Generating abstractions

POLITECNICO
MILANO 1863

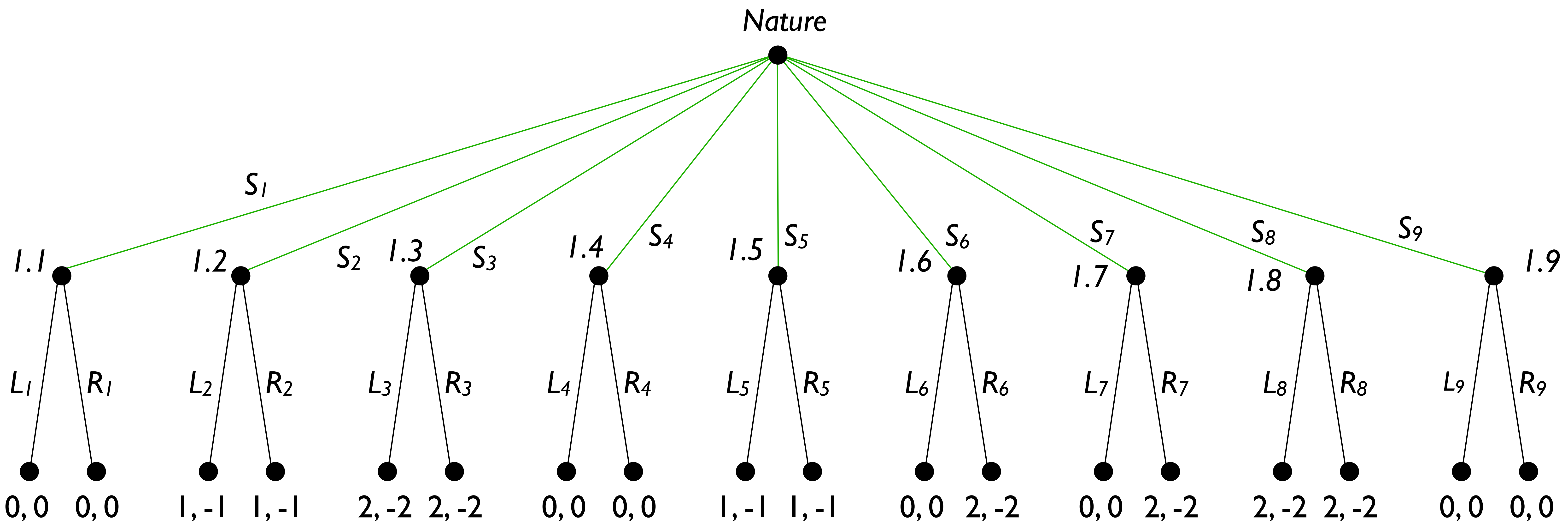
Lecturers: *Prof. Nicola Gatti, Dr. Andrea Celli, Dr. Alberto Marchesi*

