Pebble games
with algebraic rules

Bjarki Holm

(joint work with Anuj Dawar)

UNIVERSITY OF CAMBRIDGE

The 39th International Colloquium on Automata, Languages & Programming

University of Warwick

2012

Overview

- Logical equivalence relations between finite structures.
- Connection between equivalence relations and games.
- Stronger equivalence relations obtained by new types of game.

Logical equivalence of finite structures

Two relational structures are said to be elementarily equivalent if they agree on all sentences of first-order logic

Over finite structures: elementary equivalence = isomorphism

(here: focus only on finite graphs)

Motivation

Obtain polynomial-time decidable approximations of elementary equivalence that approach isomorphism in the limit.

Applications

- descriptive complexity theory (finding a logic for PTIME)
- families of polynomial-time algorithms for graph isomorphism

Relaxations of elementary equivalence

Common approach: study suitable restrictions of first-order logic

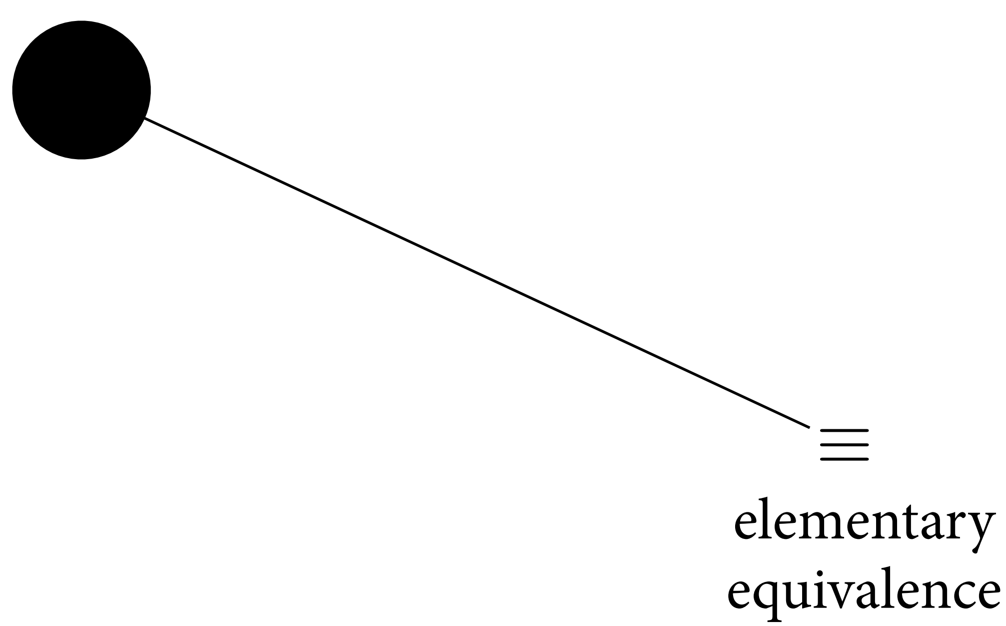
	Equivalence
$\equiv$	Elementary equivalence
$\equiv^{C^k}$	k-variable FO with counting quantifiers $\exists^{\geq i} x \,.\, \varphi(x)$
$\equiv^{L^k}$	First-order logic with variables $x_1, \ldots, x_k$
$\equiv_r$	First-order logic up to quantifier rank r

decidable in polynomial time

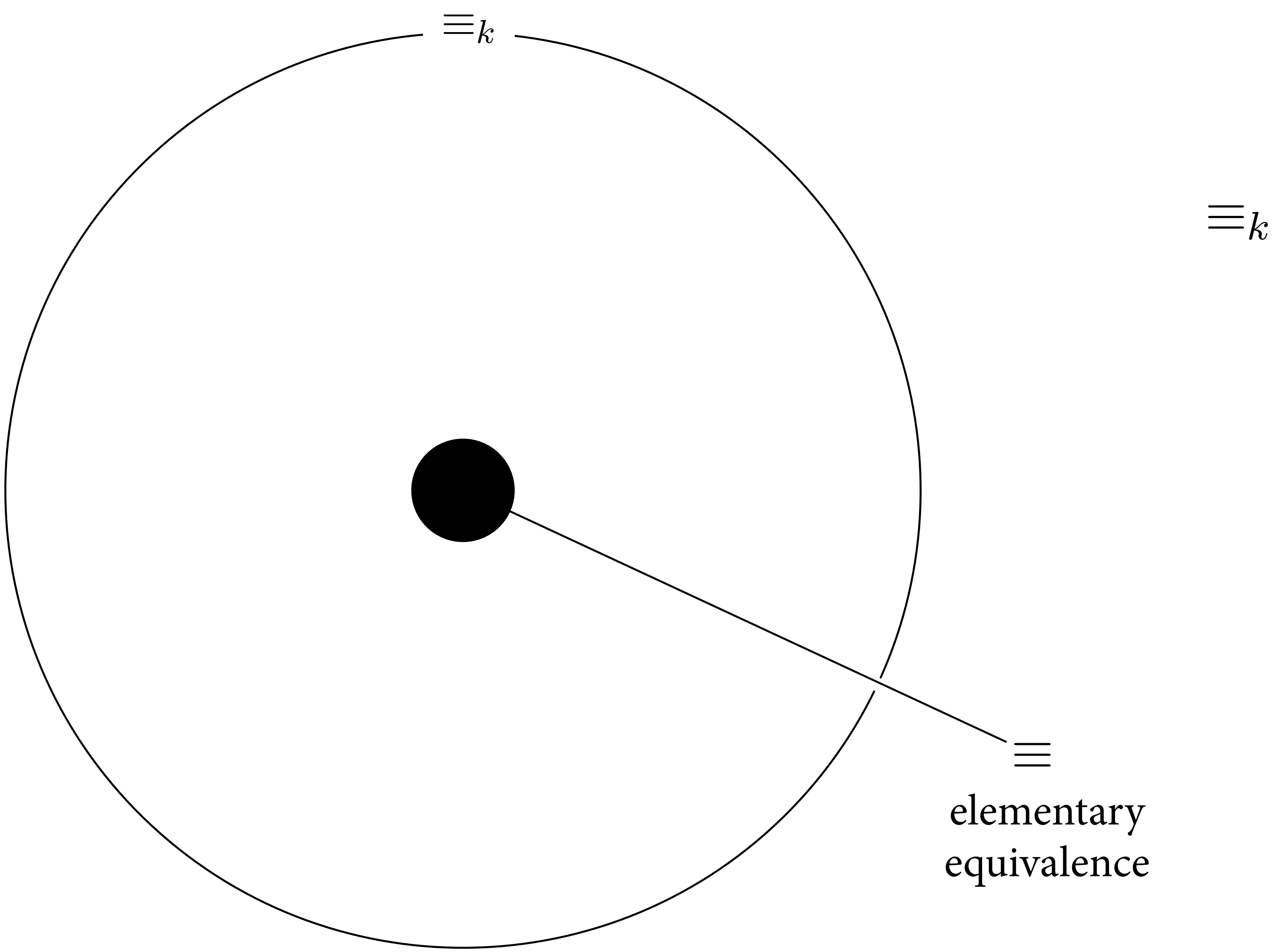
Relations between the different approximations

•

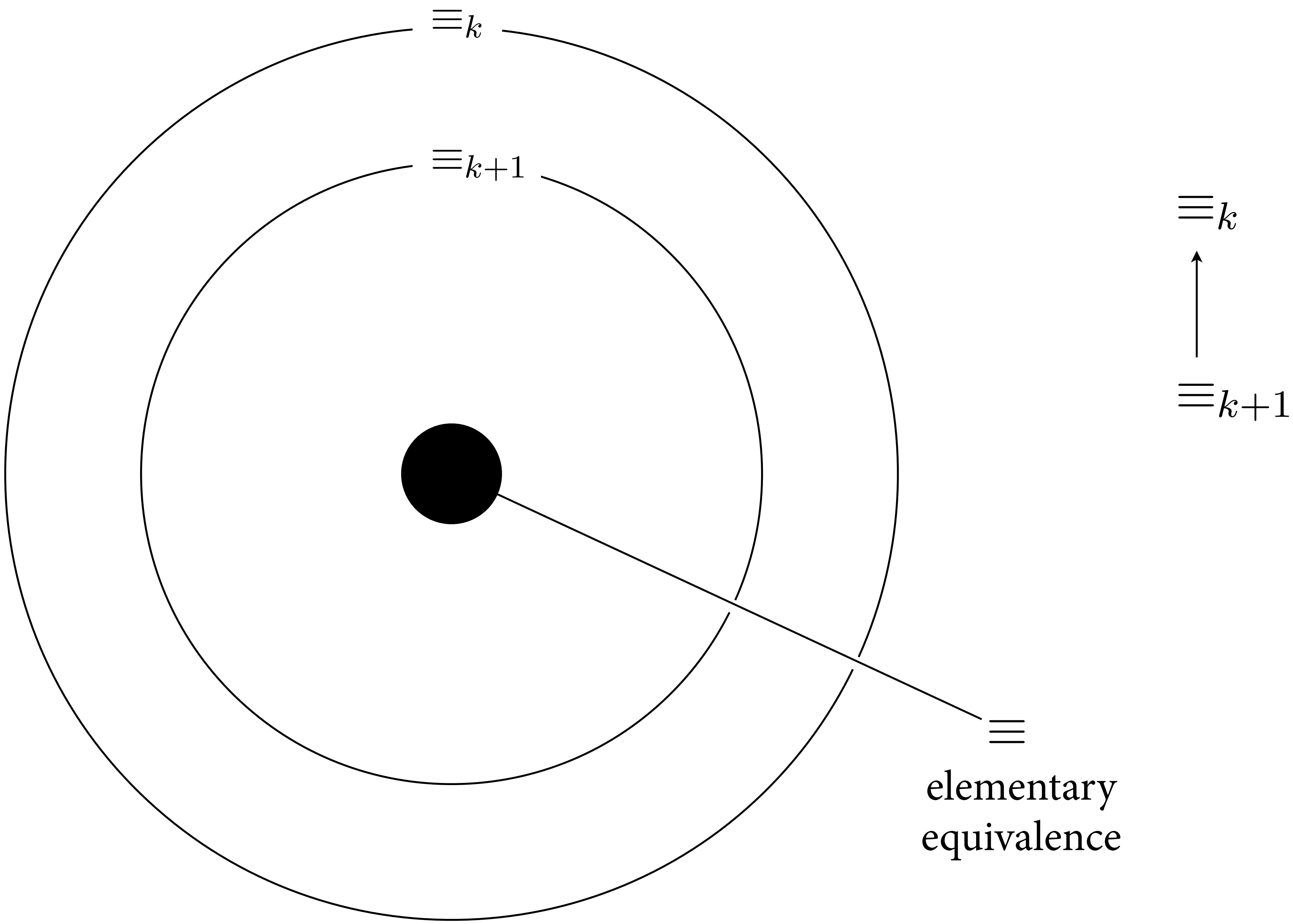
Relations between the different approximations



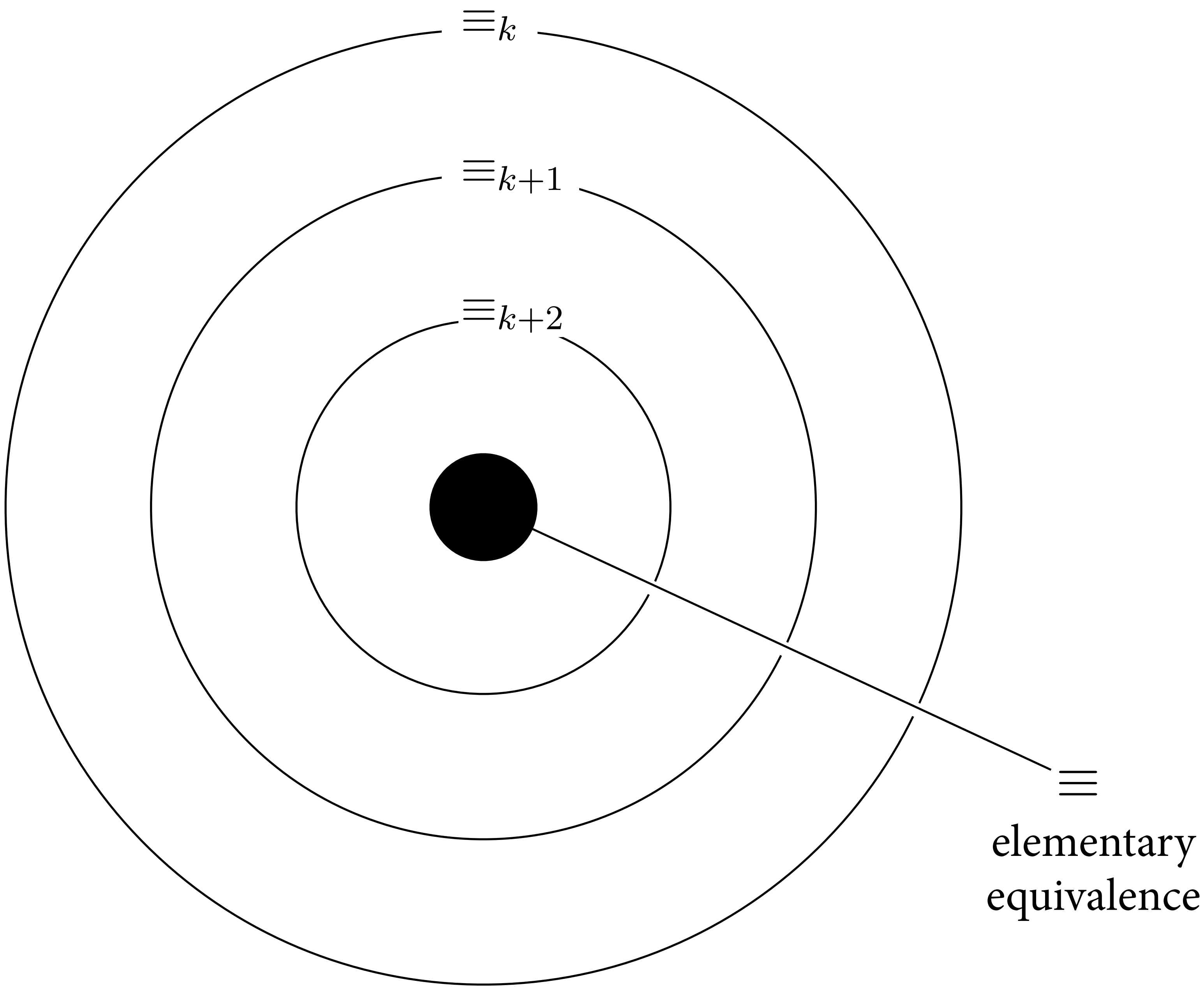
Relations between the different approximations

