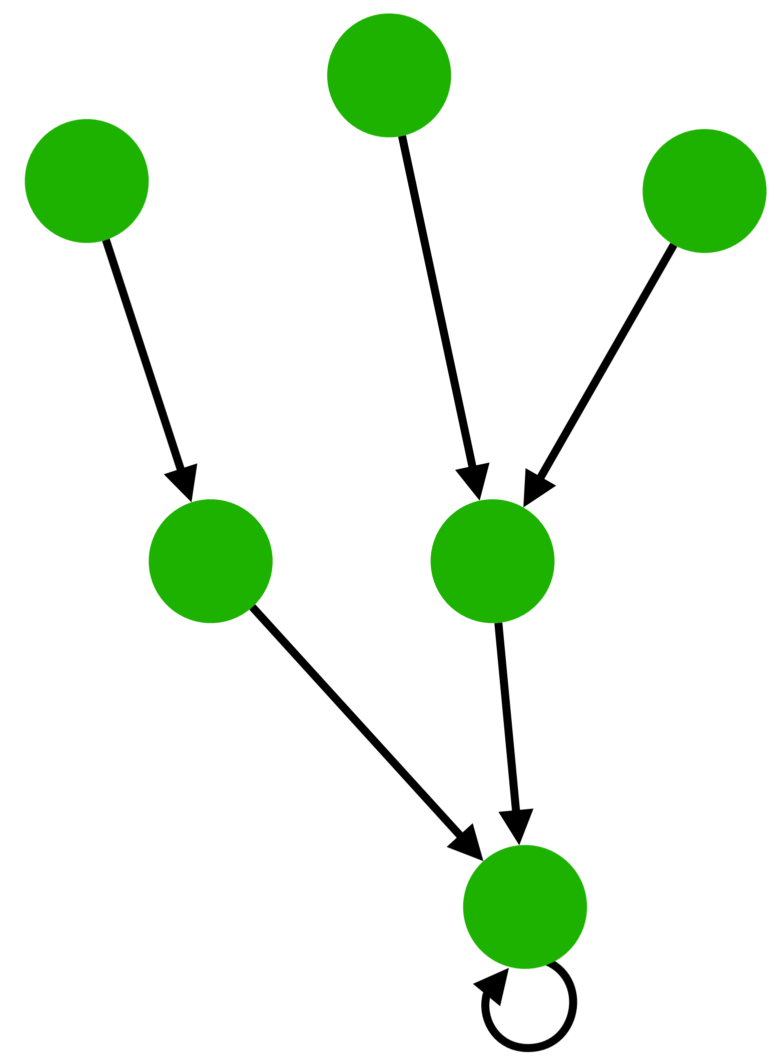
What do primes look like, if they exist at all?

- Connected



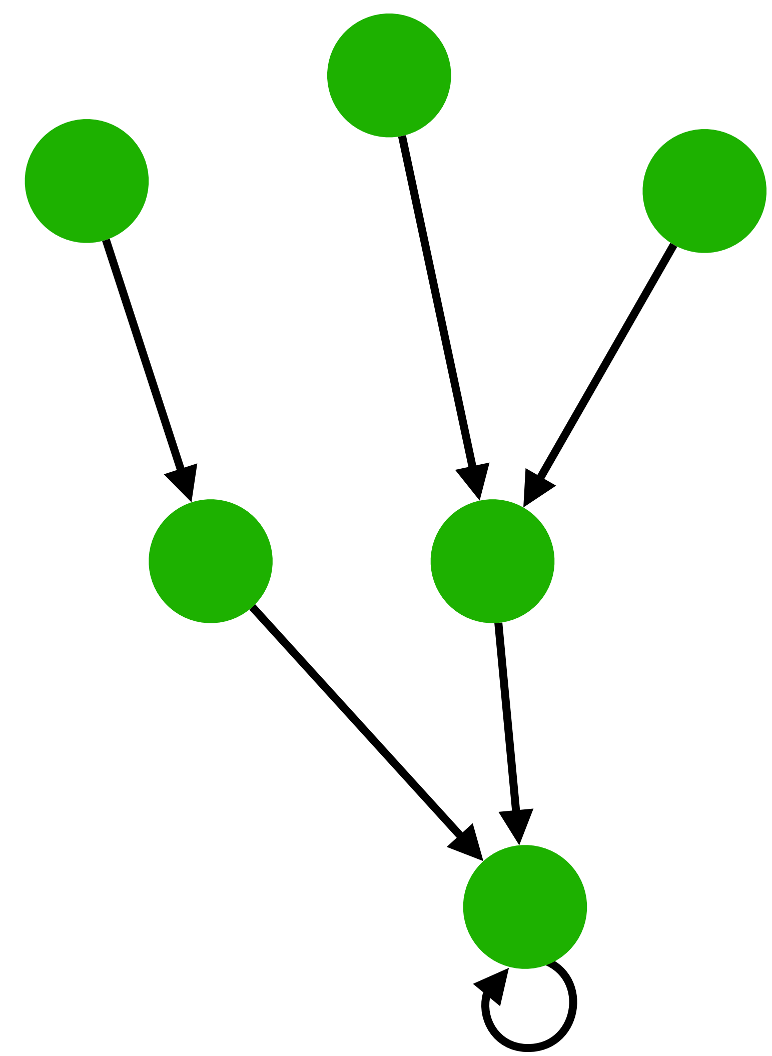
What do primes look like, if they exist at all?

- Connected
- Fixed point (no cycles of length > 1)



What do primes look like, if they exist at all?

- Connected
- Fixed point (no cycles of length > 1)
- gcd of the number of predecessors across all states must be 1



Future developments

Future developments

- Find more solvable equations, and at least one class of equations that is solvable efficiently

Future developments

- Find more solvable equations, and at least one class of equations that is solvable efficiently

- How many solutions for a given equation? E.g., $AX = B$ has at most one if B is connected (new result!)

Future developments

- Find more solvable equations, and at least one class of equations that is solvable efficiently
- How many solutions for a given equation? E.g., $AX = B$ has at most one if B is connected (new result!)
- An enumeration algorithm for dynamical systems up to isomorphism, to find examples and counterexamples (almost done)

Future developments

- Find more solvable equations, and at least one class of equations that is solvable efficiently

- How many solutions for a given equation? E.g., $AX = B$ has at most one if B is connected (new result!)

- An enumeration algorithm for dynamical systems up to isomorphism, to find examples and counterexamples (almost done)

- Find out if prime systems exist, or at least find a primality algorithm!

Bibliography

- A. Dennunzio, V. Dorigatti, E. Formenti, L. Manzoni, A.E. Porreca, Polynomial equations over finite, discrete-time dynamical systems

- F. Doré, E. Formenti, A.E. Porreca, S. Riva, Algorithmic reconstruction of discrete dynamics (and its bibliography)

- É. Naquin, M. Gadouleau, Factorisation in the semiring of finite dynamical systems (new and fresh)

- A.E. Porreca, E. Timofeeva, Efficient generation of functional digraphs up to isomorphism (soon 🤞)

Thanks for your attention!

Any questions?