Games with signals

Many information sets (states), due to the moves (signals) of the Nature

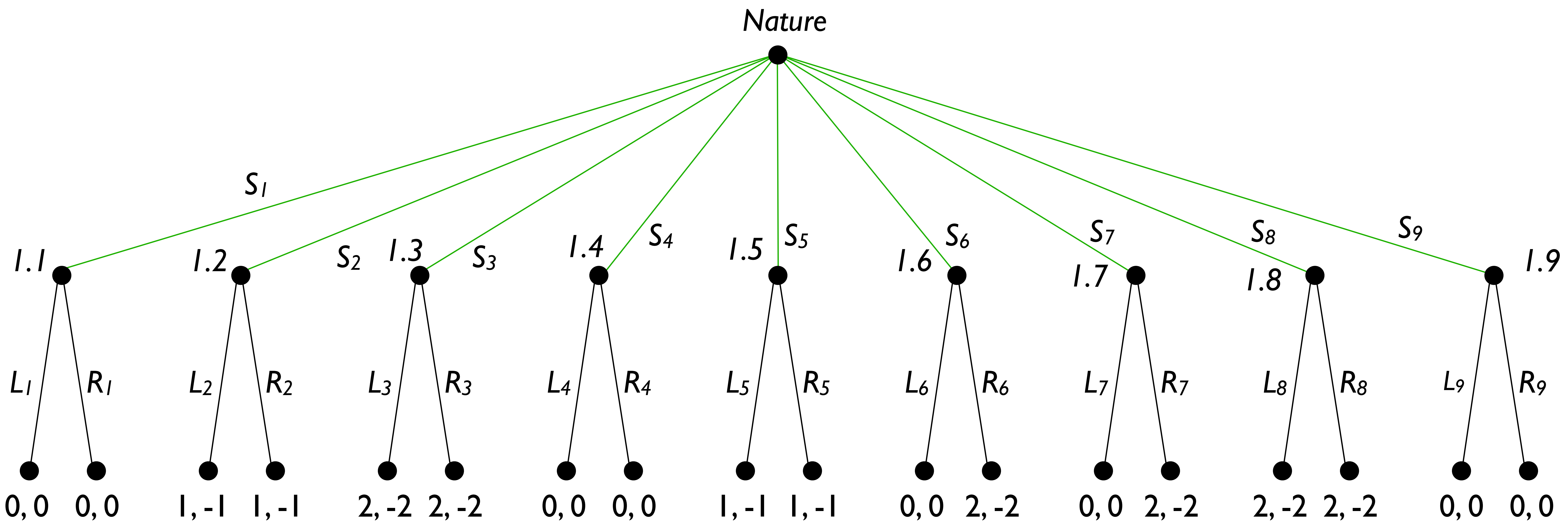
Games with signals

Many information sets (states), due to the moves (signals) of the Nature

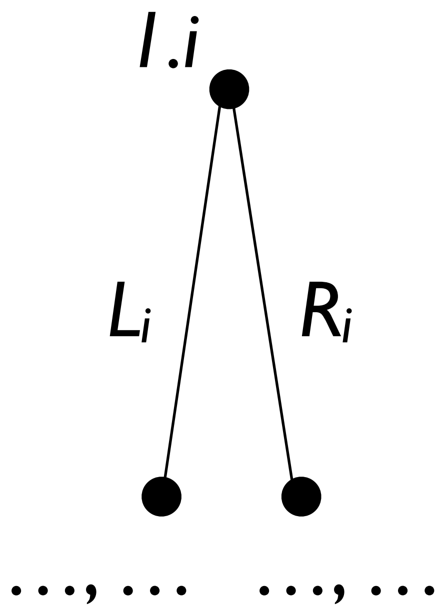


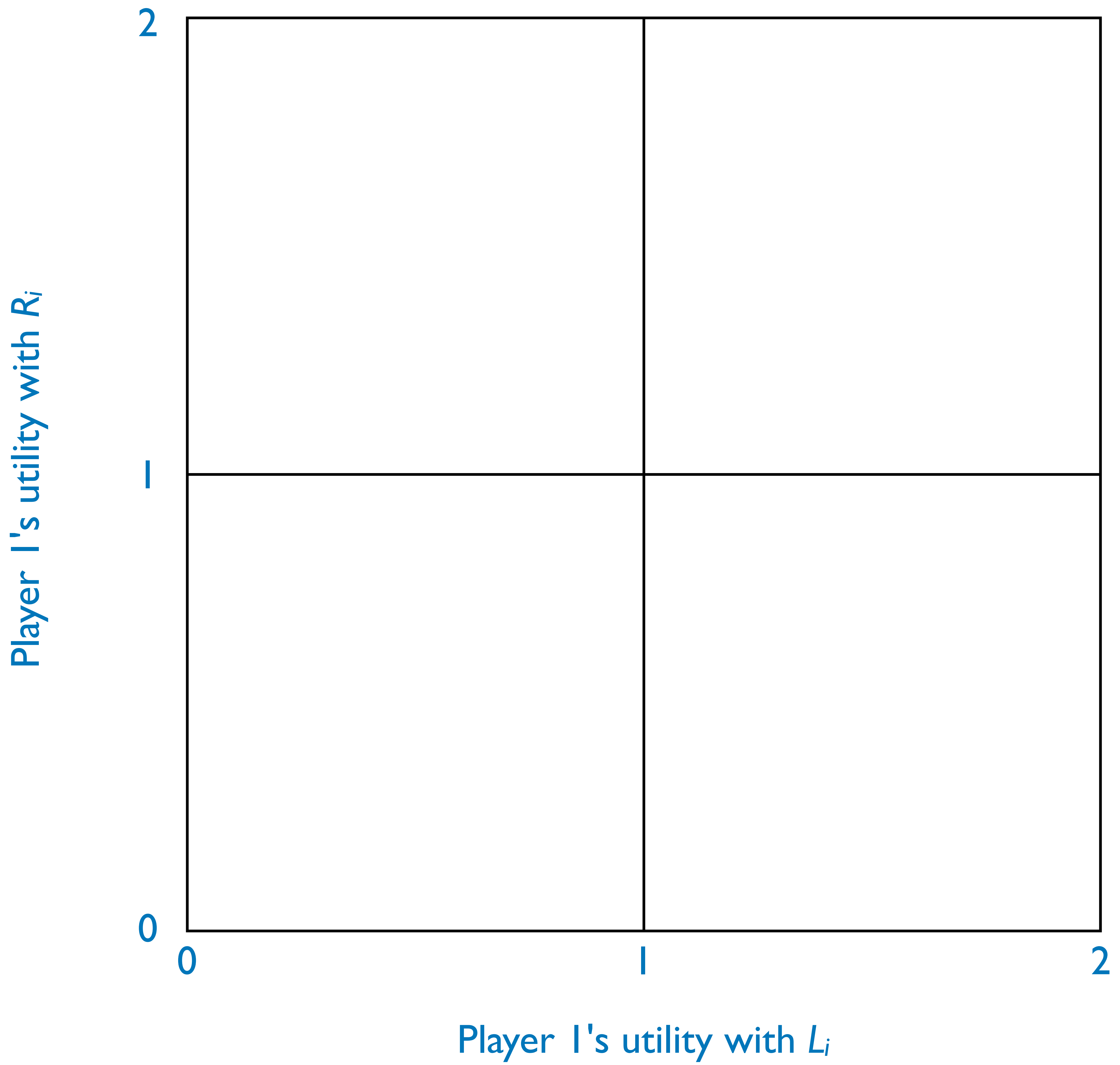
State aggregation (without information loss)