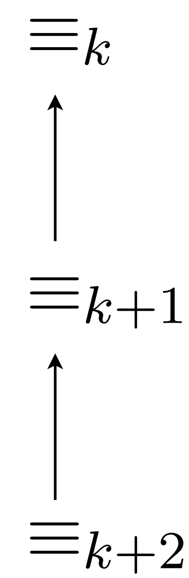


Relations between the different approximations

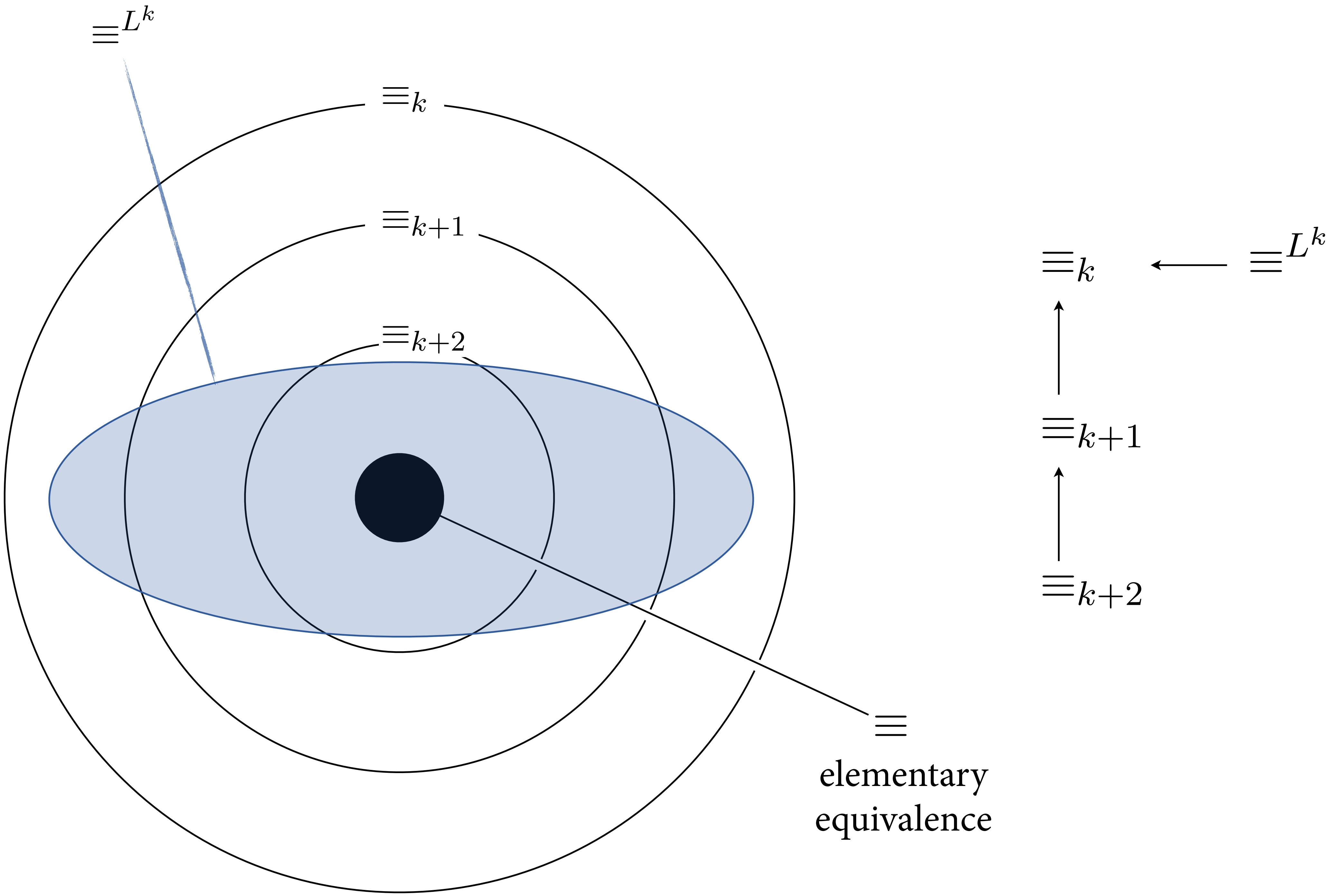


Relations between the different approximations

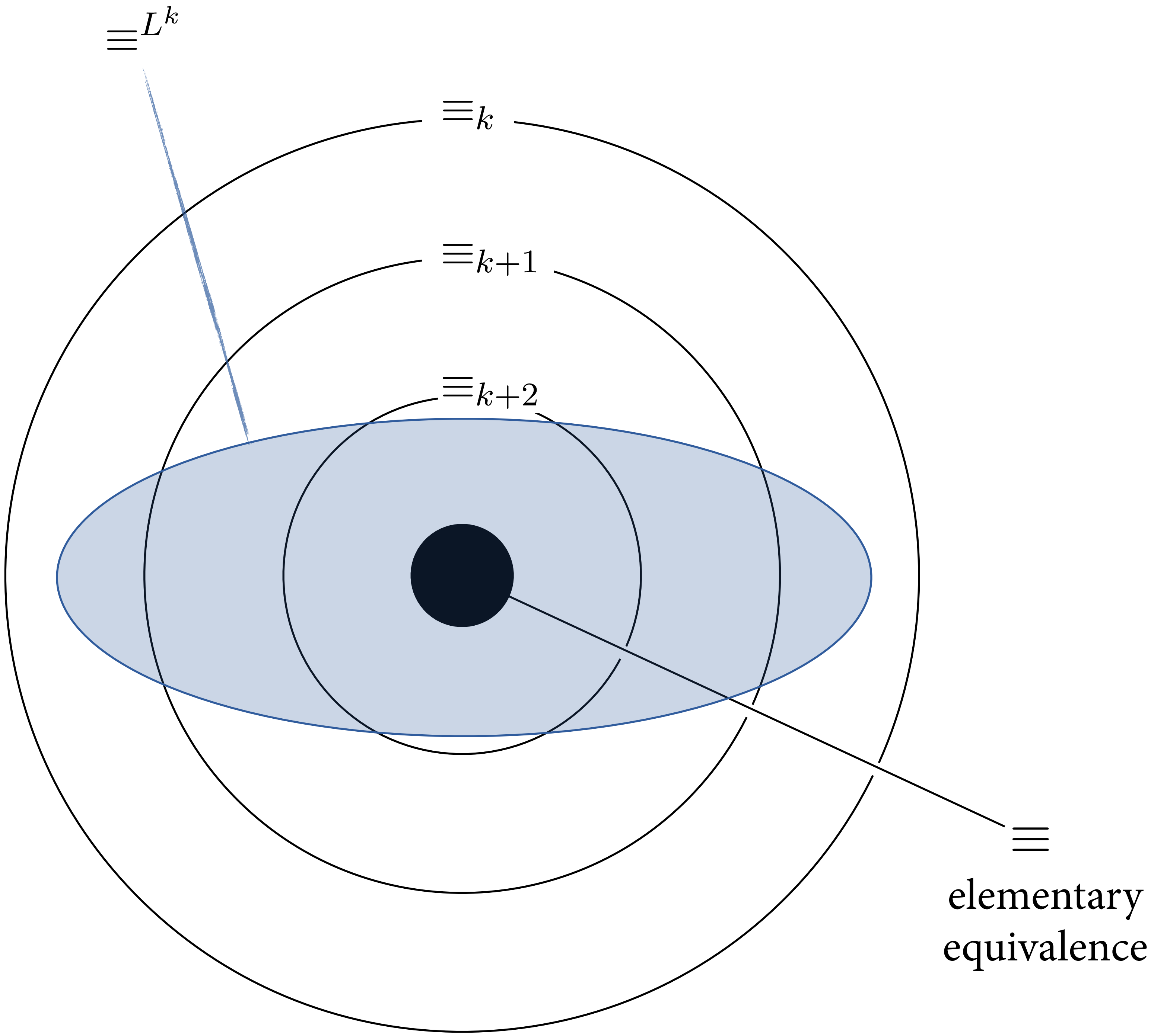


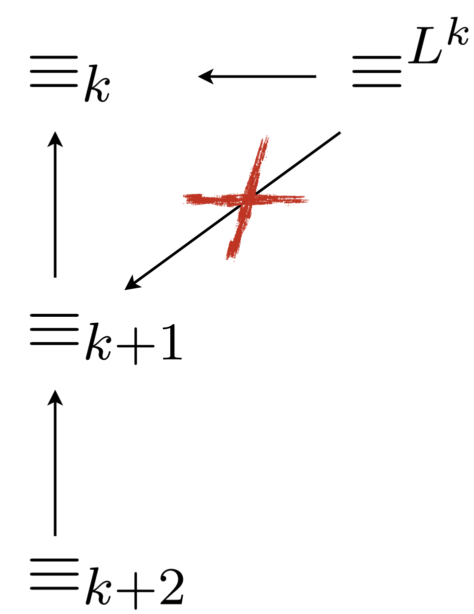


Relations between the different approximations

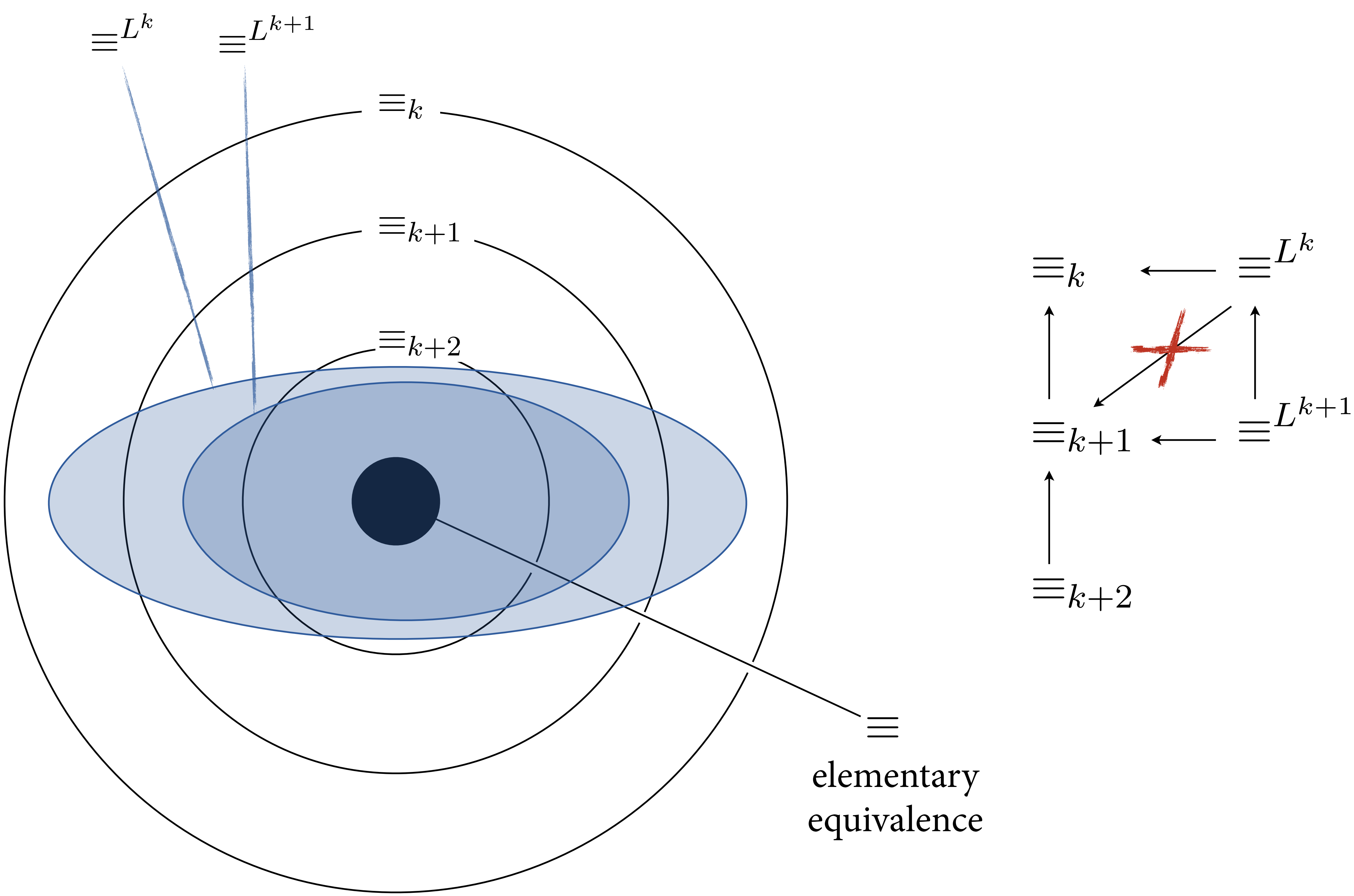


Relations between the different approximations

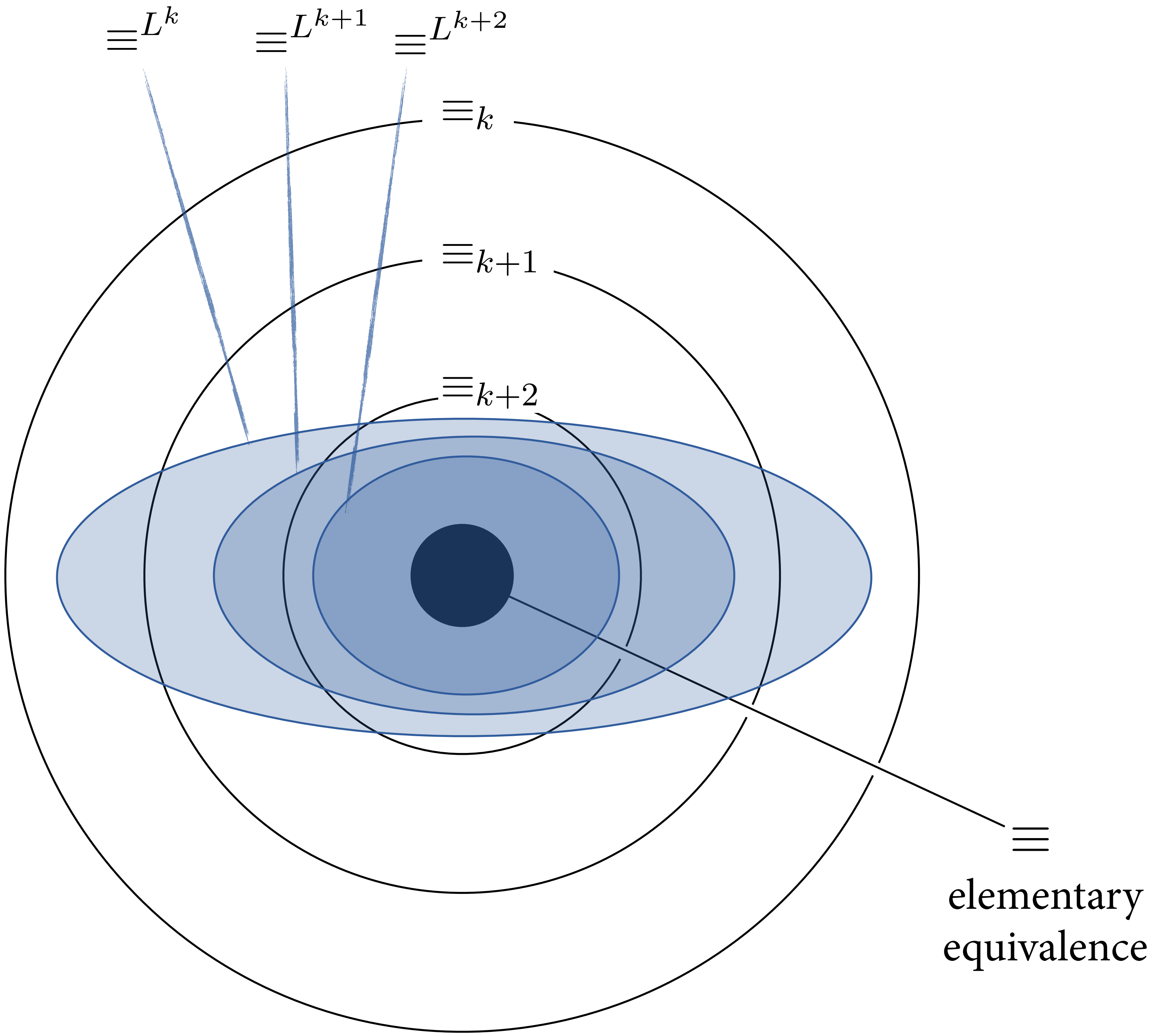


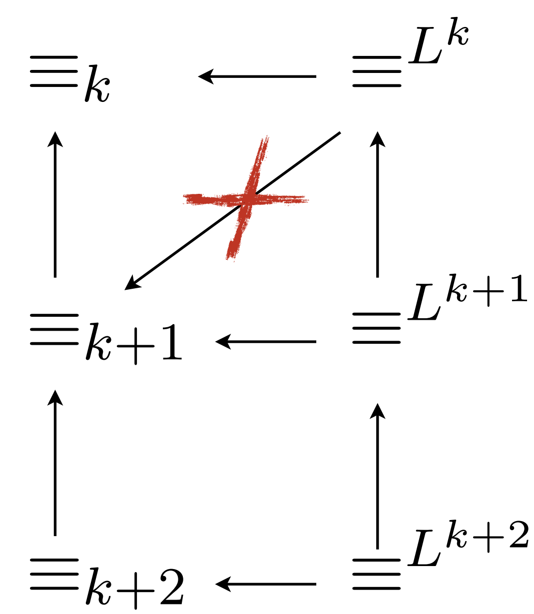


Relations between the different approximations

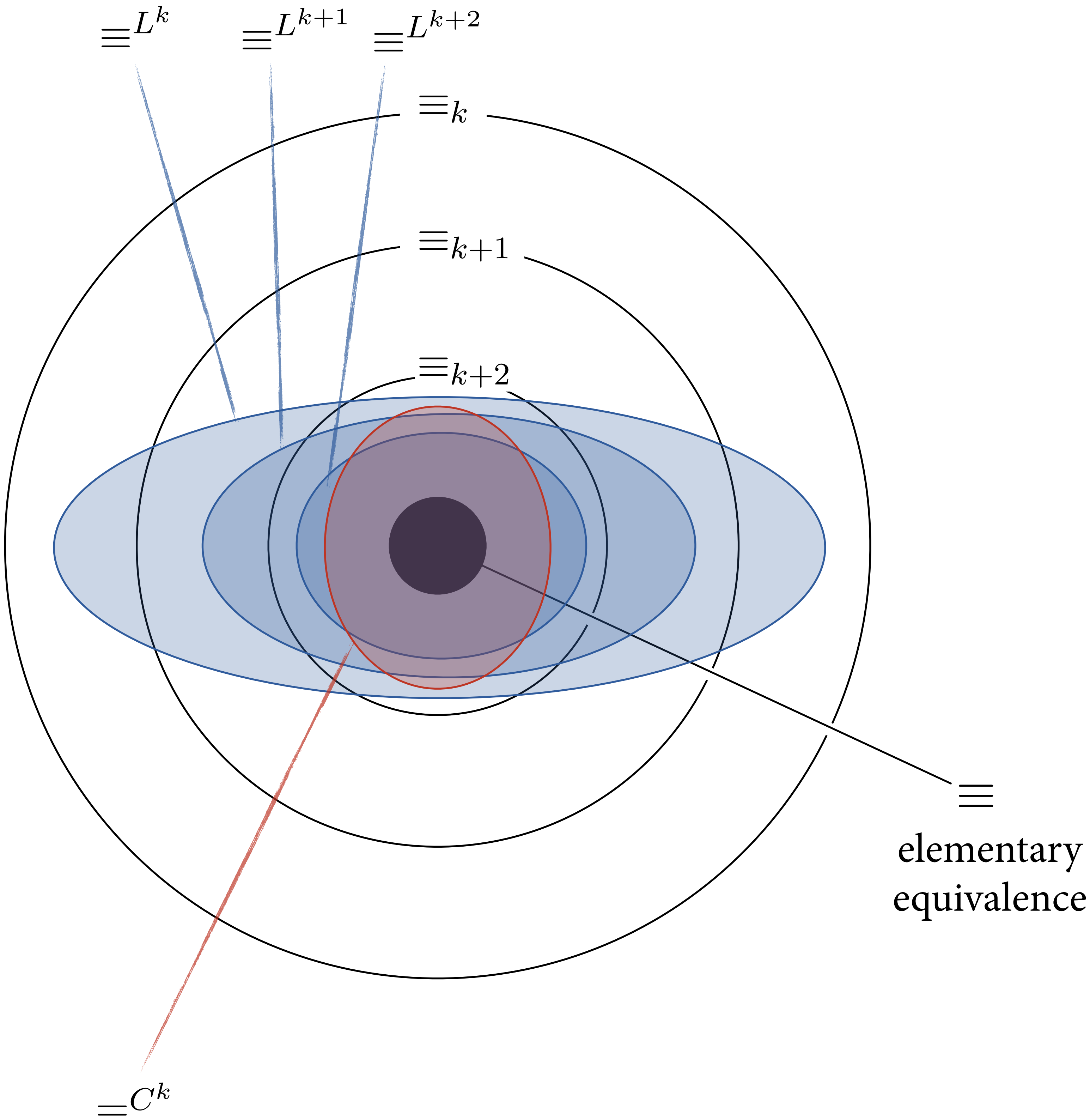


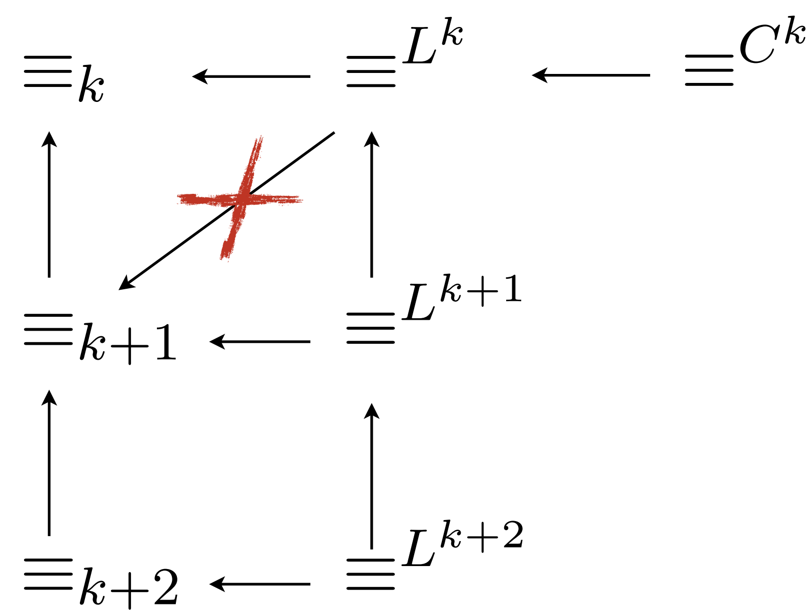
Relations between the different approximations



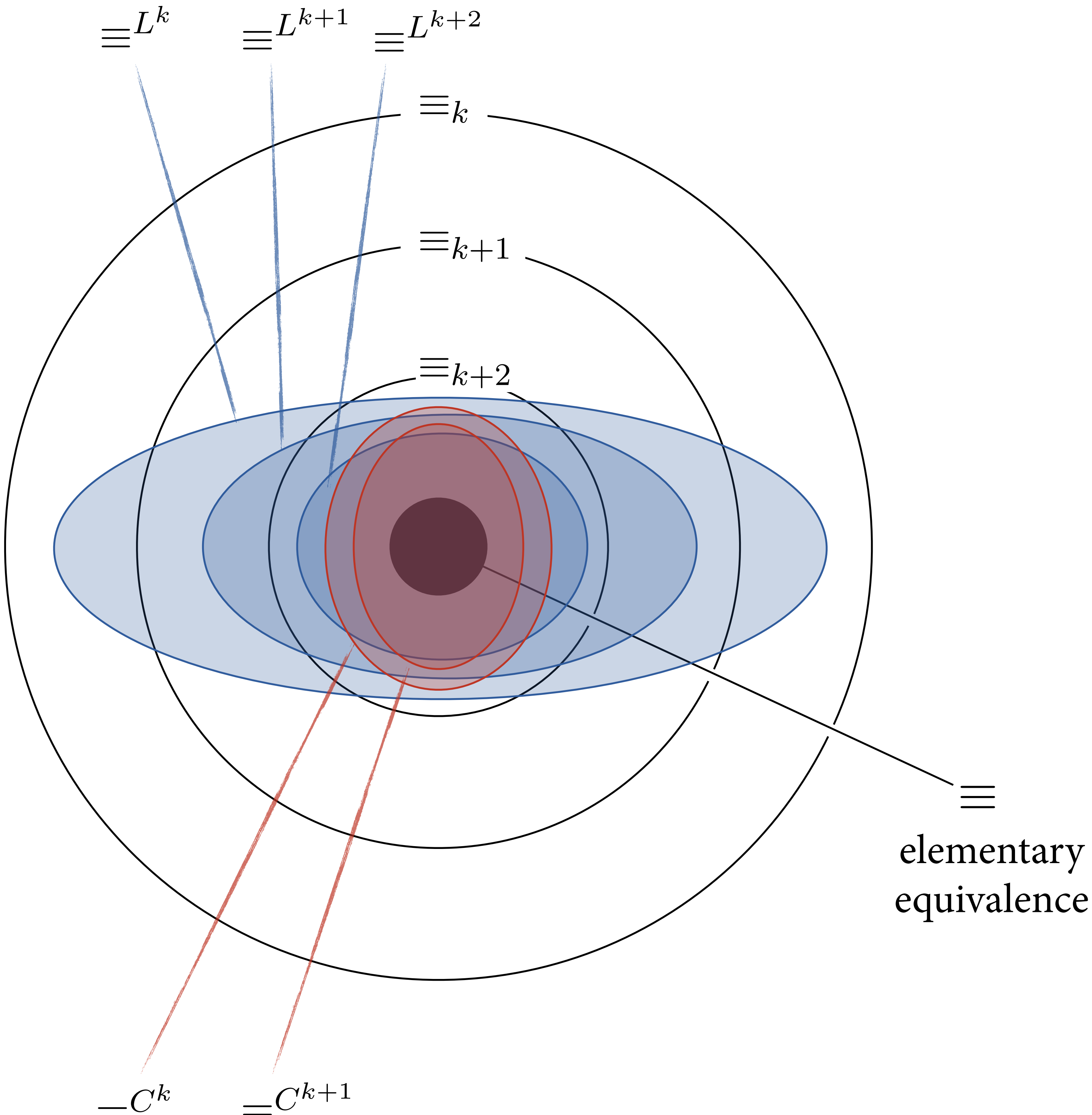


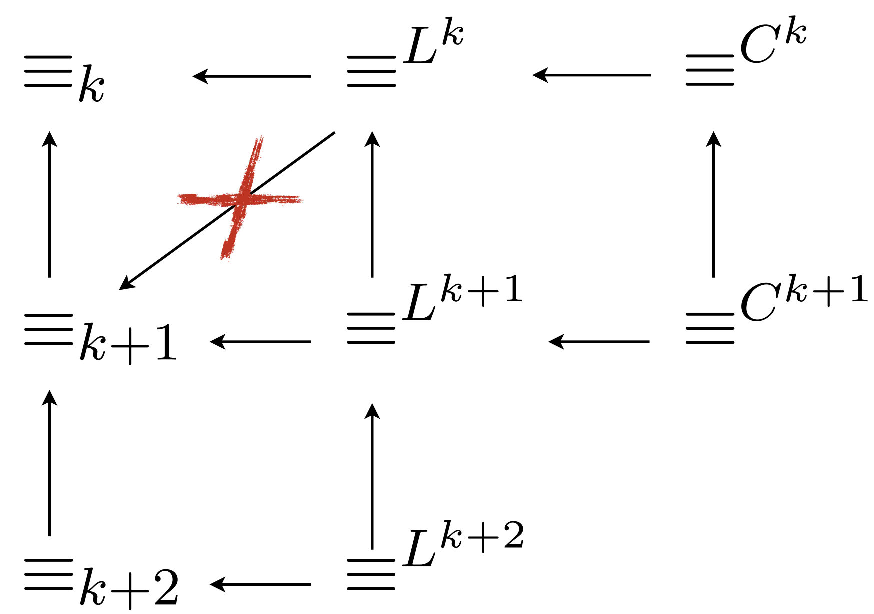
Relations between the different approximations