Merge two or more information sets if they lead to the same payoffs

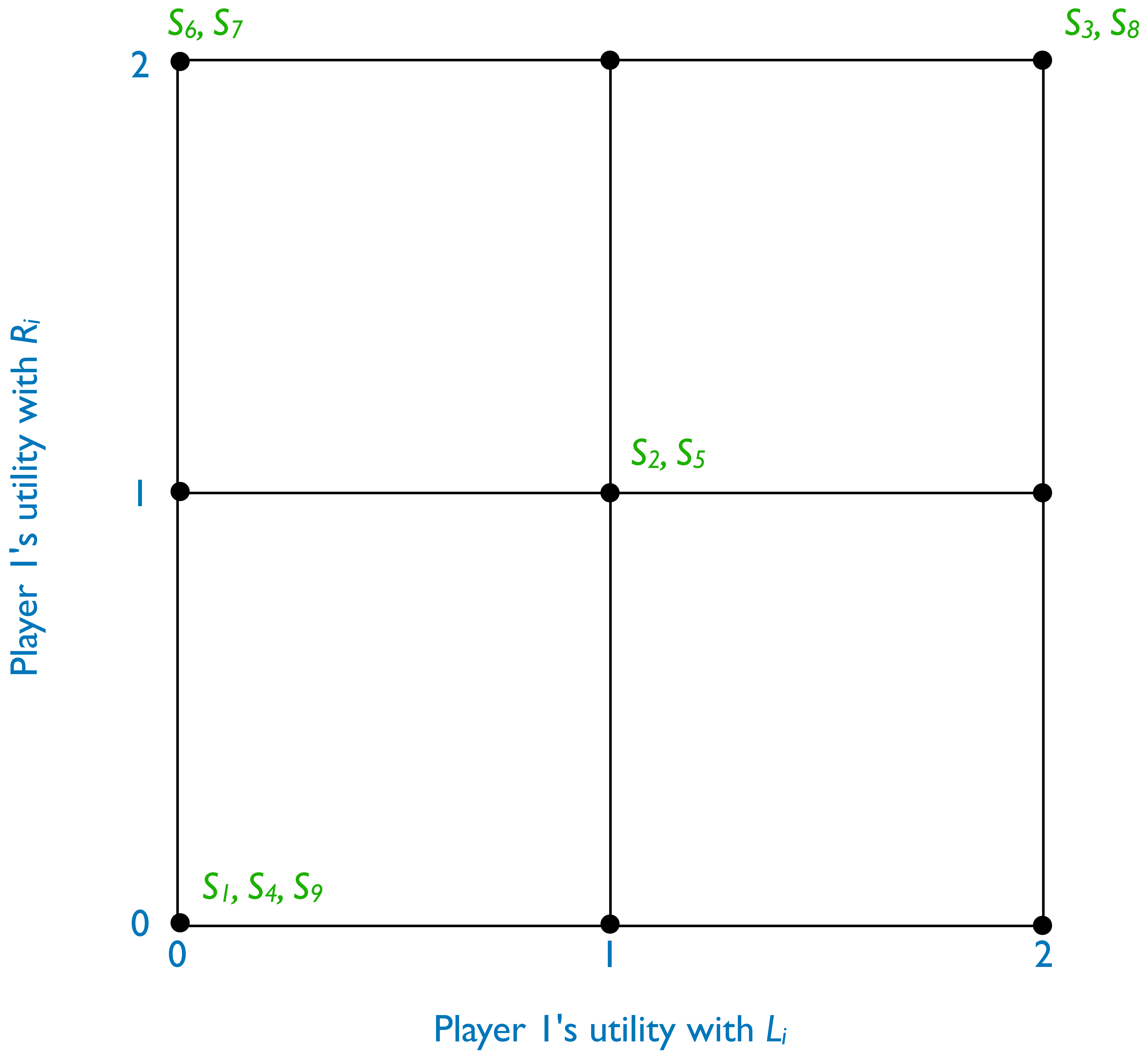


Payoff space

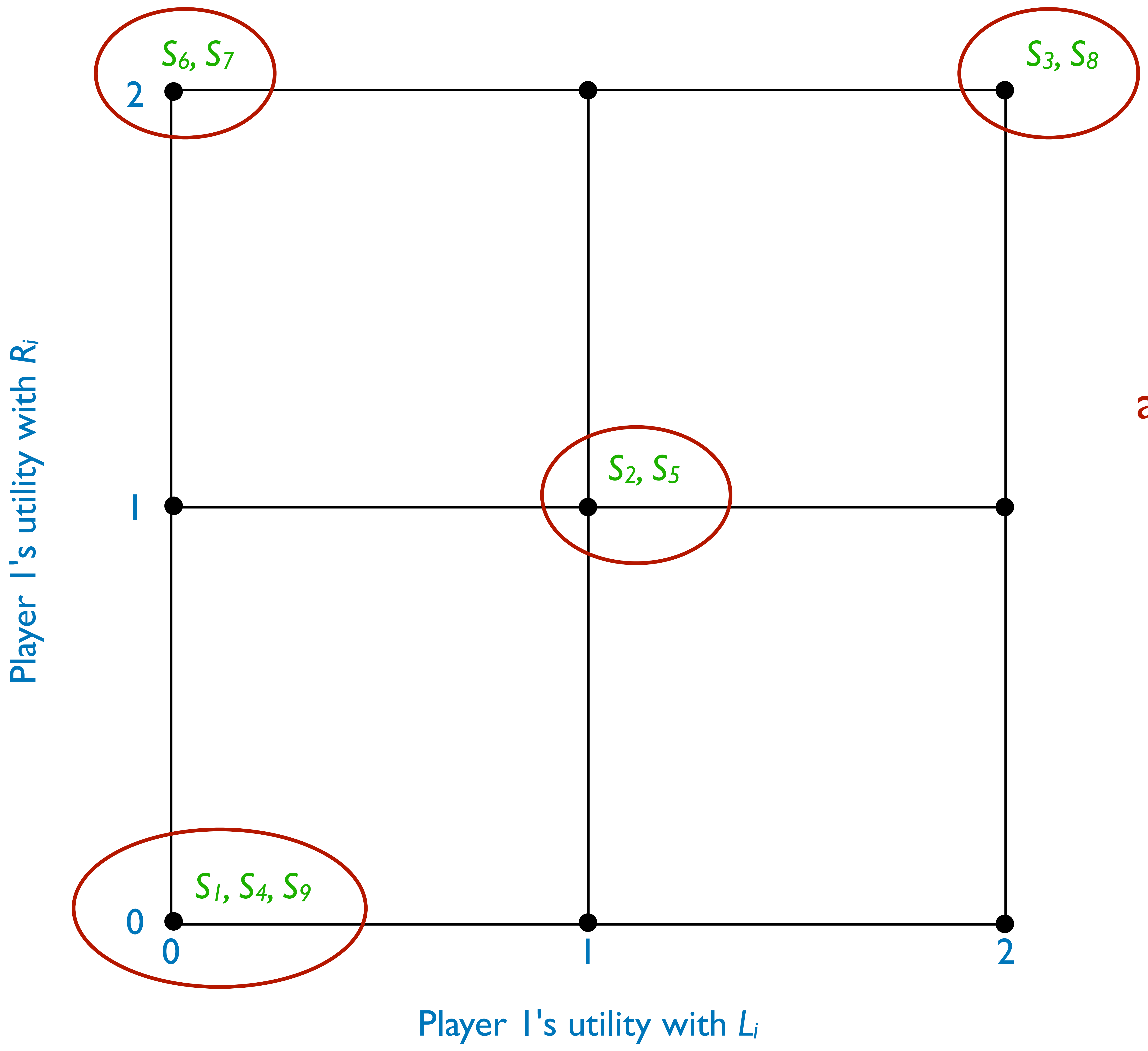




Payoff space



Payoff space

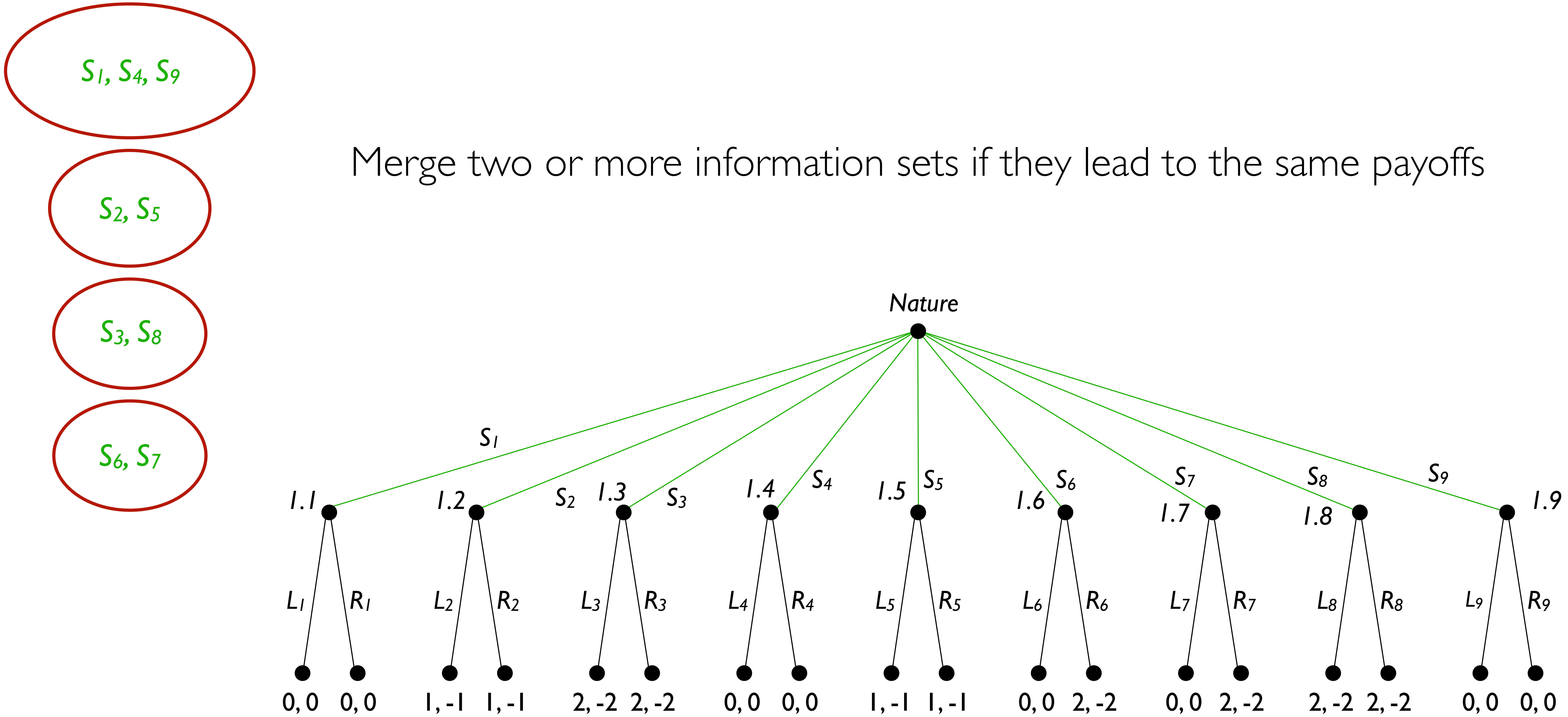