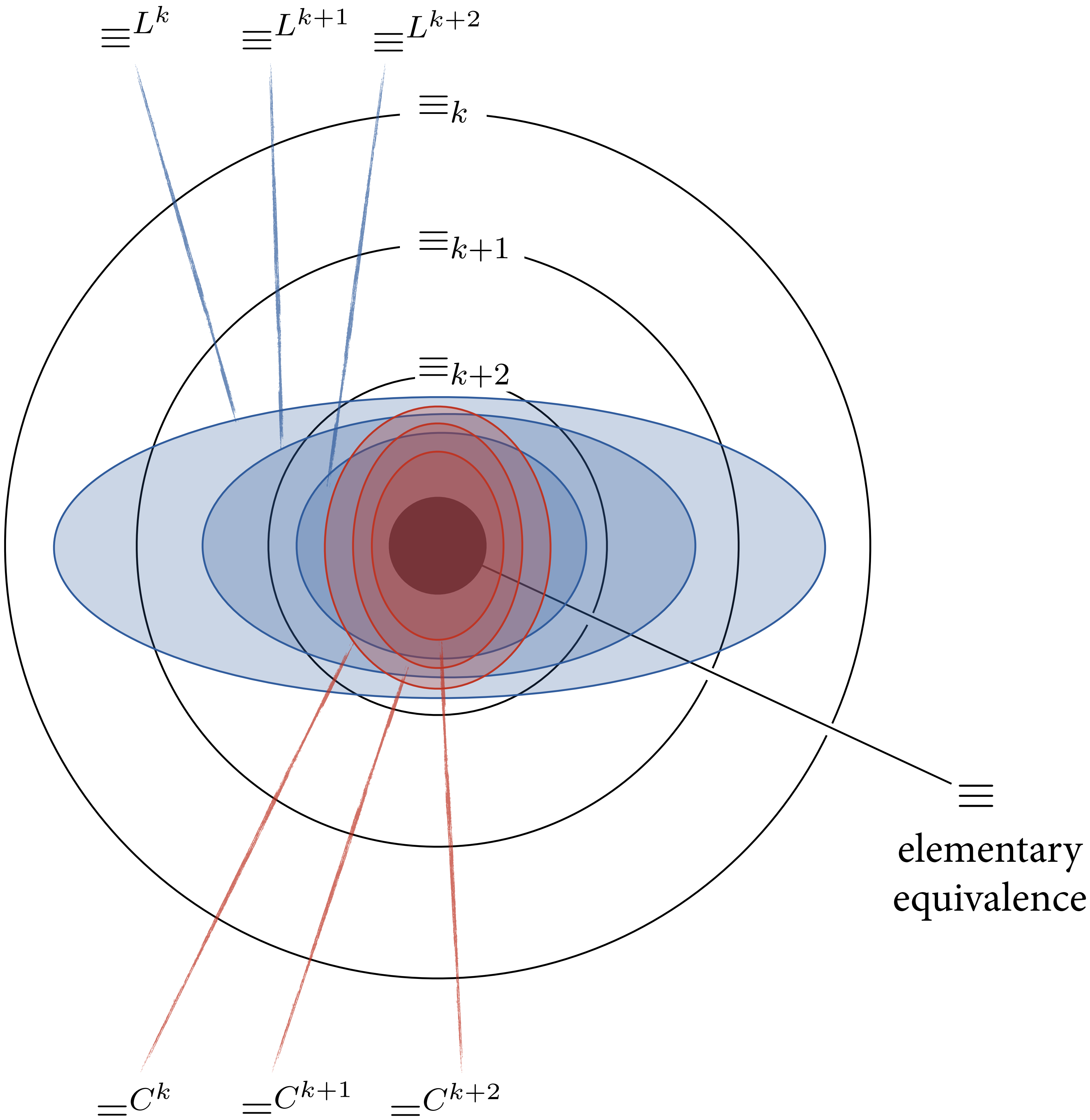


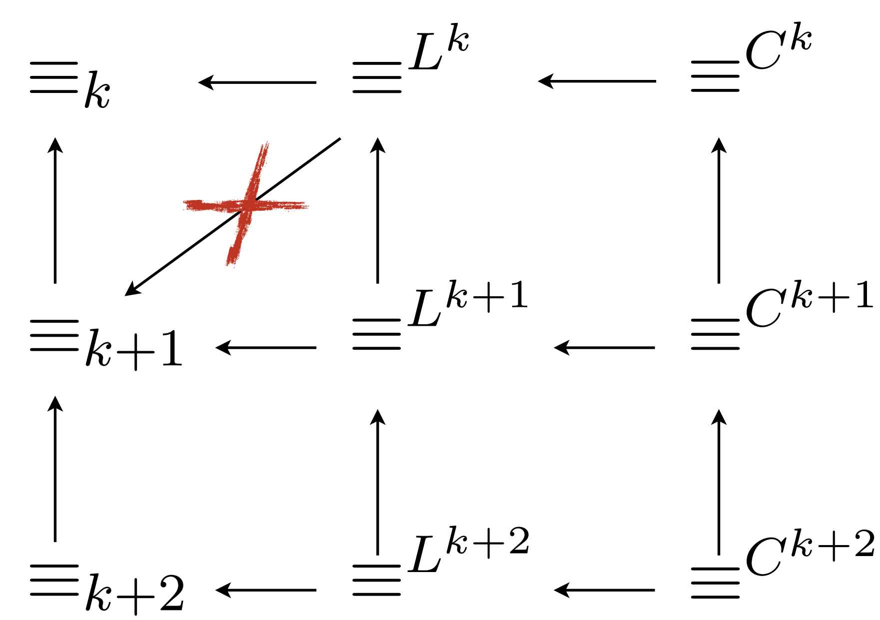
Relations between the different approximations



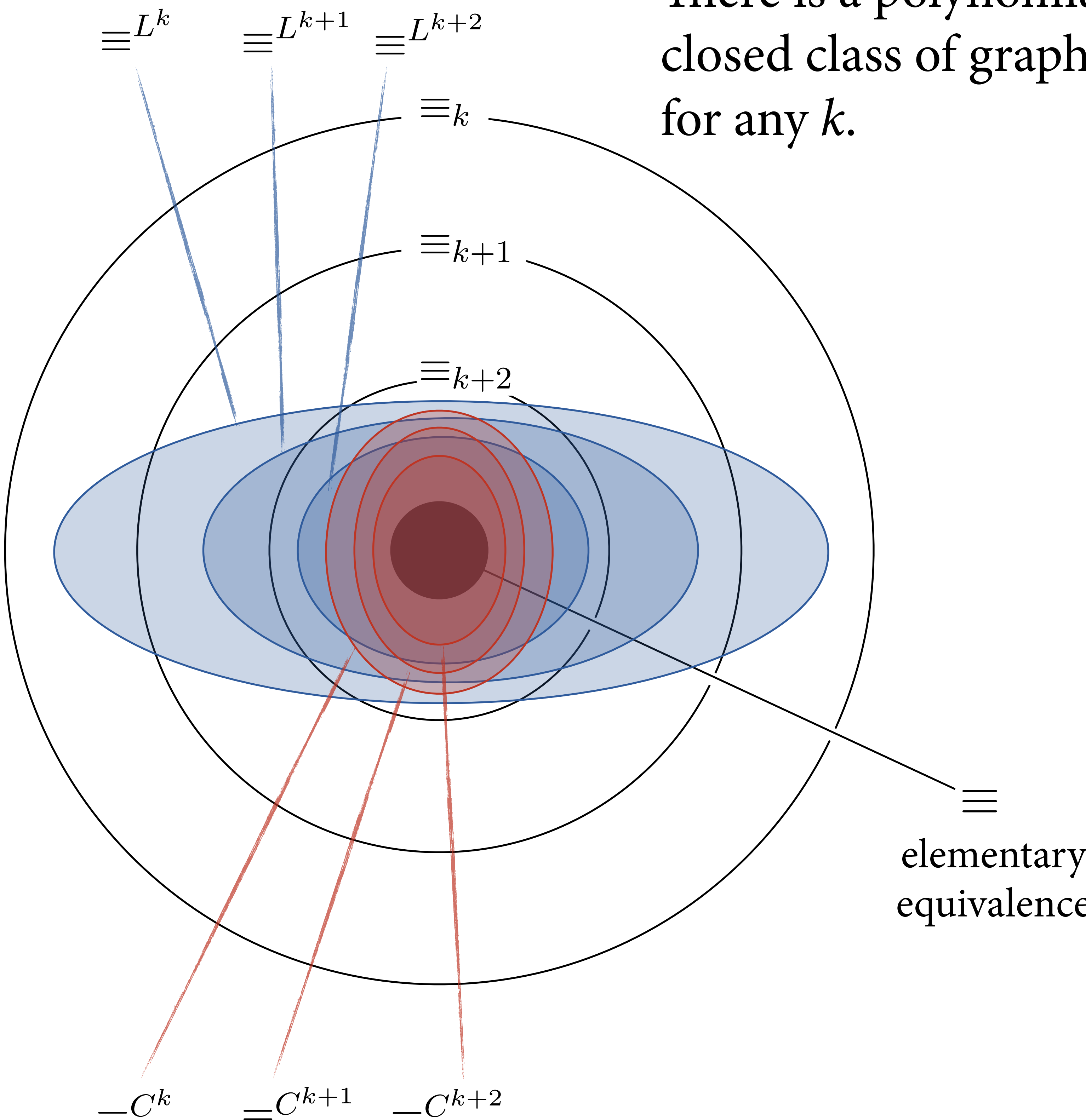


Relations between the different approximations



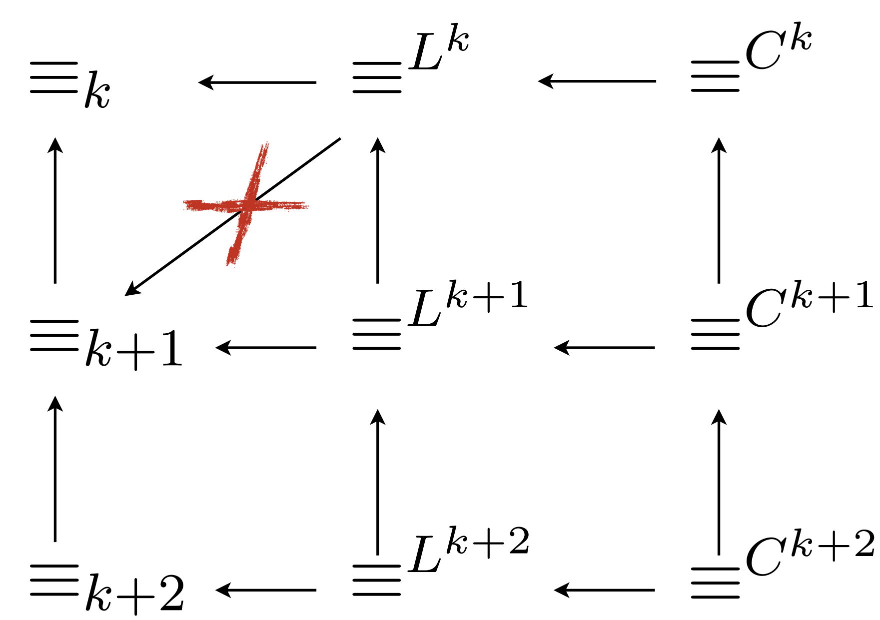


Relations between the different approximations



There is a polynomial-time decidable, isomorphism-closed class of graphs which is not closed under $\equiv_{C^k}$ for any k.

Cai, Furer and Immerman (1992)



Characterising logical equivalence by games

Our approach: variation of the EF game with algebraic game rules

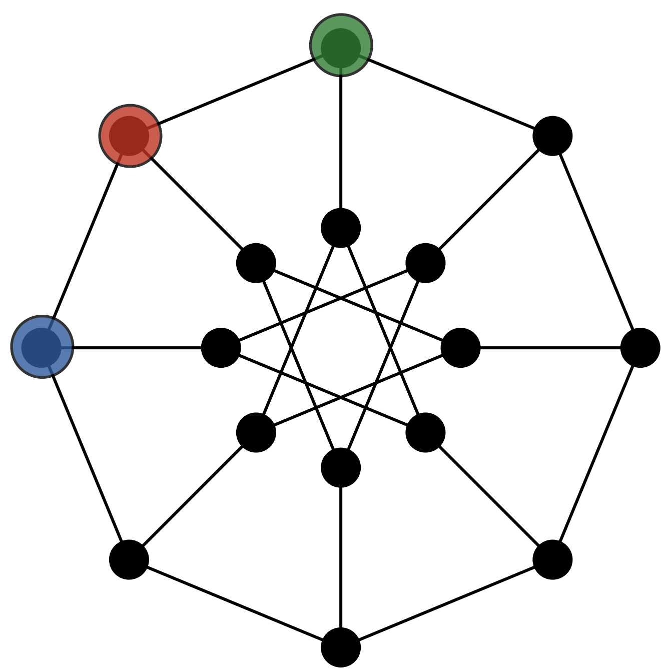
	Equivalence	Model-comparison game
$\equiv$	Elementary equivalence	Ehrenfeucht–Fraïsse (EF)
$\approx$	?	Invertible-map game
$\equiv^{C^k}$	*k*-variable FO with counting quantifiers $\exists^{\geq i} x \,.\, \varphi(x)$	*k*-pebble cardinality game
$\equiv^{L^k}$	First-order logic with variables $x_1, \ldots, x_k$	*k*-pebble game
$\equiv_r$	First-order logic up to quantifier rank *r*	*r*-round EF game

The k-pebble cardinality game

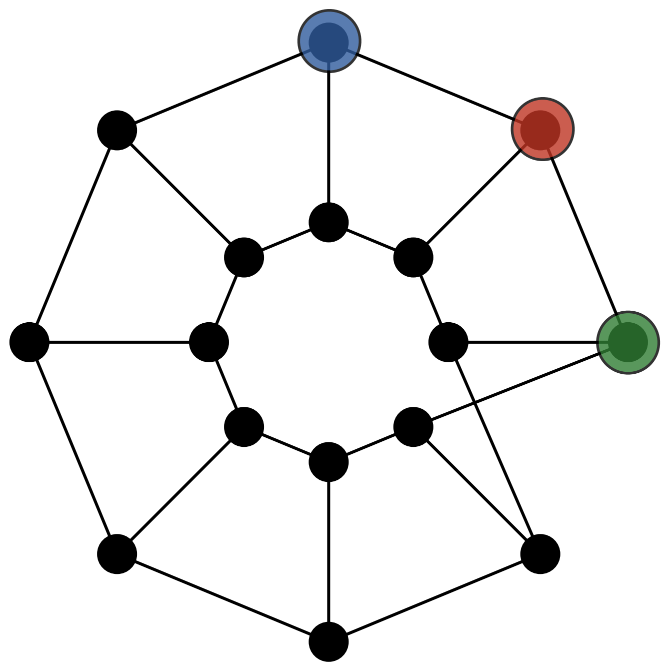
C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$