Merging these signals in a unique information set is safe

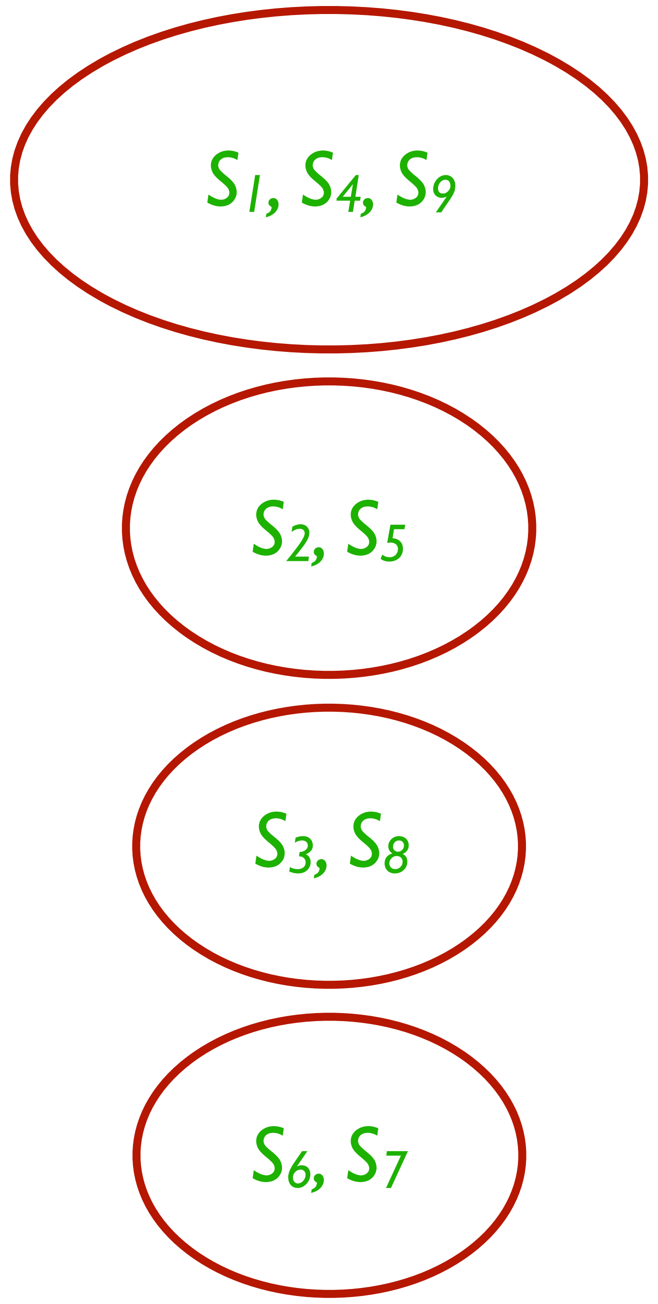
No utility loss

State aggregation (without information loss)

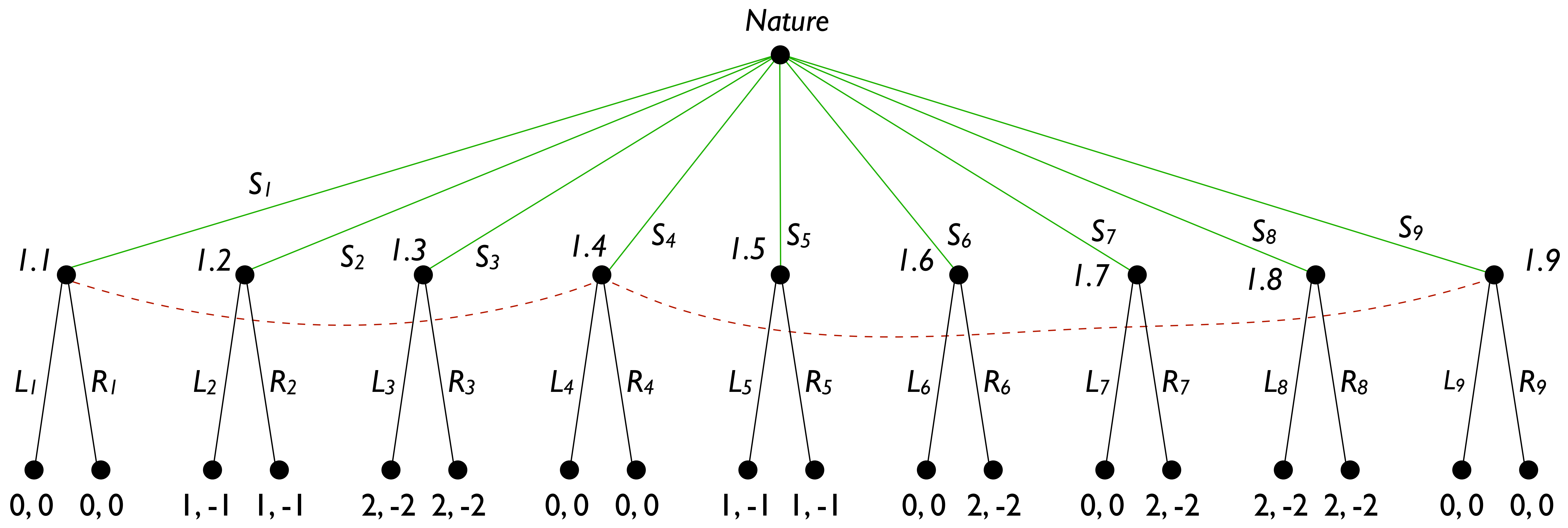
Merge two or more information sets if they lead to the same payoffs