G



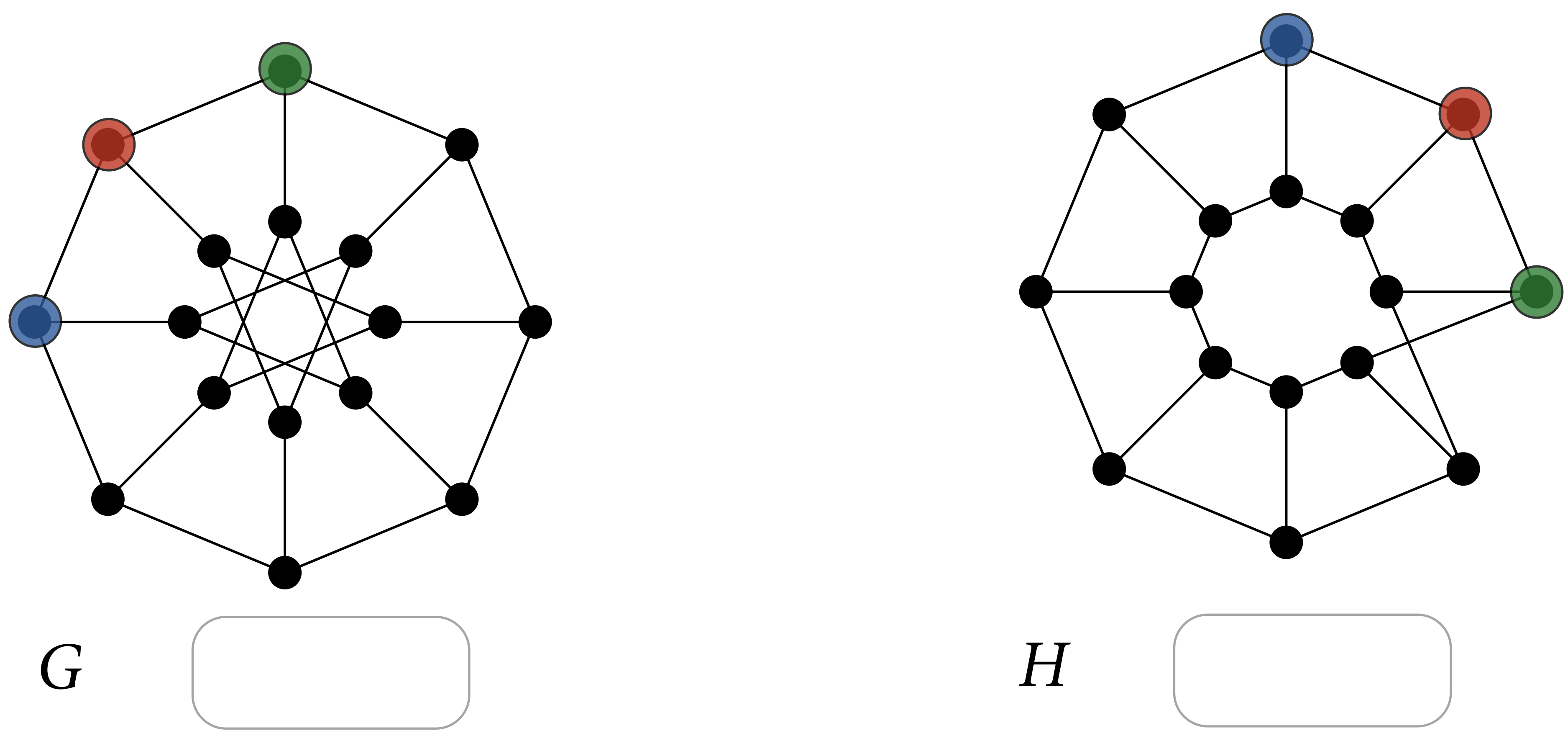
H

Spoiler

Duplicator

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

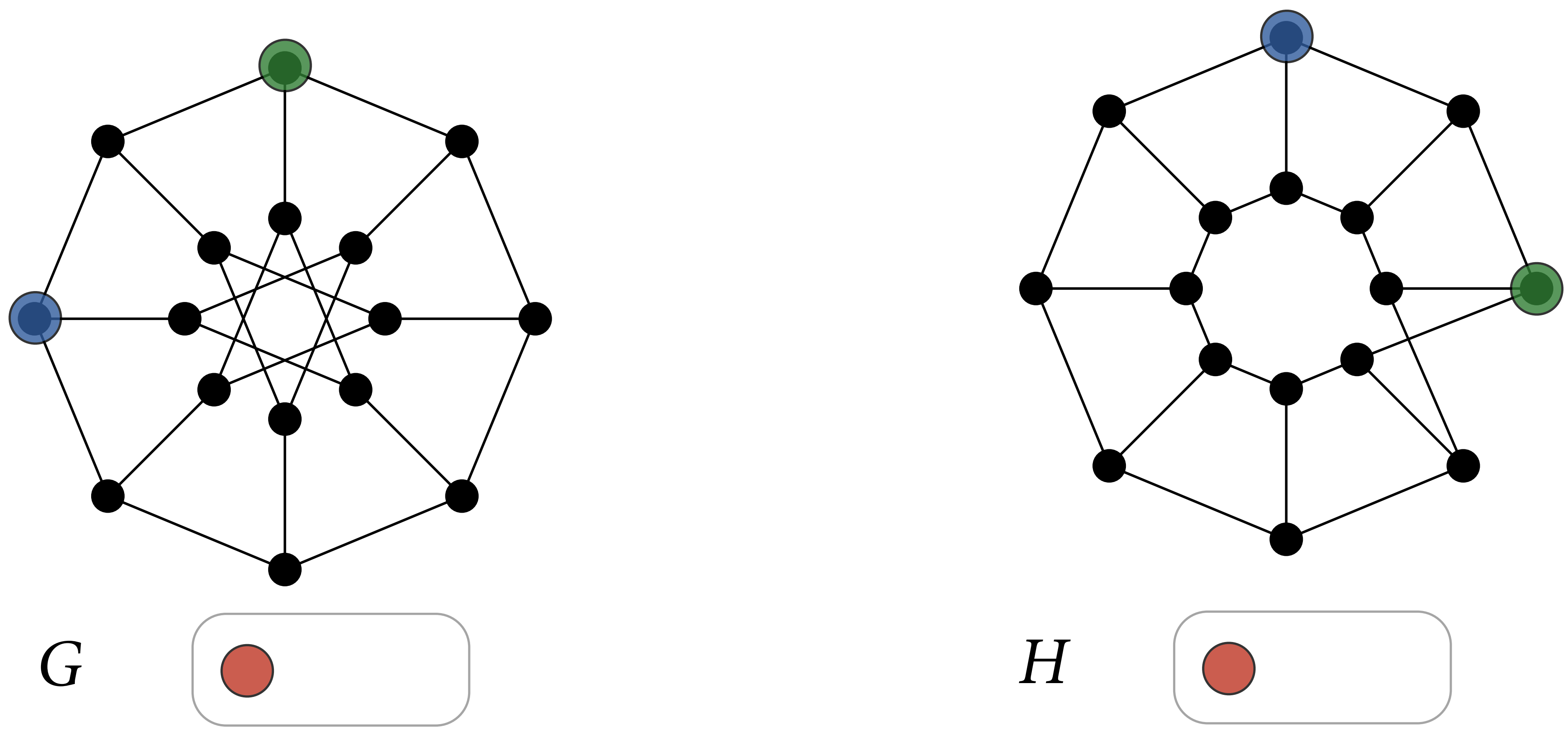


Spoiler chooses pebbles to move from the two graphs

Duplicator

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \, . \, \varphi(x)$

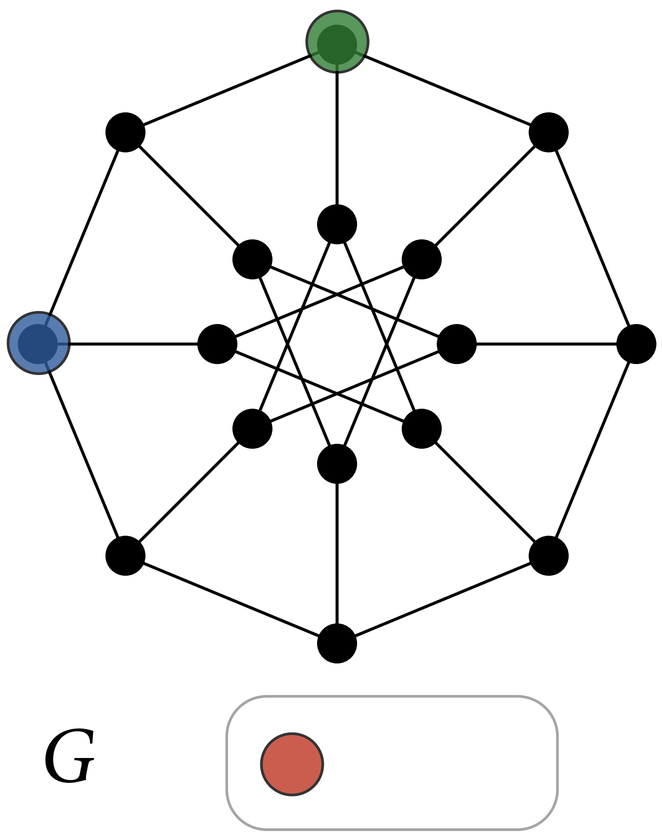


Spoiler chooses pebbles to move from the two graphs

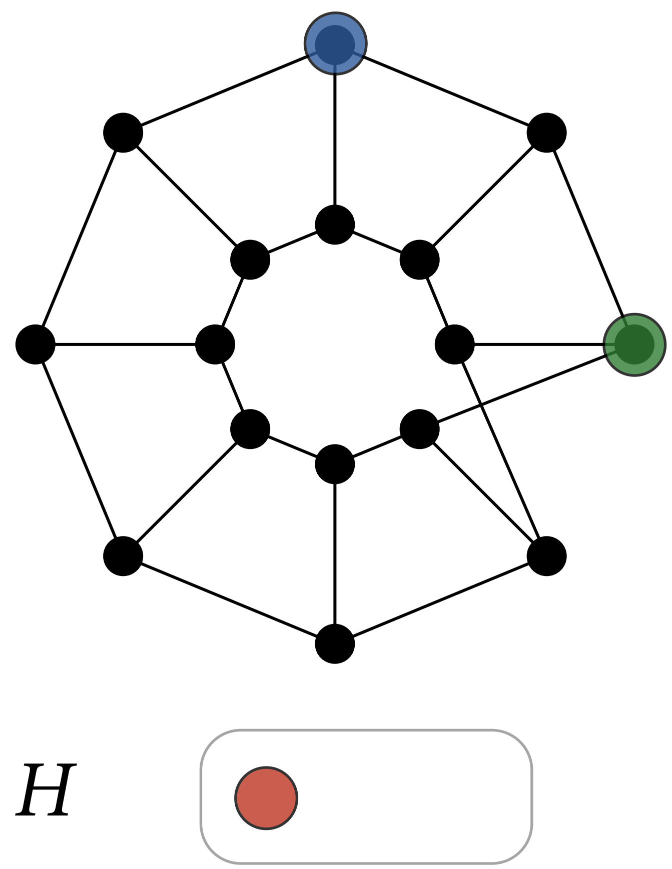
Duplicator

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$



G



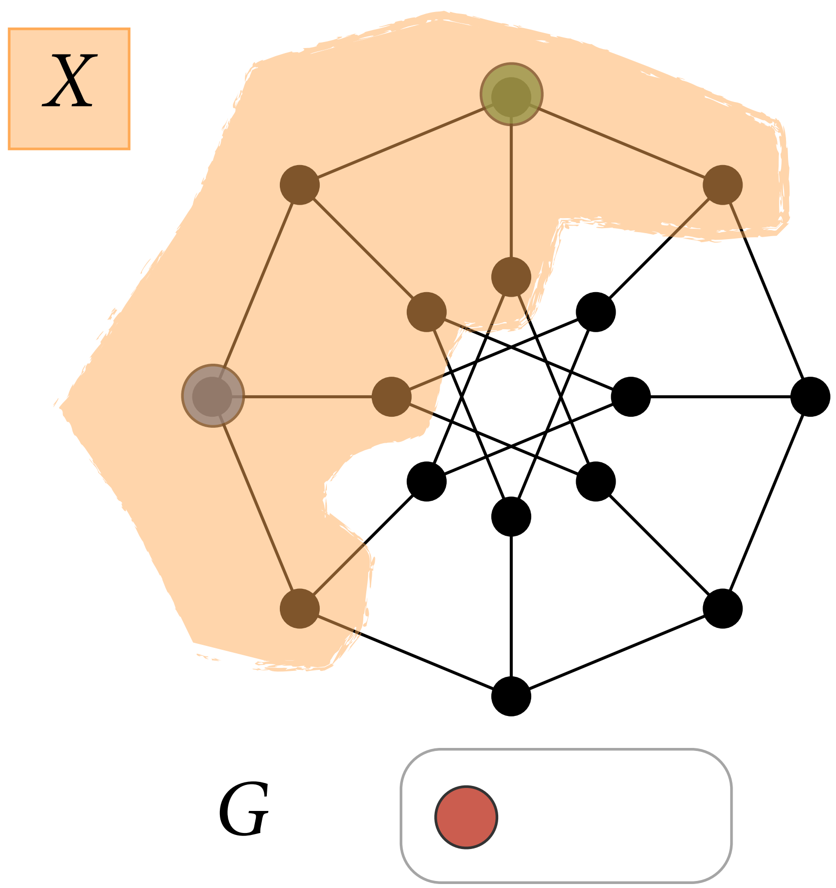
H

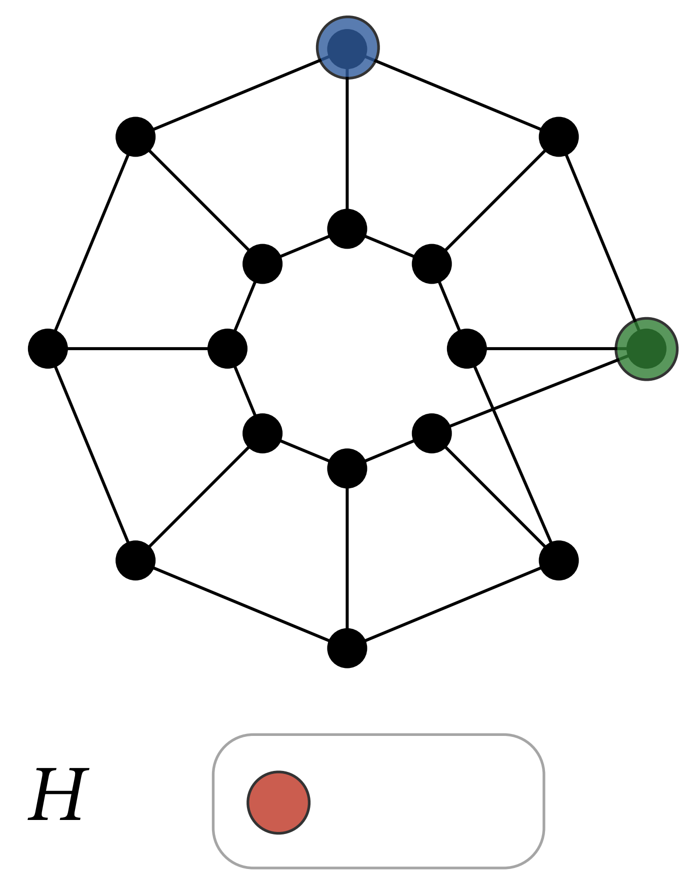
Spoiler chooses of set X of vertices in one of the graphs