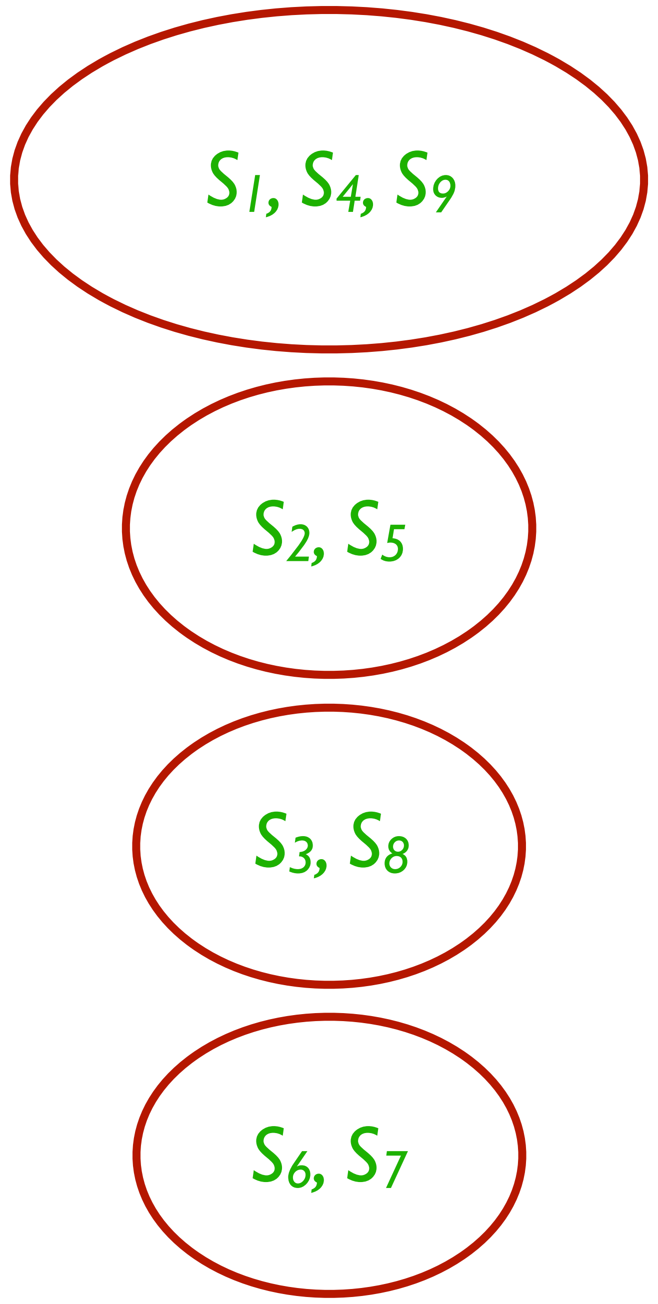
State aggregation (without information loss)