Duplicator

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$



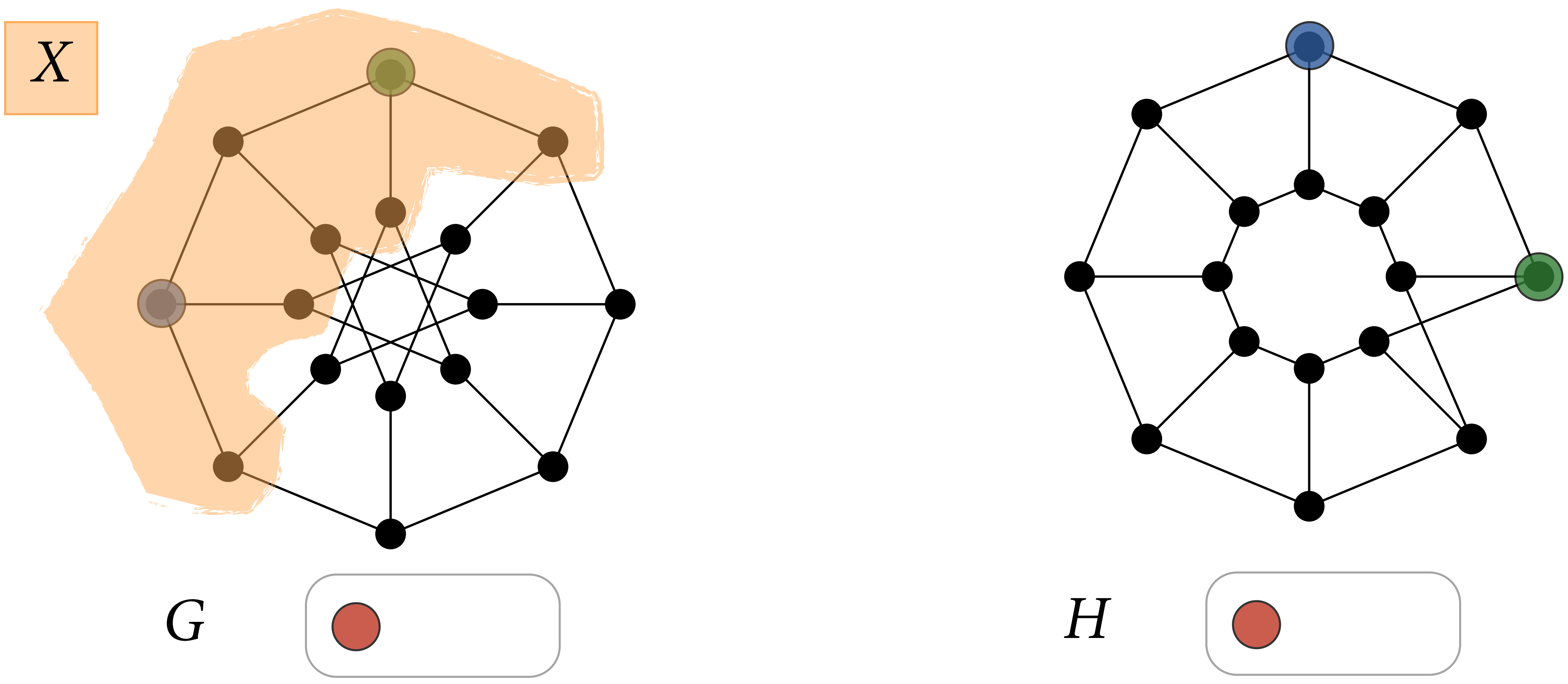


Spoiler chooses of set X of vertices in one of the graphs

Duplicator

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

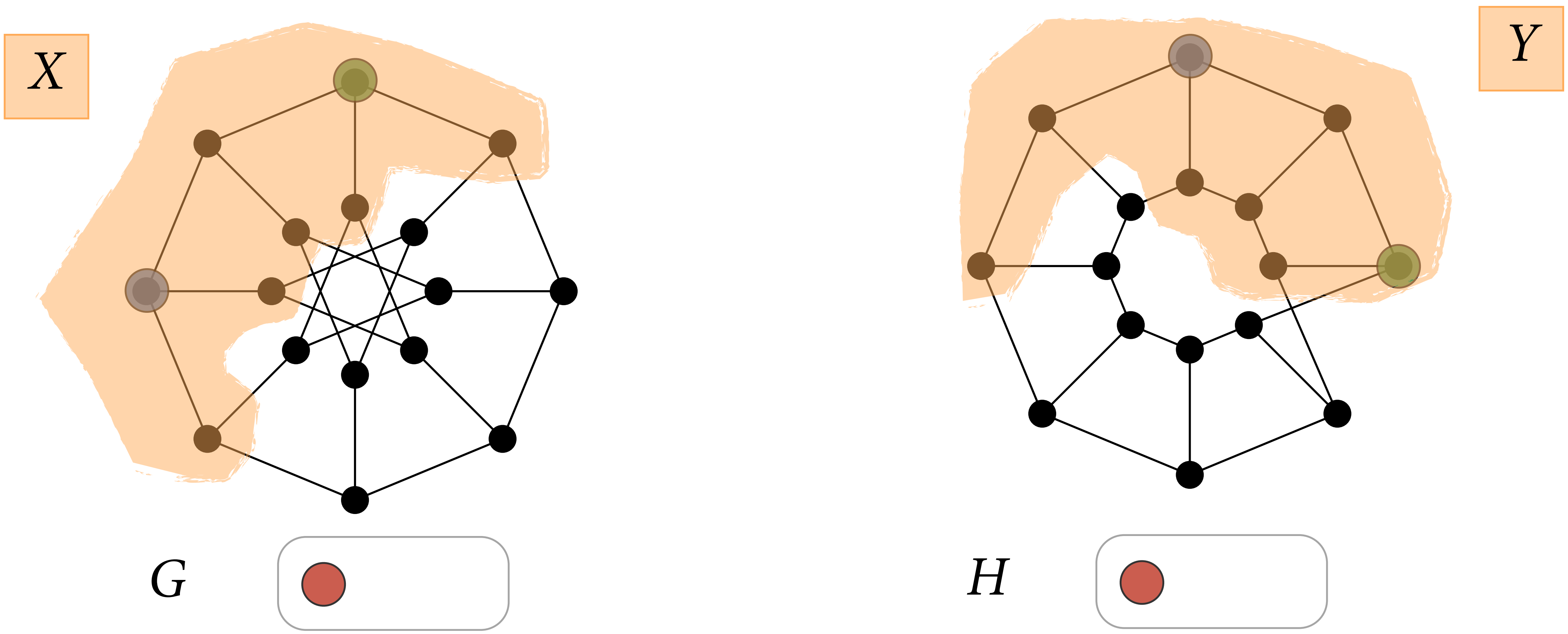


Spoiler

Duplicator responds with a set Y of H of the same cardinality

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x\,.\,\varphi(x)$

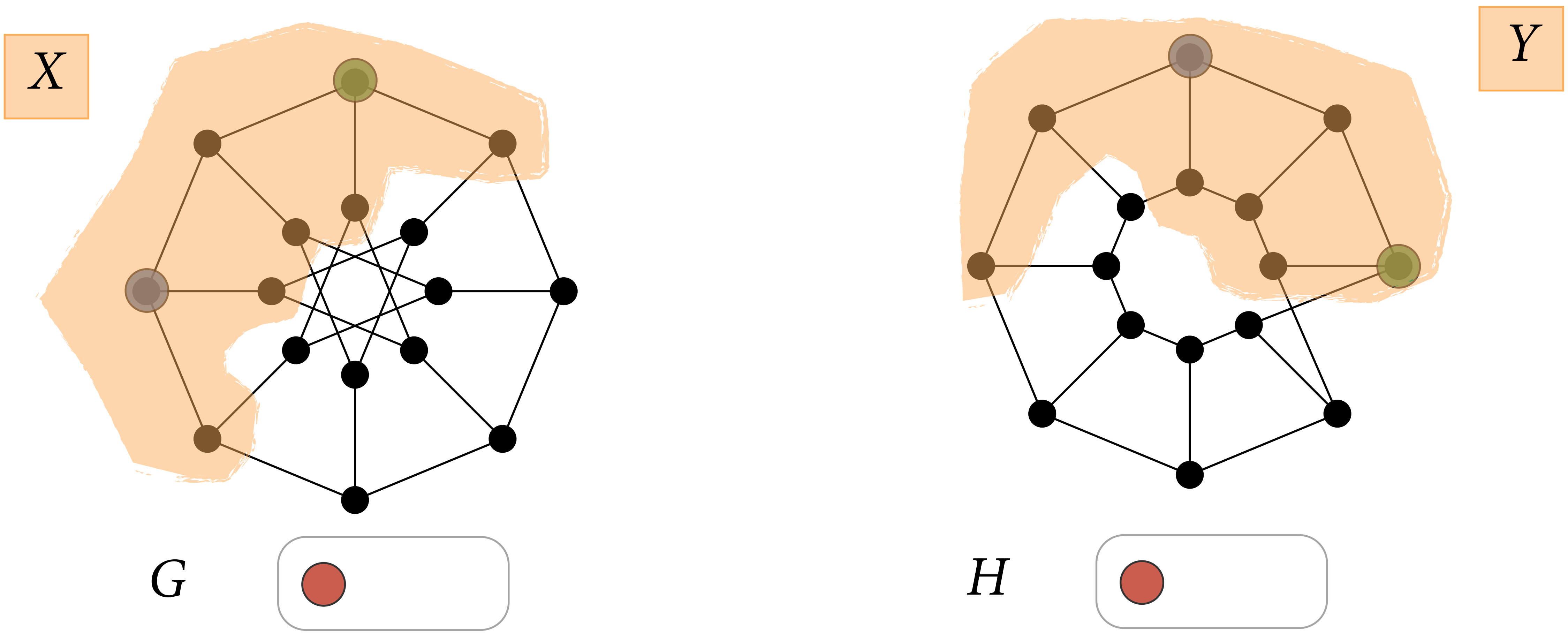


Spoiler

Duplicator responds with a set Y of H of the same cardinality

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

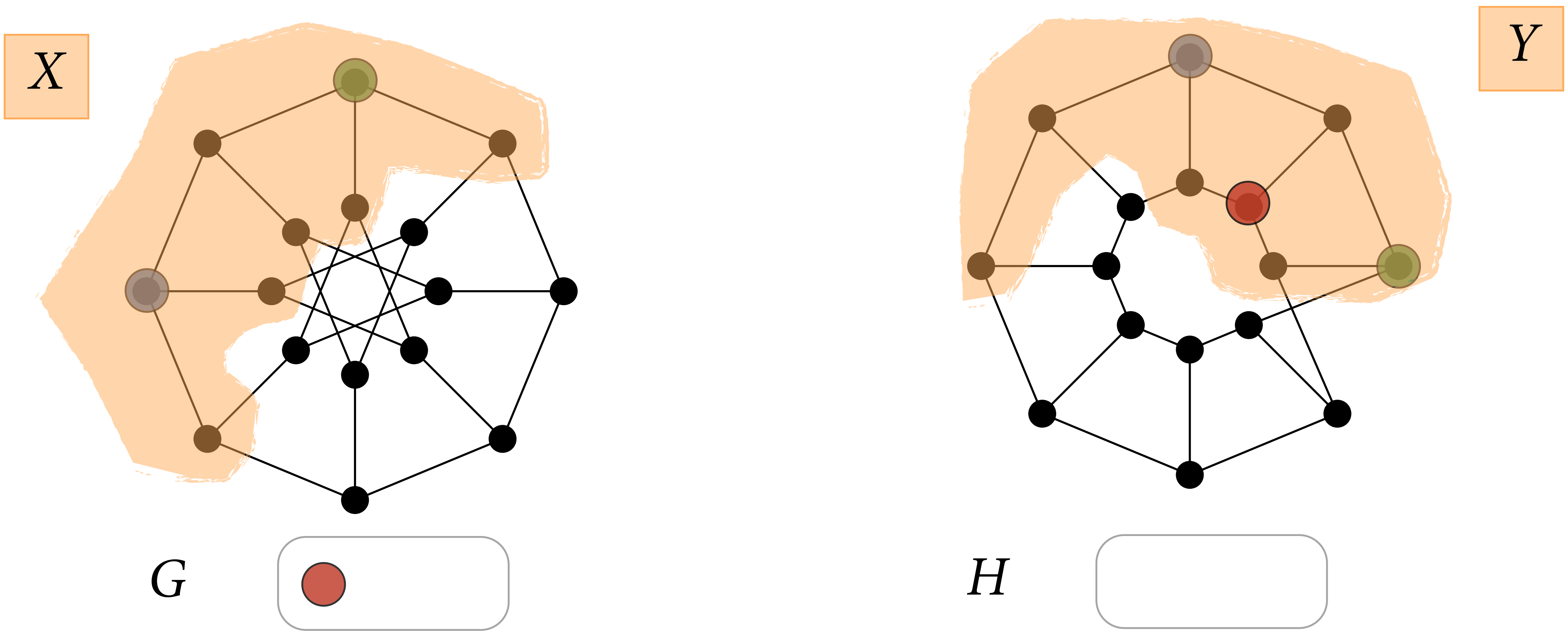


Spoiler places the red pebble in H on an element of Y

Duplicator

The *k*-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

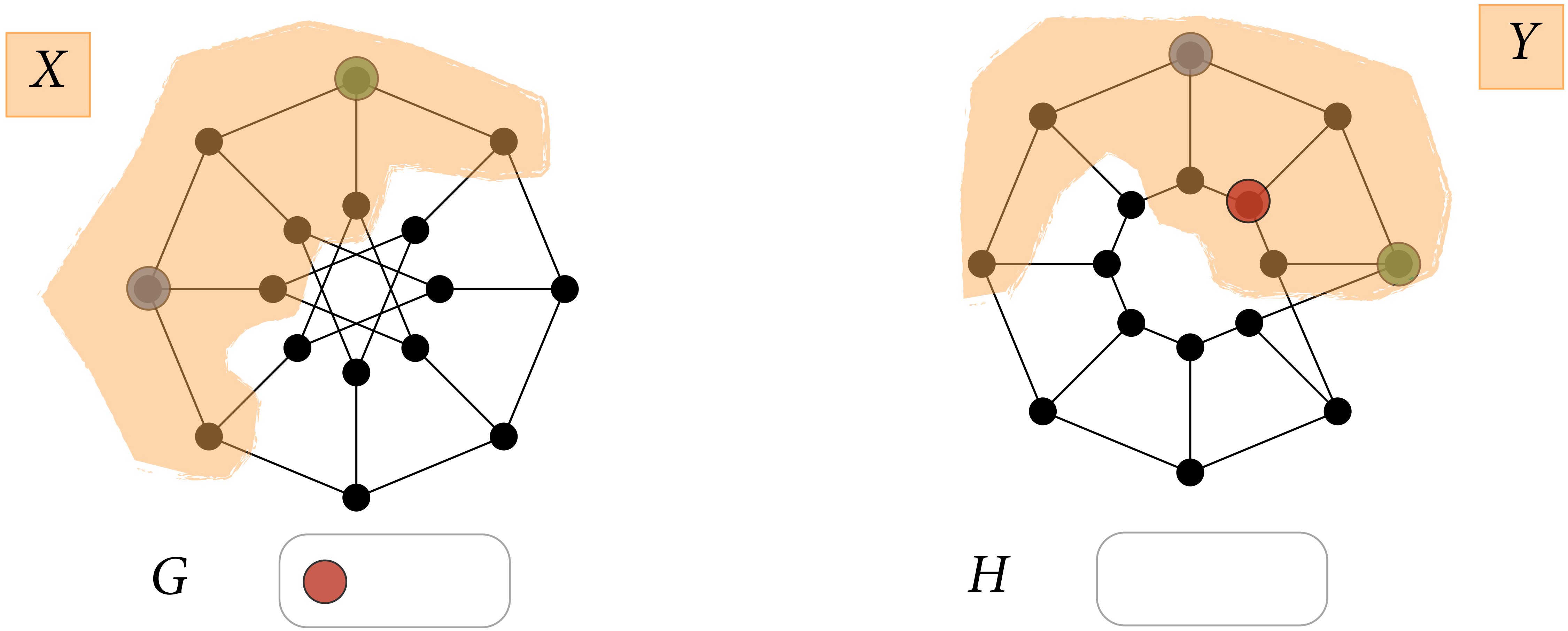


Spoiler places the red pebble in H on an element of Y

Duplicator

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

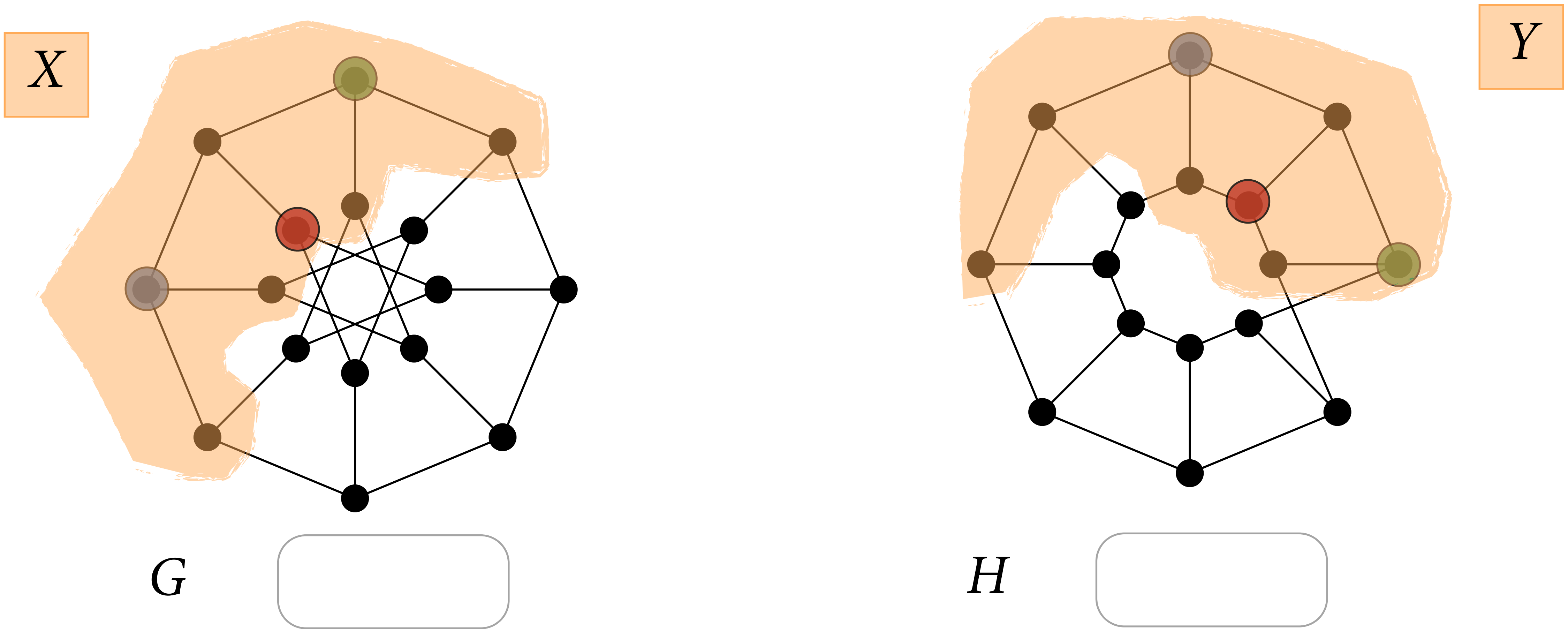


Spoiler

Duplicator places the red pebble in G on an element of X

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

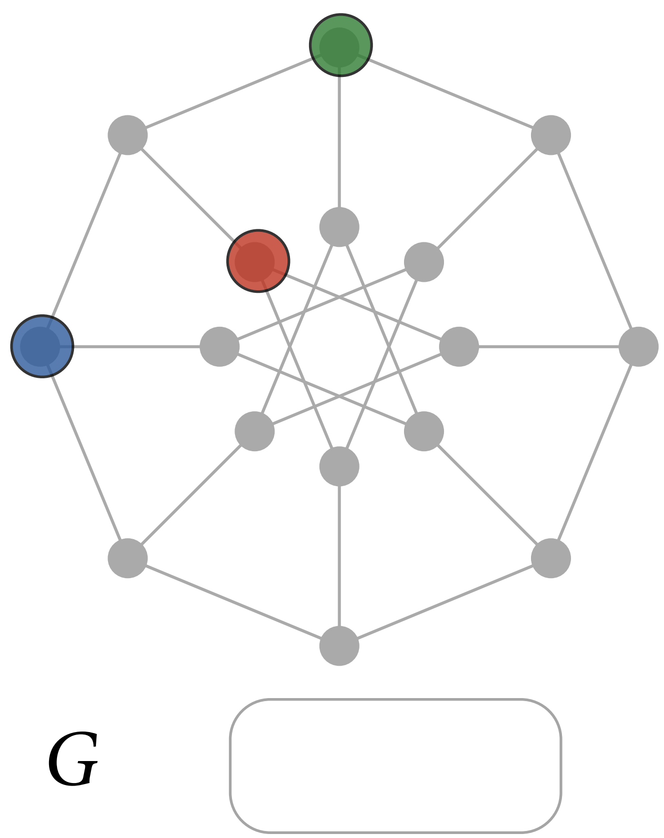


Spoiler

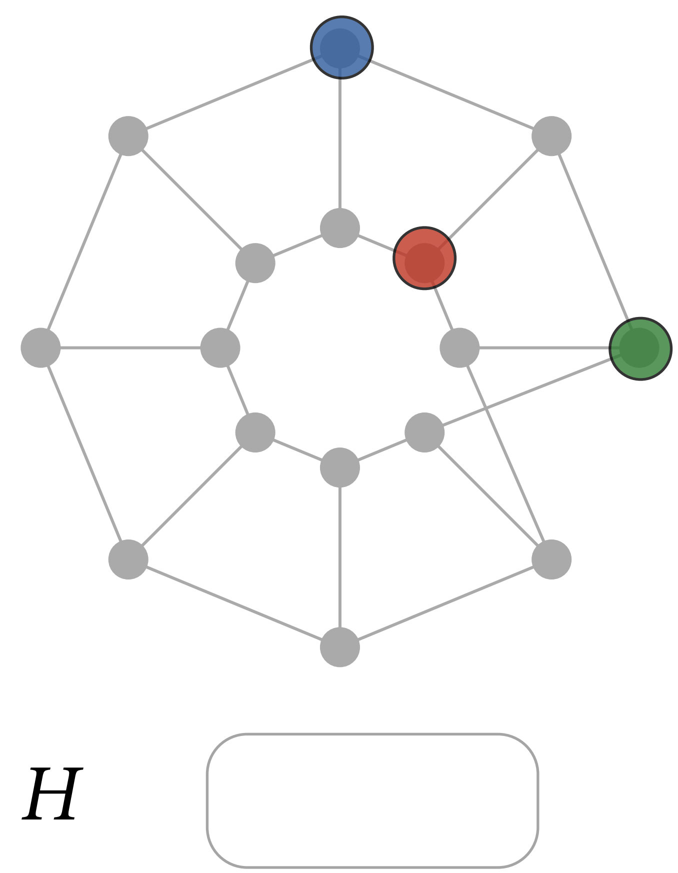
Duplicator places the red pebble in G on an element of X

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$



G

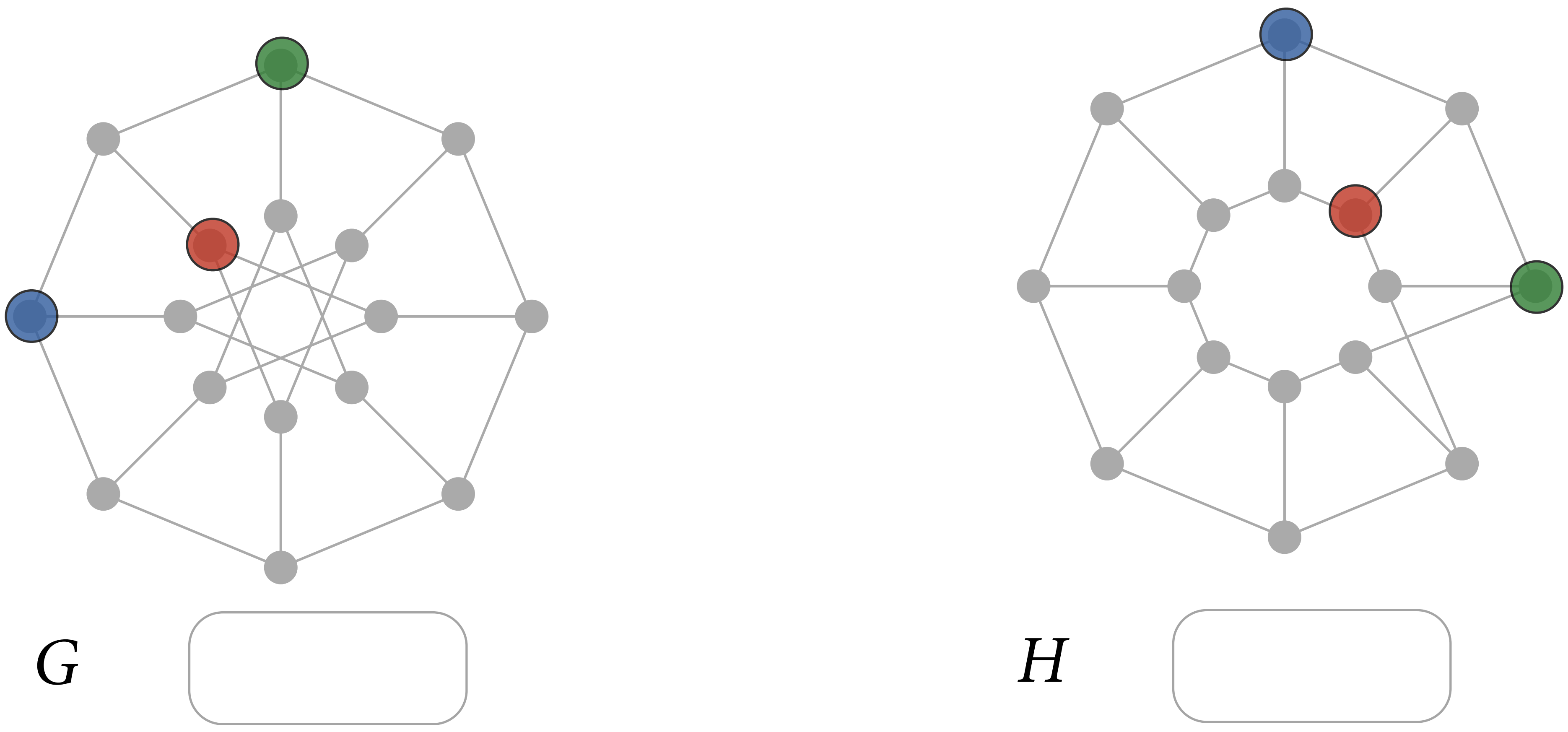


H

The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

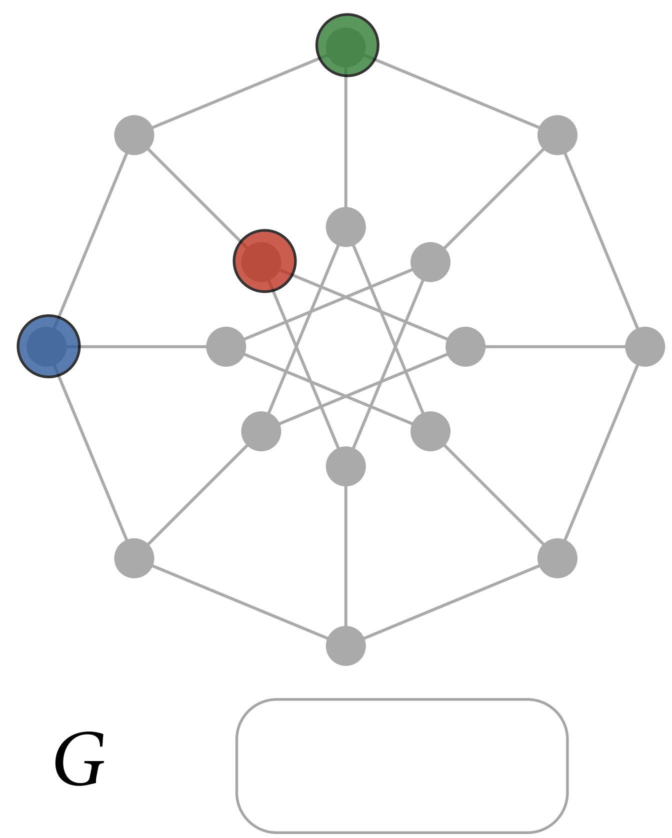
Is the pebble mapping a partial isomorphism?

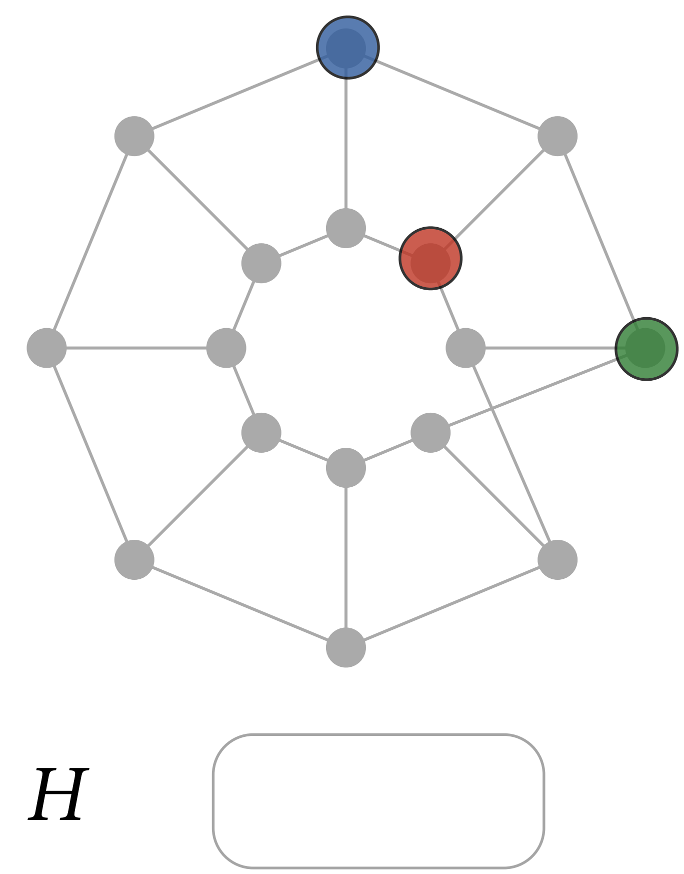


The k-pebble cardinality game

C^k — extension of L^k with counting quantifiers: $\exists^{\geq i} x \,.\, \varphi(x)$

Is the pebble mapping a partial isomorphism?





Duplicator has a strategy to play forever in the k-pebble cardinality game on G and H iff $G \equiv^{C^k} H$.

Immerman and Lander (1990)

Where C^k-equivalence fails

Limitations of C^k can be largely explained by its inability to express basic problems in linear algebra over finite domains.

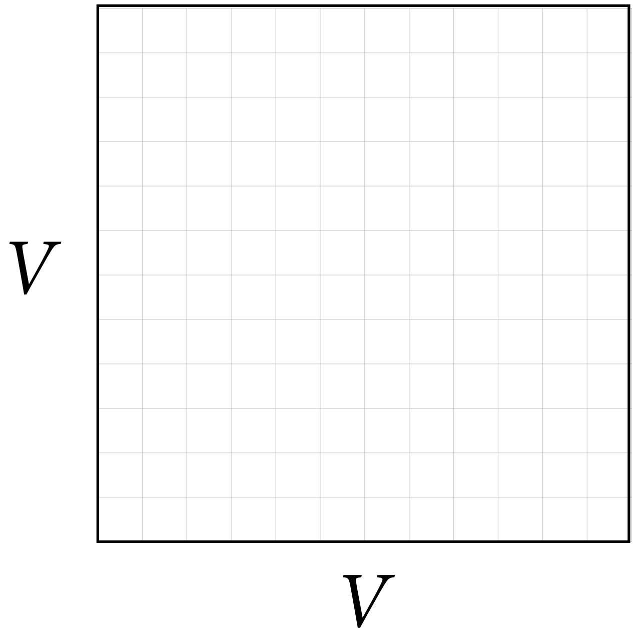
Atserias, Bulatov and Dawar (2008)

Dawar, Grohe, H., Laubner (2009)

⇝ Study pebble games with linear-algebraic game rules

Realising matrices over a finite graph

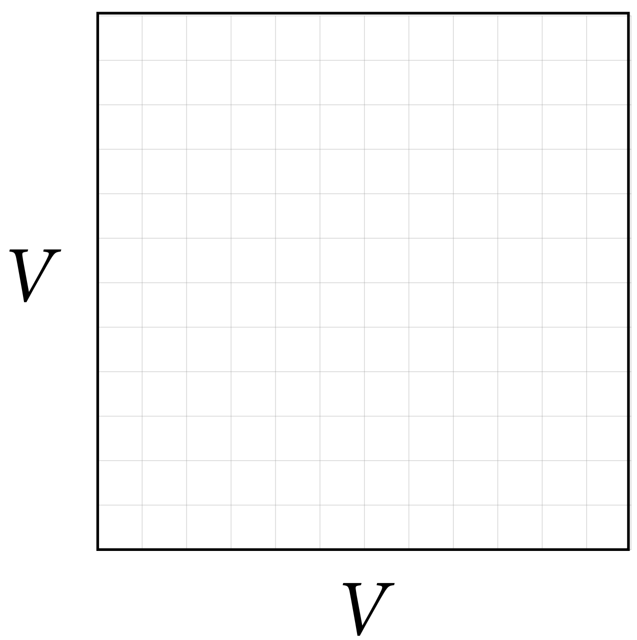
Let $G = (V, E)$ be a graph



Realising matrices over a finite graph

Let $G = (V, E)$ be a graph

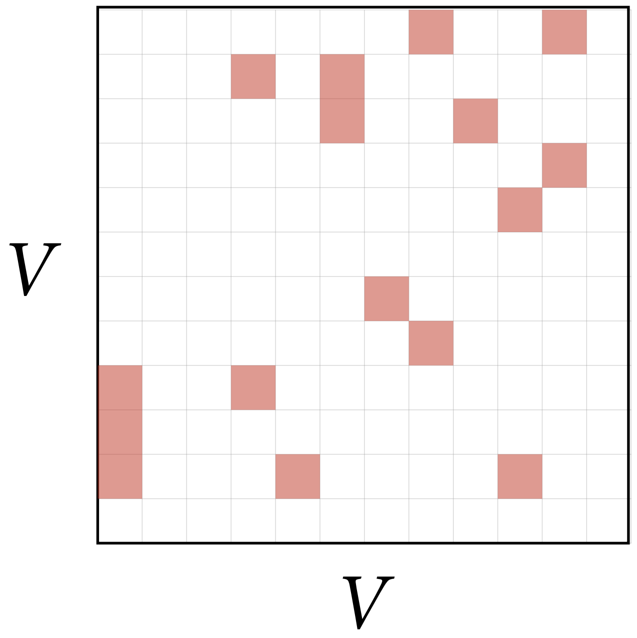
Subset of $V \times V$



Realising matrices over a finite graph

Let $G = (V, E)$ be a graph

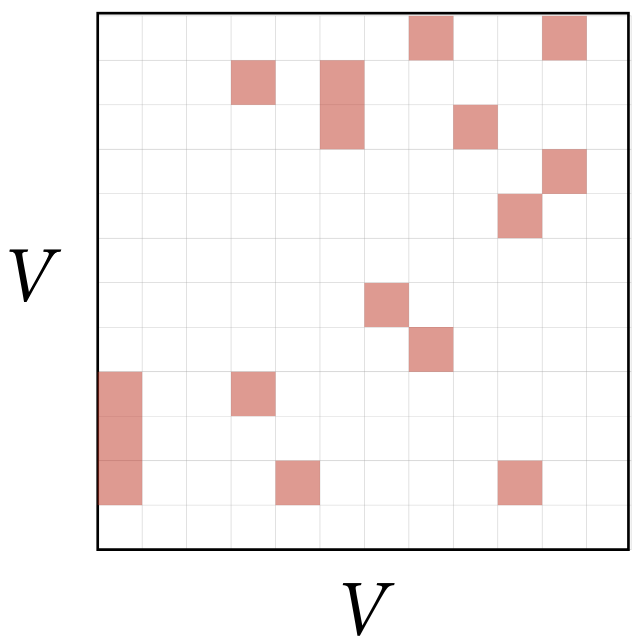
Subset of $V \times V$



Realising matrices over a finite graph

Let $G = (V, E)$ be a graph

Subset of $V \times V$ $\leadsto$ $\{0,1\}$-matrix, rows and columns indexed by V

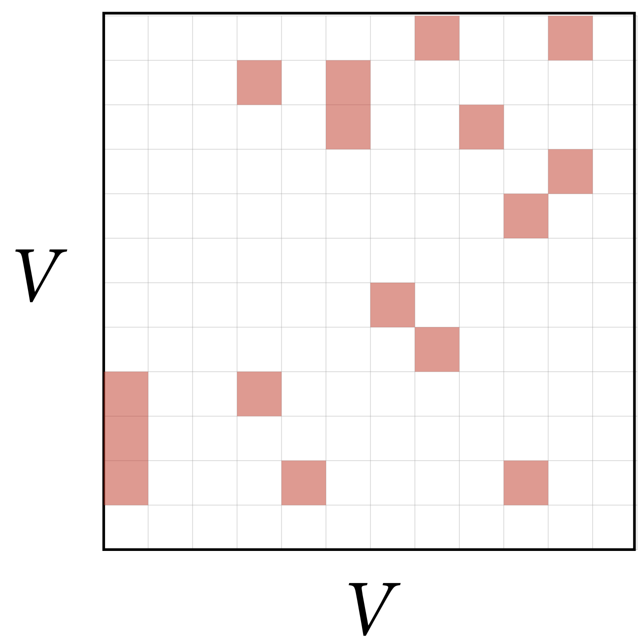


Realising matrices over a finite graph

Let $G = (V, E)$ be a graph

Subset of $V \times V$ $\rightsquigarrow$ $\{0,1\}$-matrix, rows and columns indexed by V

Example: Adjacency matrix of G — induced by $E \subseteq V \times V$

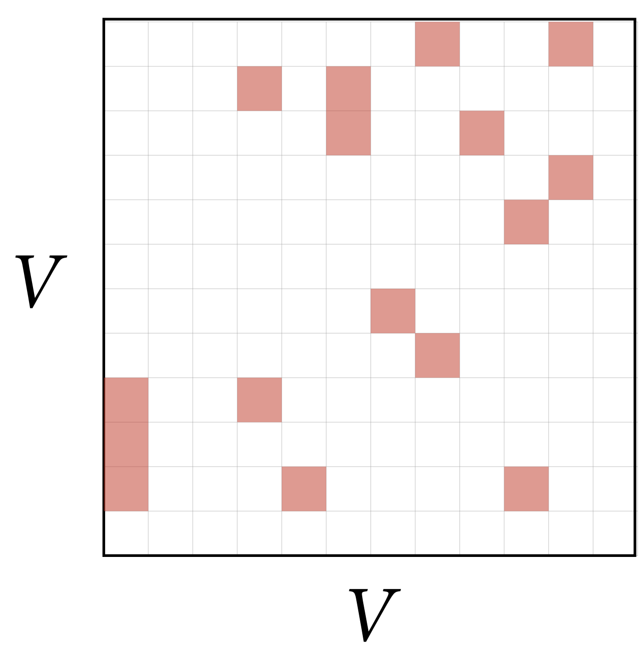


Realising matrices over a finite graph

Let $G = (V, E)$ be a graph

Subset of $V \times V$ $\rightsquigarrow$ $\{0,1\}$-matrix, rows and columns indexed by V

Example: Adjacency matrix of G — induced by $E \subseteq V \times V$



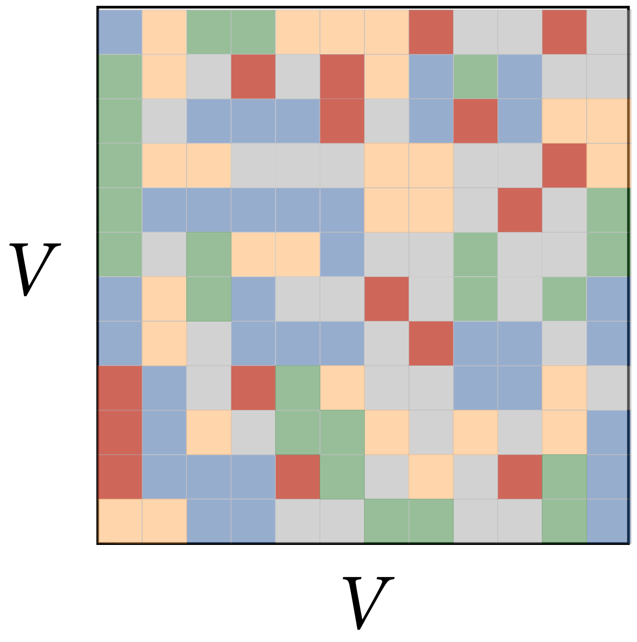
Partition of $V \times V$

Realising matrices over a finite graph

Let $G = (V, E)$ be a graph

Subset of $V \times V$ ⟶ $\{0,1\}$-matrix, rows and columns indexed by V

Example: Adjacency matrix of G — induced by $E \subseteq V \times V$



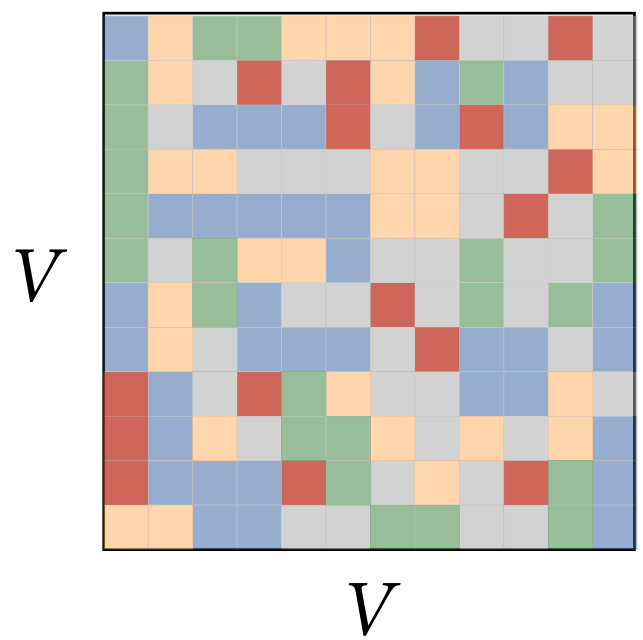
Partition of $V \times V$

Realising matrices over a finite graph

Let $G = (V, E)$ be a graph

Subset of $V \times V$ ⟶ $\{0,1\}$-matrix, rows and columns indexed by V

Example: Adjacency matrix of G — induced by $E \subseteq V \times V$



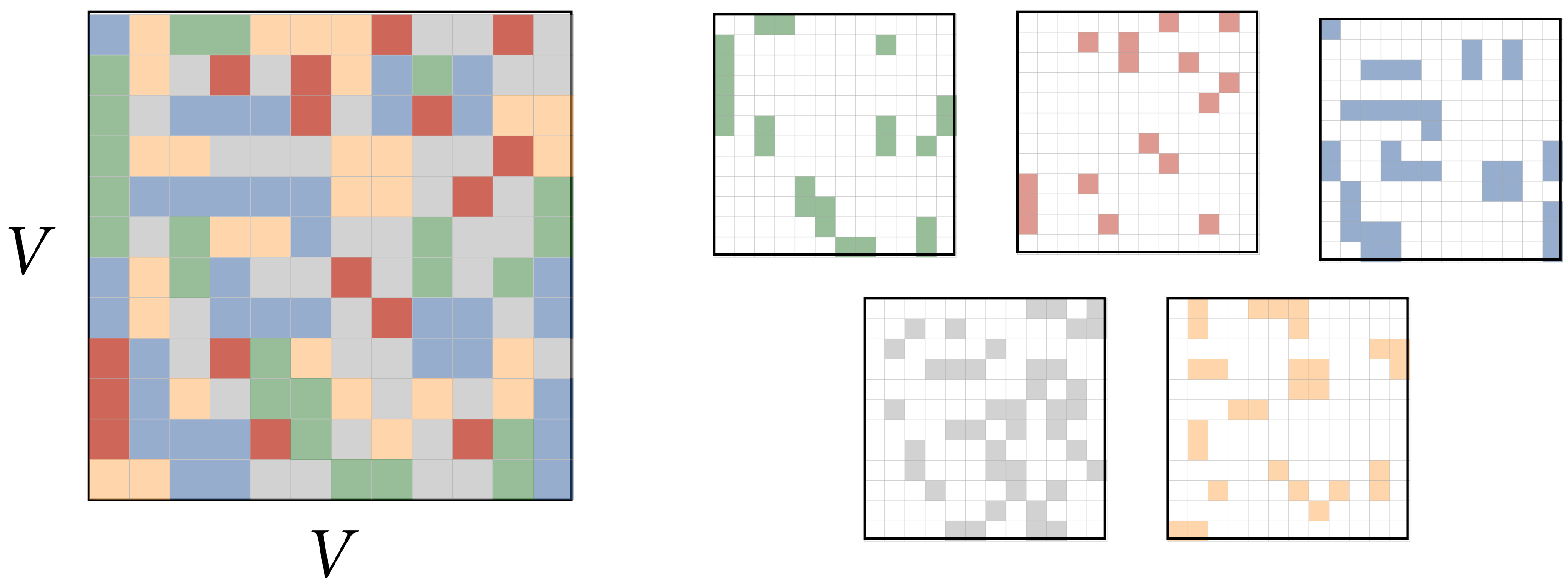
Partition of $V \times V$ ⟶ Family of pairwise disjoint $\{0,1\}$-matrices

Realising matrices over a finite graph

Let $G = (V, E)$ be a graph

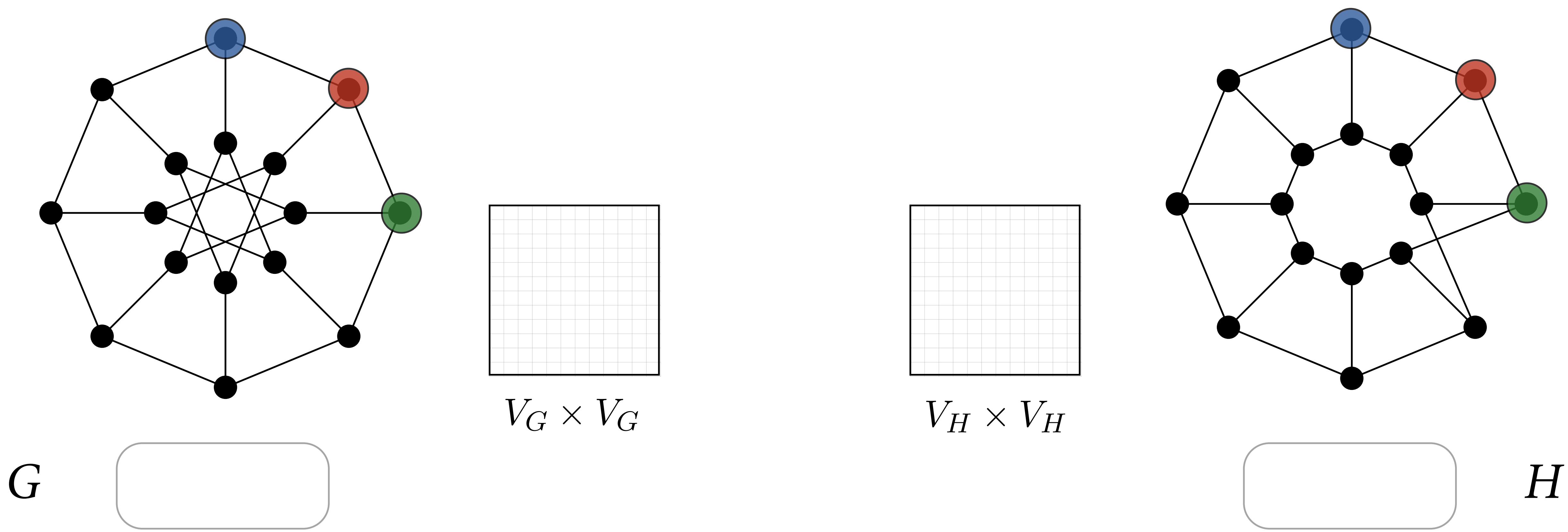
Subset of $V \times V$ ⟶ $\{0,1\}$-matrix, rows and columns indexed by V

Example: Adjacency matrix of G — induced by $E \subseteq V \times V$



Partition of $V \times V$ ⟶ Family of pairwise disjoint $\{0,1\}$-matrices

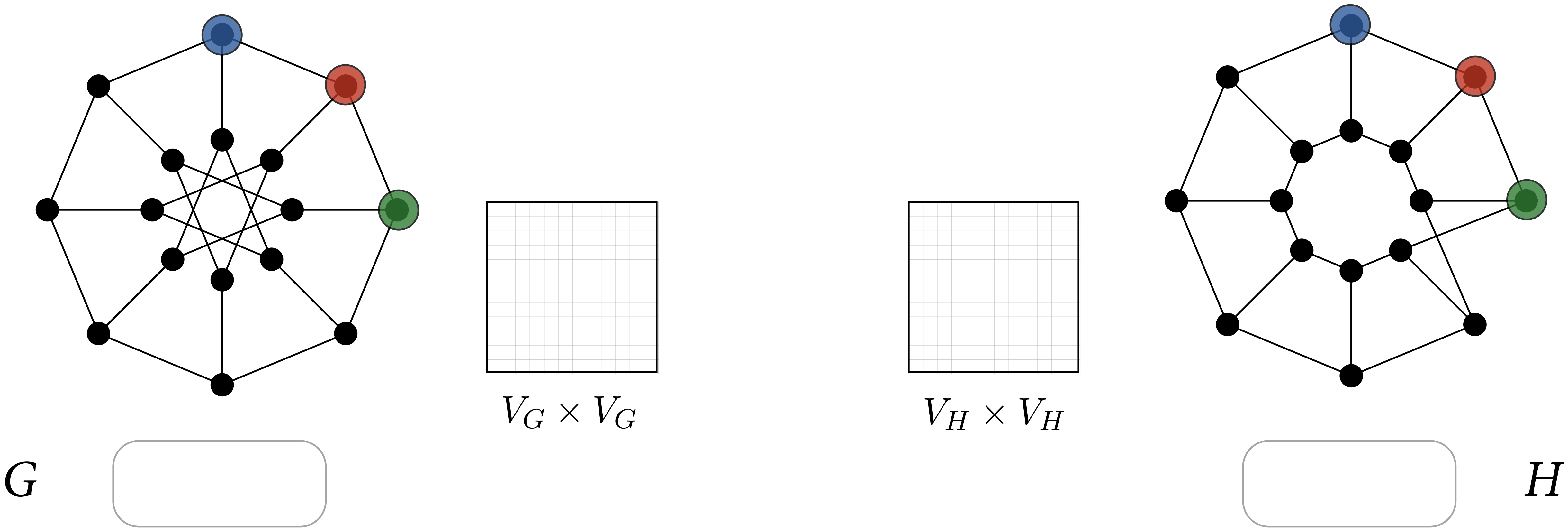
Pebble game based on invertible linear maps



Spoiler

Duplicator

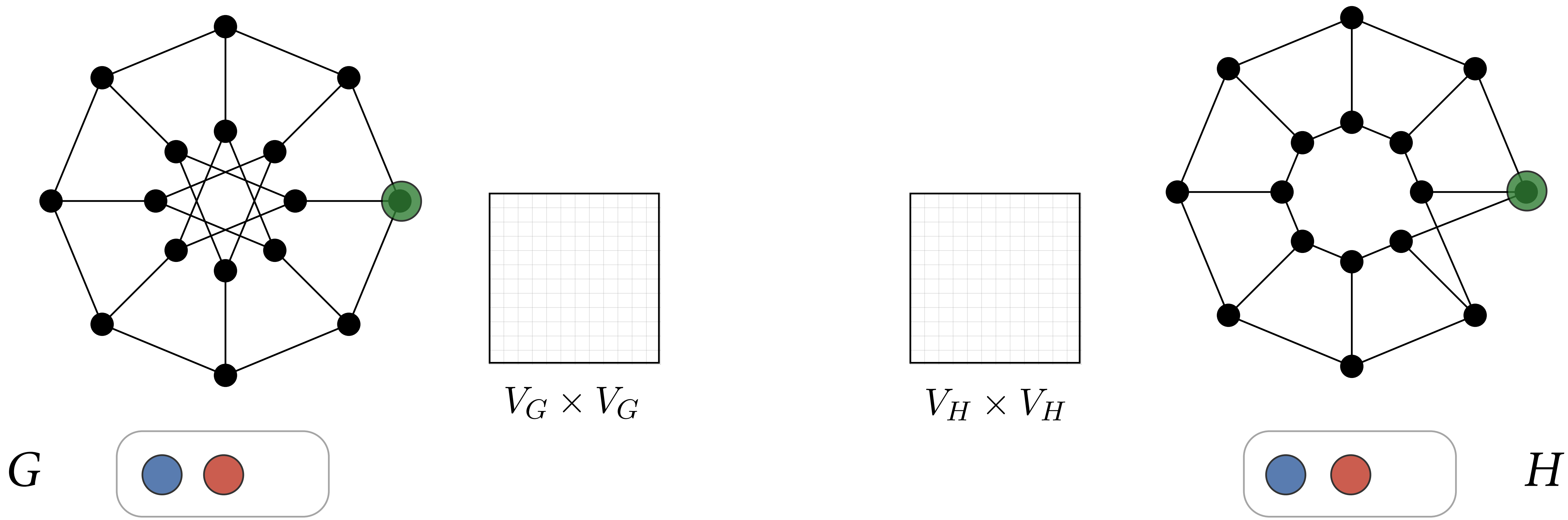
Pebble game based on invertible linear maps



Spoiler removes two pairs of corresponding pebbles

Duplicator

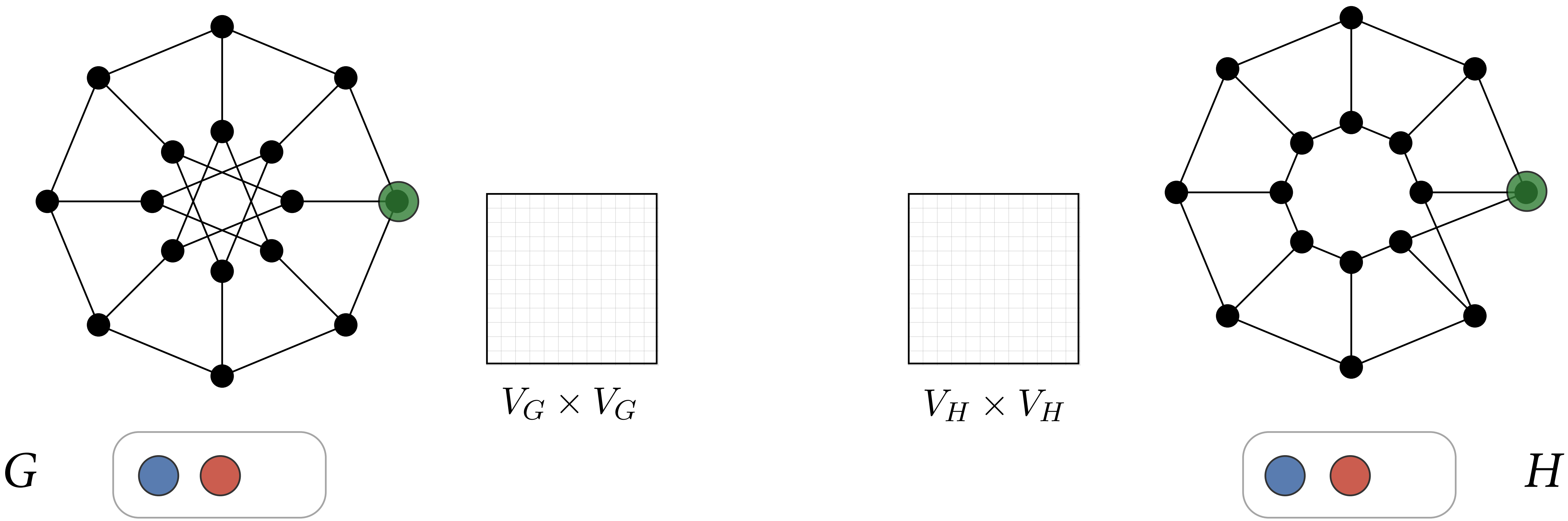
Pebble game based on invertible linear maps



Spoiler removes two pairs of corresponding pebbles

Duplicator

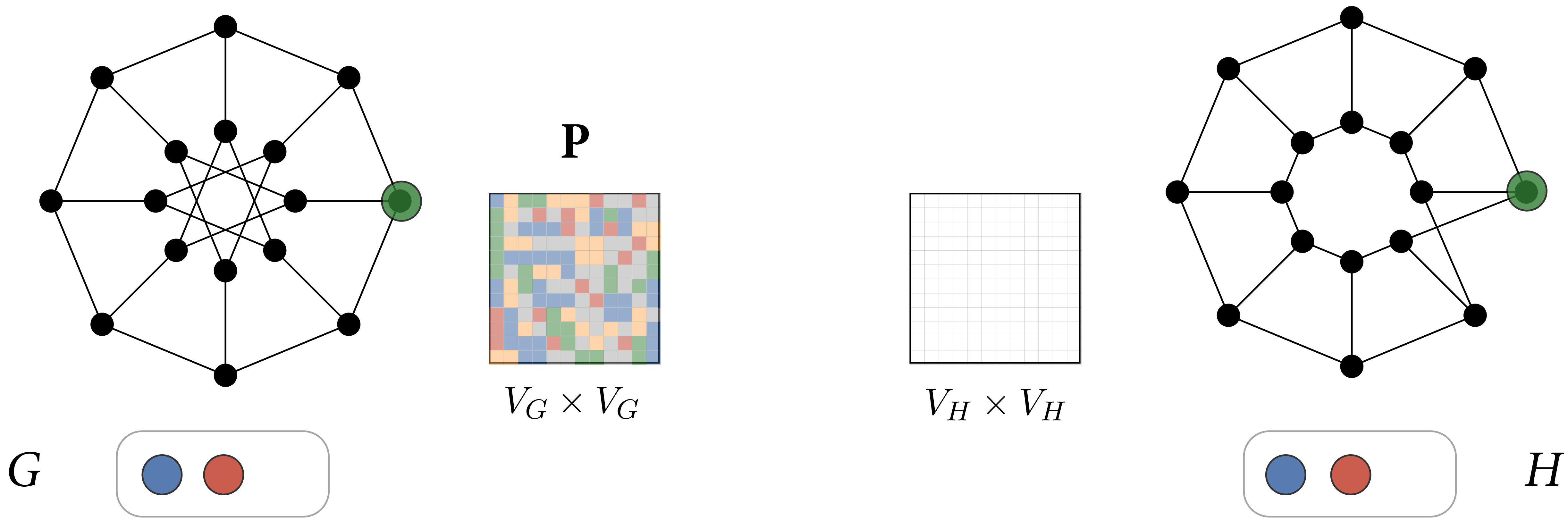
Pebble game based on invertible linear maps



Spoiler

Duplicator ▸ gives a partition **P** of $V_G \times V_G$,

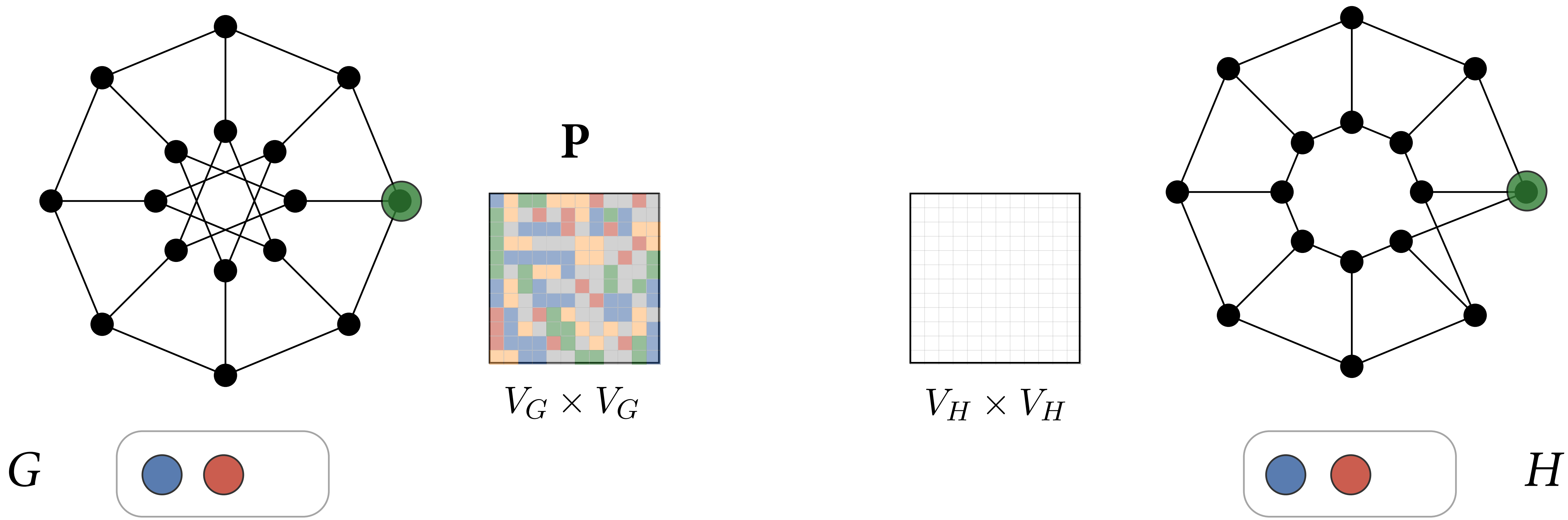
Pebble game based on invertible linear maps



Spoiler

Duplicator ▸ gives a partition **P** of $V_G \times V_G$,

Pebble game based on invertible linear maps

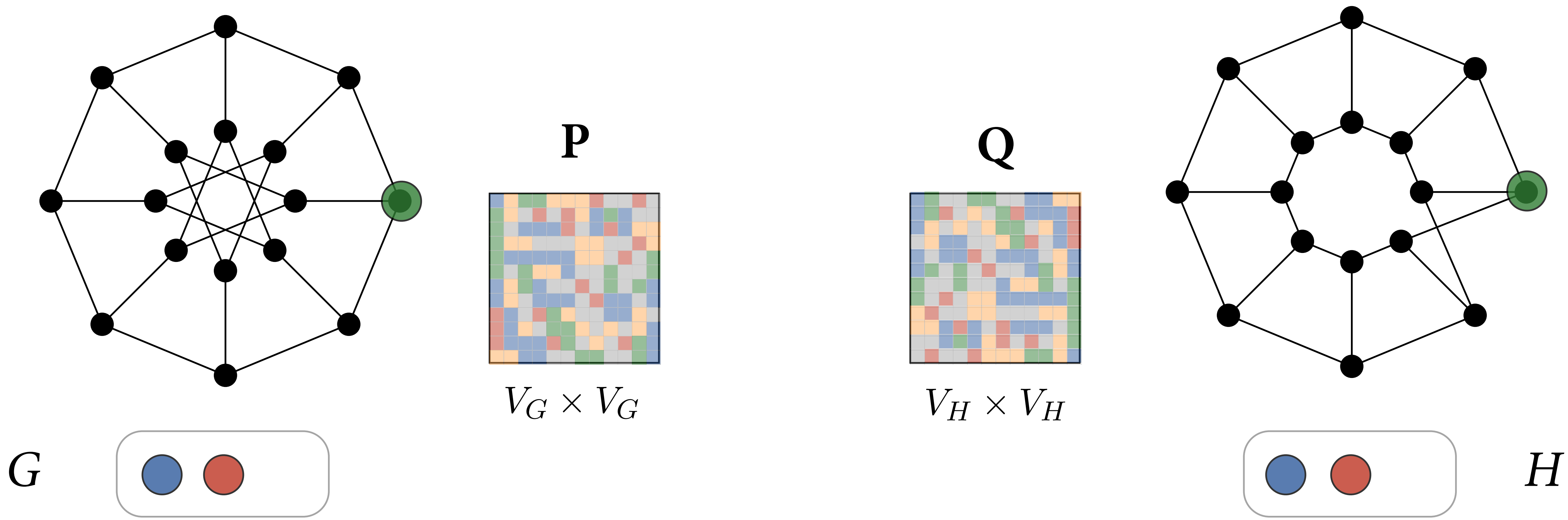


Spoiler

Duplicator

- gives a partition **P** of $V_G \times V_G$,
- a partition **Q** of $V_H \times V_H$

Pebble game based on invertible linear maps

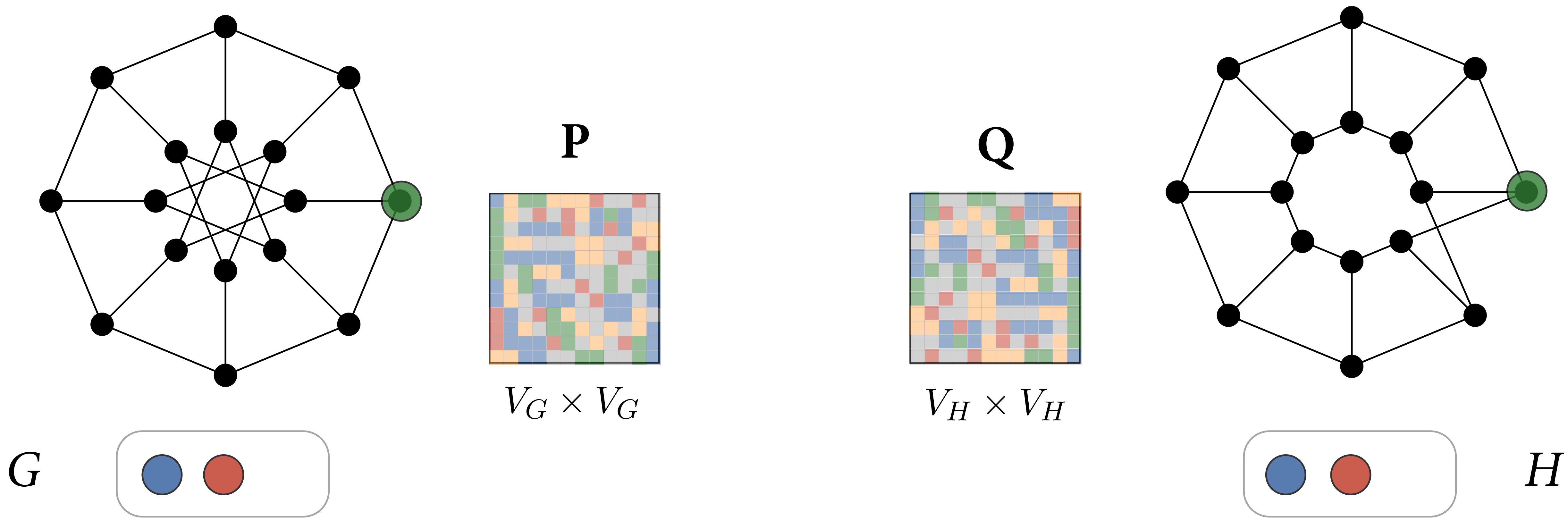


Spoiler

Duplicator

- gives a partition **P** of $V_G \times V_G$,
- a partition **Q** of $V_H \times V_H$

Pebble game based on invertible linear maps

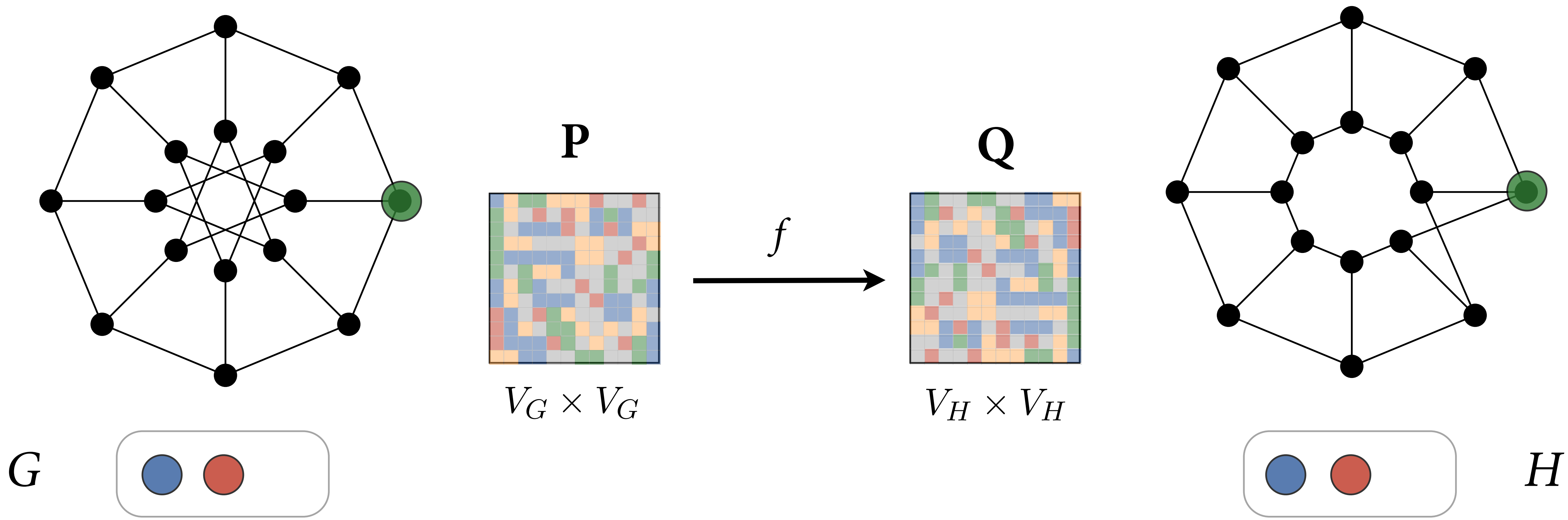


Spoiler

Duplicator

- gives a partition **P** of $V_G \times V_G$,
- a partition **Q** of $V_H \times V_H$
- and a bijection f: **P** → **Q** such that...

Pebble game based on invertible linear maps

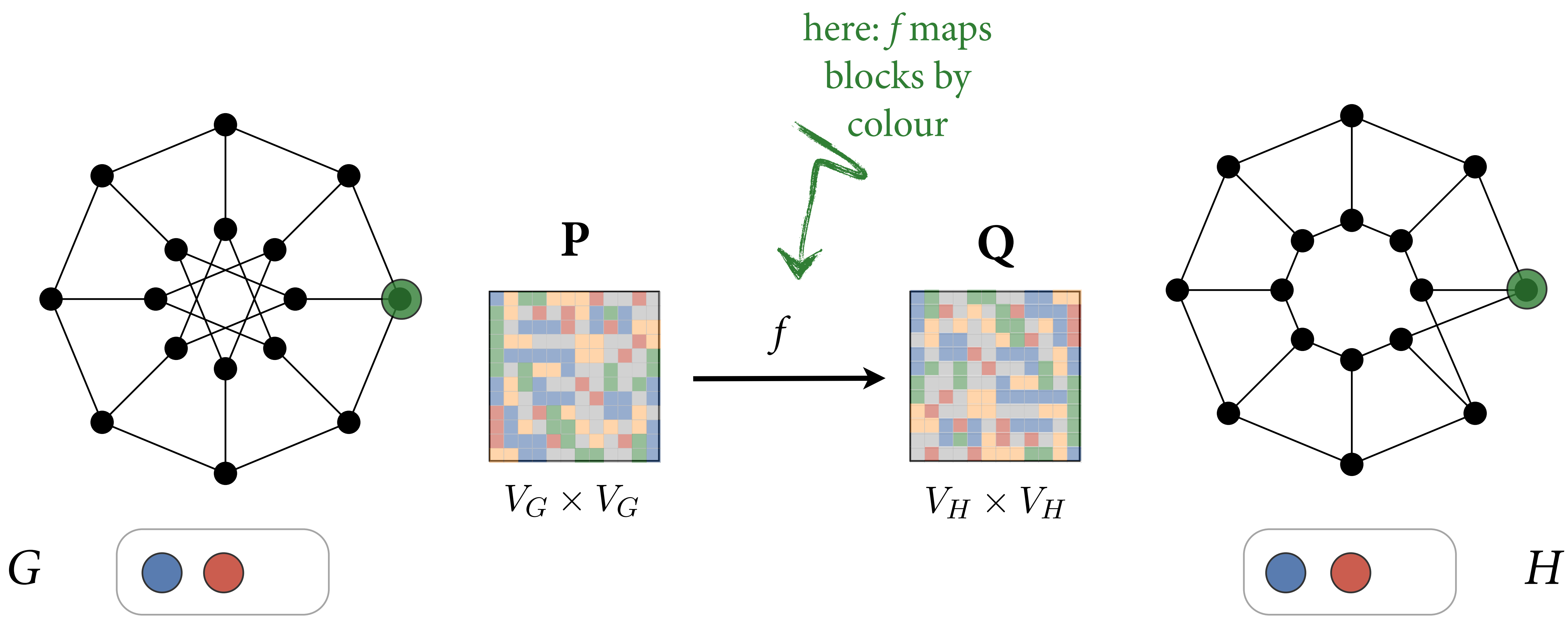


Spoiler

Duplicator

- gives a partition **P** of $V_G \times V_G$,
- a partition **Q** of $V_H \times V_H$
- and a bijection f: **P** → **Q** such that...

Pebble game based on invertible linear maps

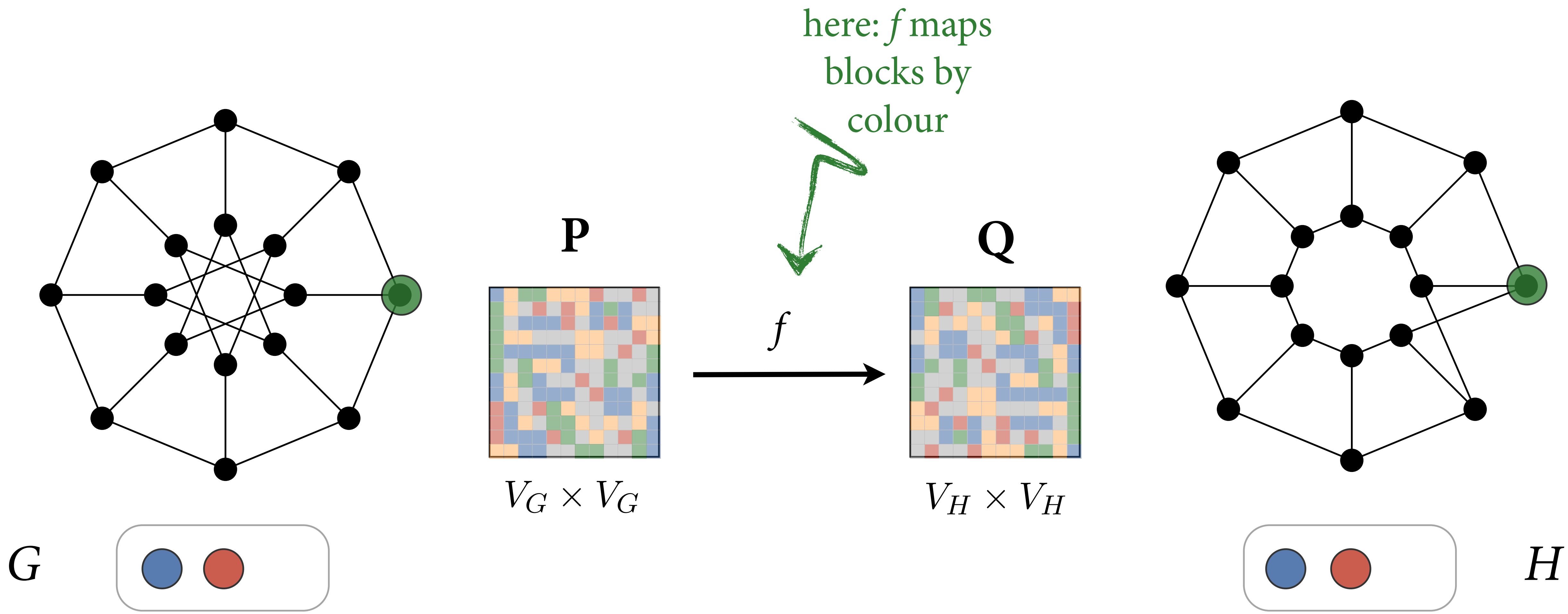


Spoiler

Duplicator

- gives a partition **P** of $V_G \times V_G$,
- a partition **Q** of $V_H \times V_H$
- and a bijection f: **P** → **Q** such that...

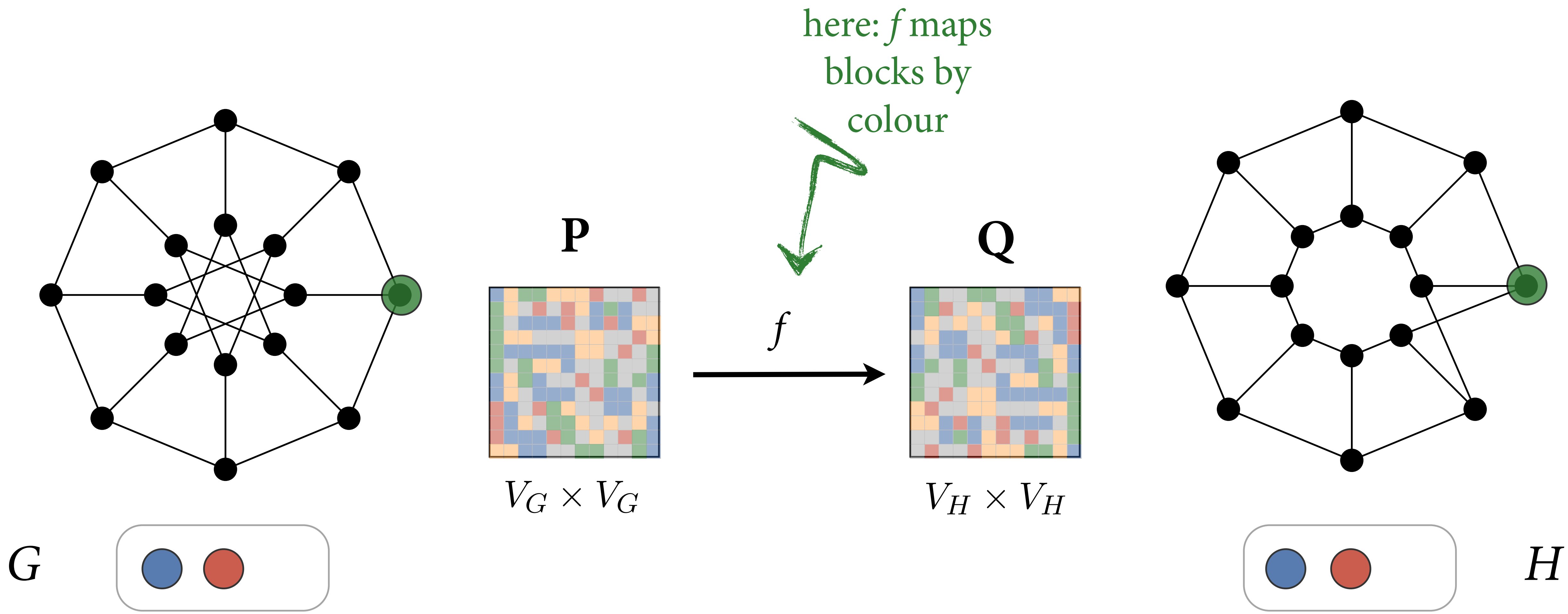
Pebble game based on invertible linear maps



Spoiler

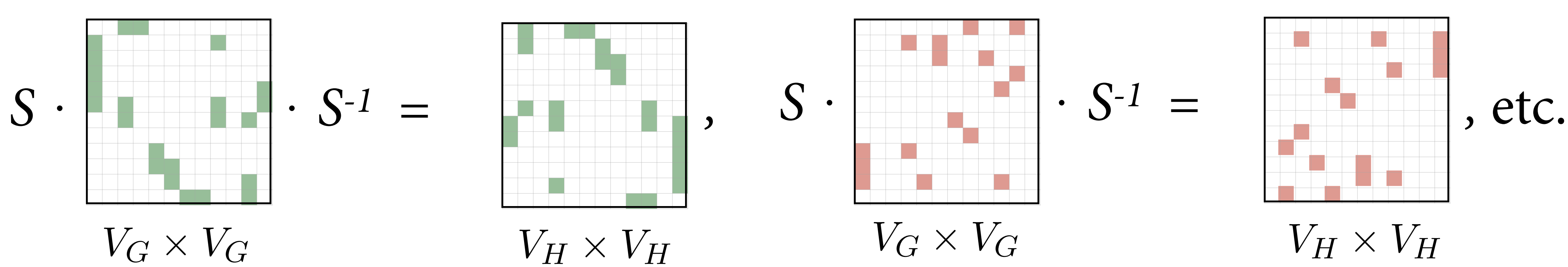
Duplicator ▹ ... there is an invertible linear map S over GF(2) for which it holds that

Pebble game based on invertible linear maps

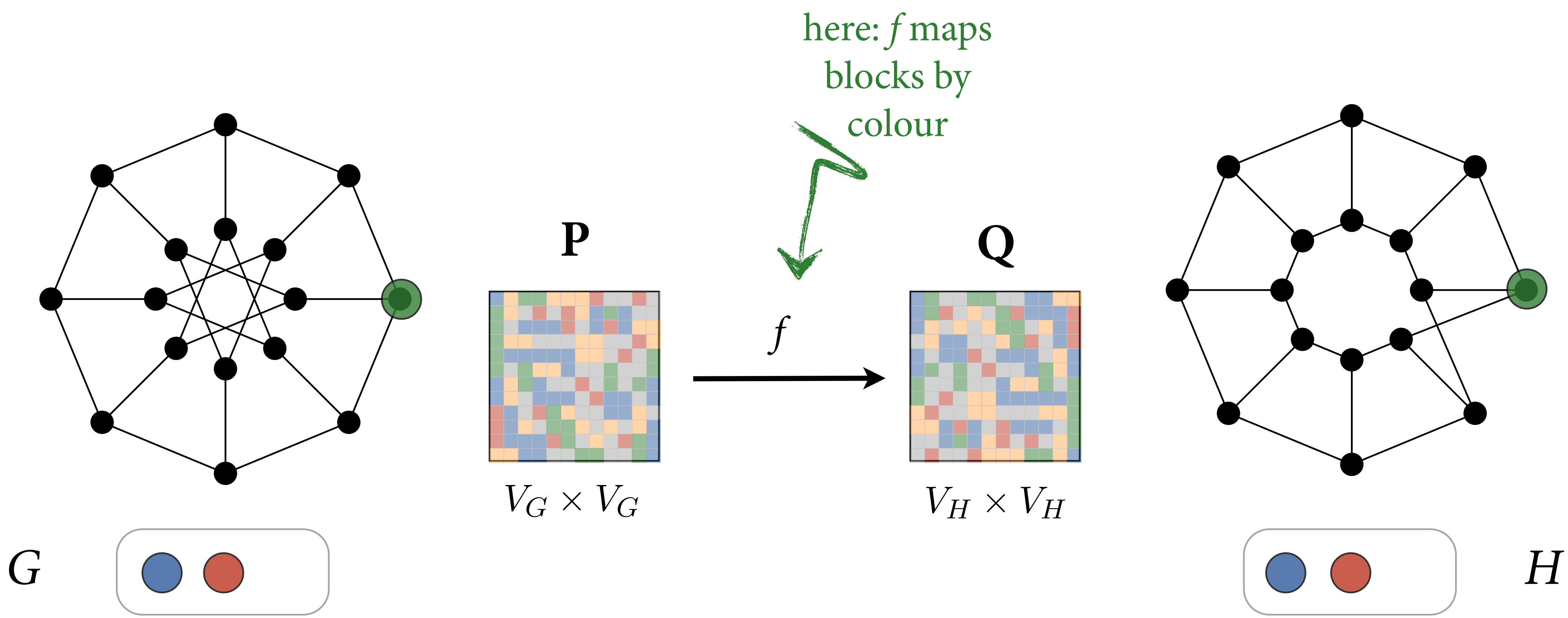


Spoiler

Duplicator ▹ ... there is an invertible linear map S over GF(2) for which it holds that



Pebble game based on invertible linear maps

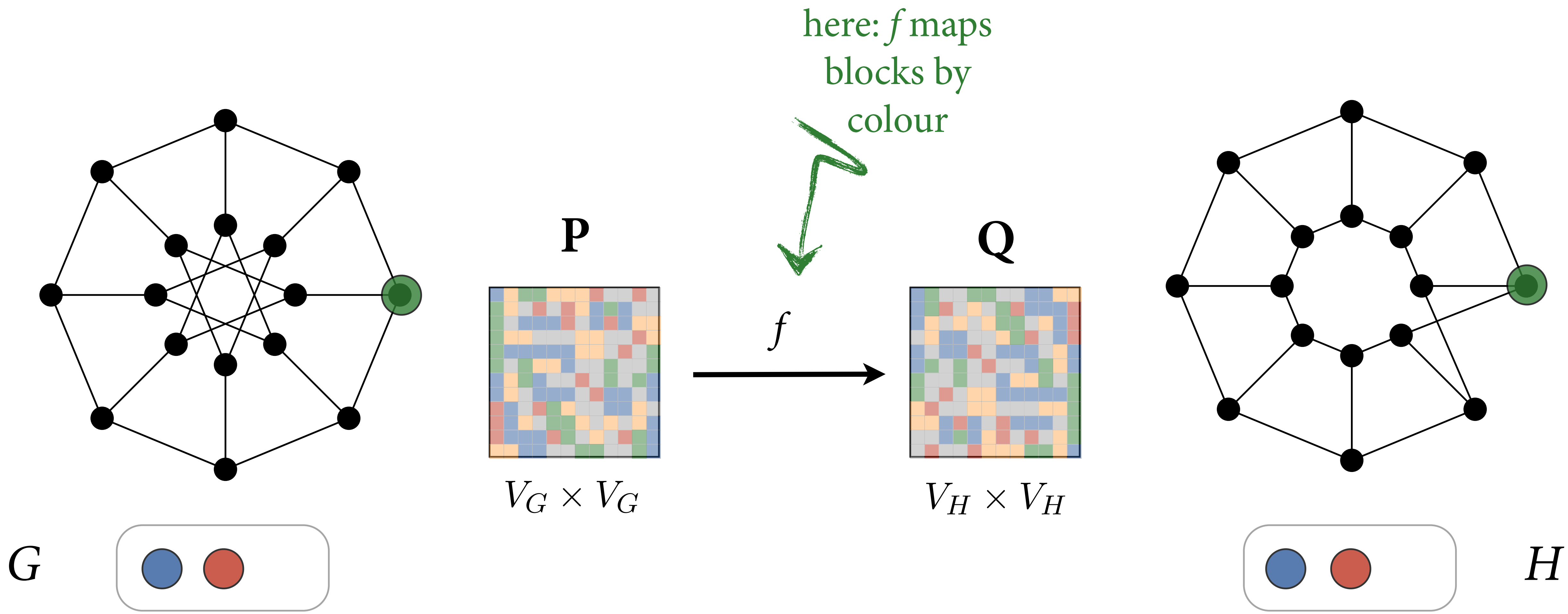


Spoiler

Duplicator ▷ … there is an invertible linear map S over GF(2) for which it holds that

$$S \cdot M \cdot S^{-1} = f(M) \qquad \forall M \in \mathbf{P}$$

Pebble game based on invertible linear maps



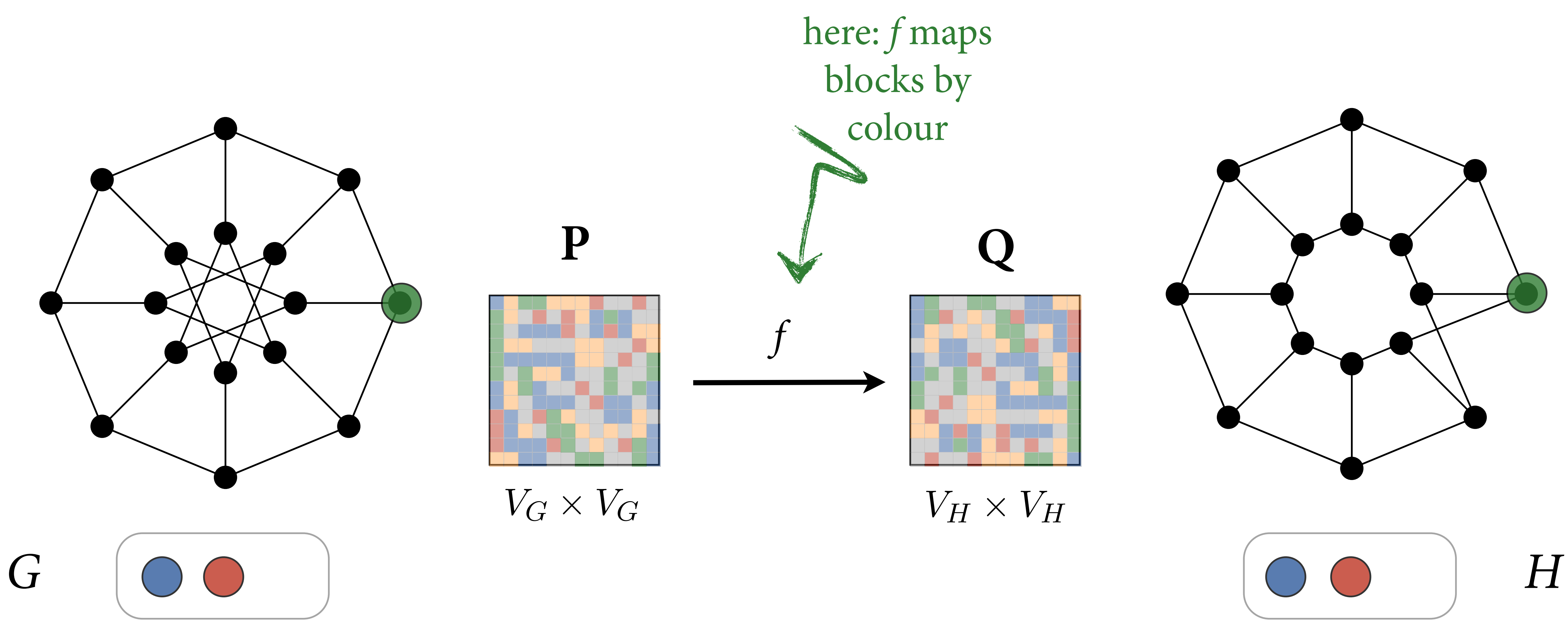
Spoiler

Duplicator ▷ ... there is an invertible linear map S over GF(2) for which it holds that

$$S \cdot M \cdot S^{-1} = f(M) \quad \forall M \in \mathbf{P}$$

(S is a similarity transformation for each M and $f(M)$)

Pebble game based on invertible linear maps



Spoiler

Duplicator

places the chosen pebbles over G on elements of some block M in **P** and the corresponding pebbles over H on the elements of $f(M)$ in **Q**

Strengths of the invertible-map game

More generally: consider games played over finite fields GF(p) for any prime p in a finite set Ω.

$G \approx_{\Omega}^{k} H$ Duplicator has a winning strategy in the k-pebble invertible-map game on G and H with primes Ω.

- For each k and all Ω, $\approx_{\Omega}^{k+1}$ is a refinement of $\equiv^{C^k}$
- For each k, there is a pair of non-isomorphic graphs that are equivalent under $\equiv^{C^k}$ but distinguished by $\approx_{\Omega}^{3}$ for any Ω.

H., Dawar (2012)

Application to graph isomorphism

There is an algorithm that, given graphs G and H of size n, decides whether $G \approx_{\Omega}^{k} H$ in time $n^{O(k)} \cdot p^{O(1)}$, where p is the largest prime in Ω.

H., Dawar (2012)

We get a family of polynomial-time algorithms IM_k (here for a fixed Ω) for which

- if IM_k distinguishes between G and H then IM_{k+1} also distinguishes between G and H (refinement)
- for each pair of graphs G and H, there is some k such that IM_k correctly decided isomorphism (limit)

Optimistic: is there is a fixed k for which IM_k = isomorphism?

answer is “no” if only consider basic version of the game

$\rightsquigarrow$ need to allow matrices indexed by tuples of vertices

From logics to games — and back again?

- Does the invertible-map game equivalence correspond to a "natural" logic?
- Does it coincide with isomorphism on classes of graphs that have polynomial-time isomorphism tests and for which C^k-equivalence is too weak ⇝ e.g. graphs of bounded degree, graphs of bounded colour class size?
- The "partition game" protocol can be adapted for any finite-variable logic with Lindström quantifiers ⇝ which kind of quantifier gives us a tractable instance of the partition game?