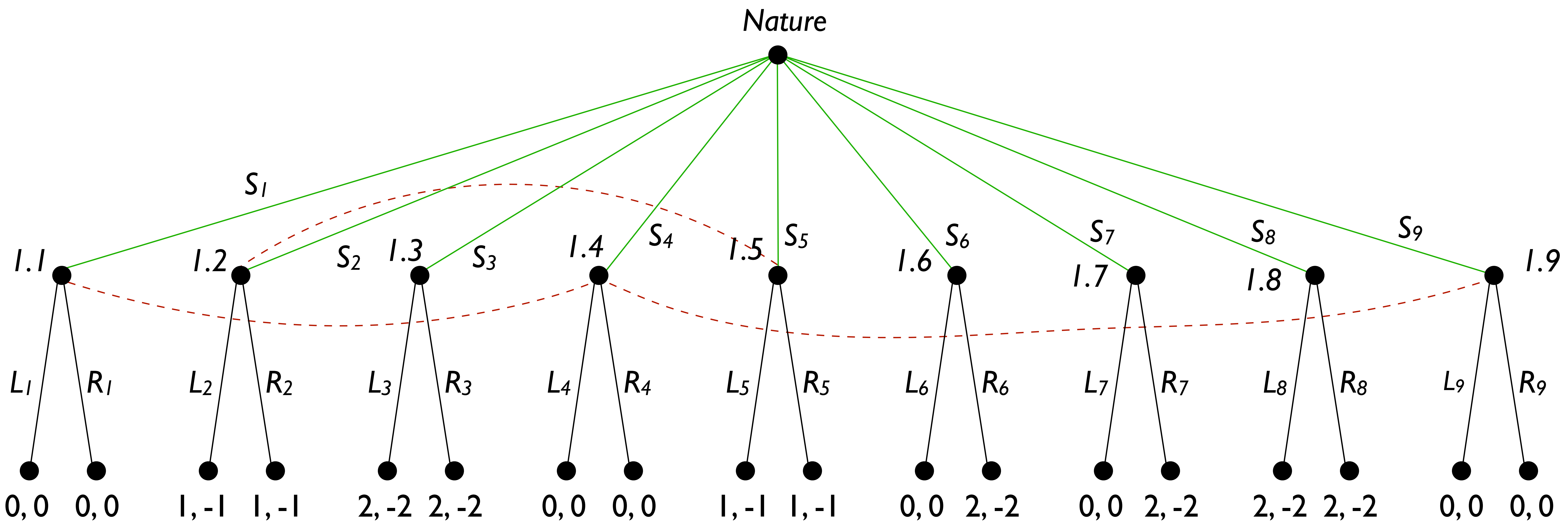
Merge two or more information sets if they lead to the same payoffs



State aggregation (without information loss)



Merge two or more information sets if they lead to the same payoffs



State aggregation (without information loss)

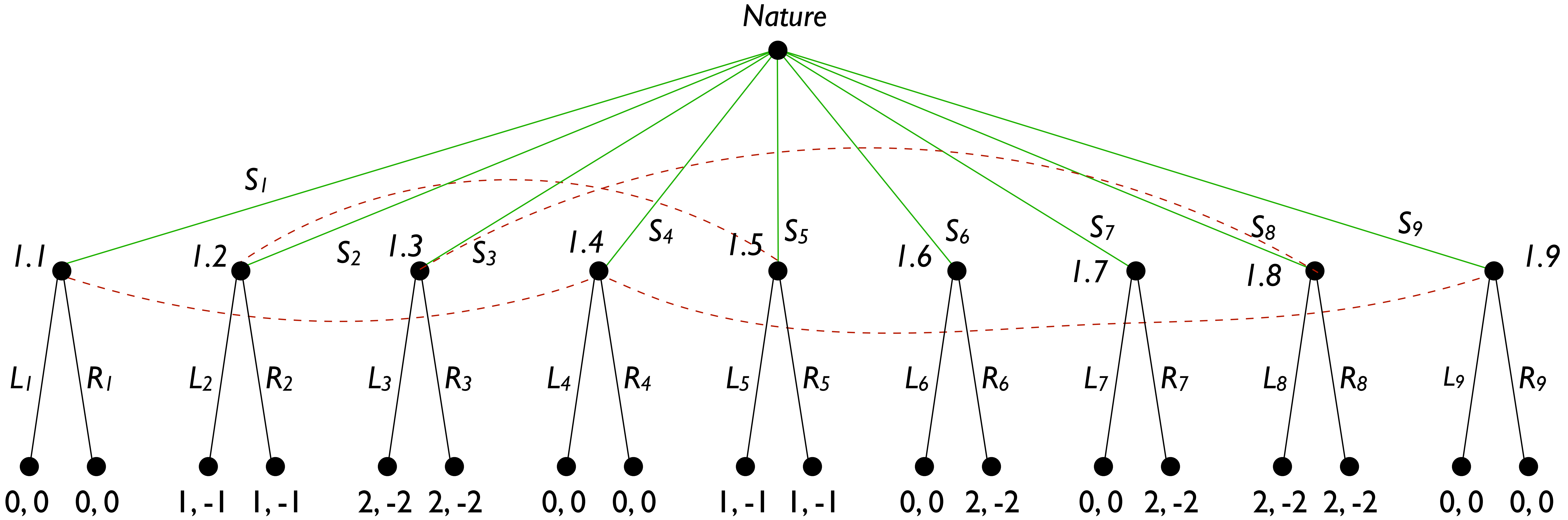
S_1, S_4, S_9

S_2, S_5

S_3, S_8

S_6, S_7

Merge two or more information sets if they lead to the same payoffs



State aggregation (without information loss)

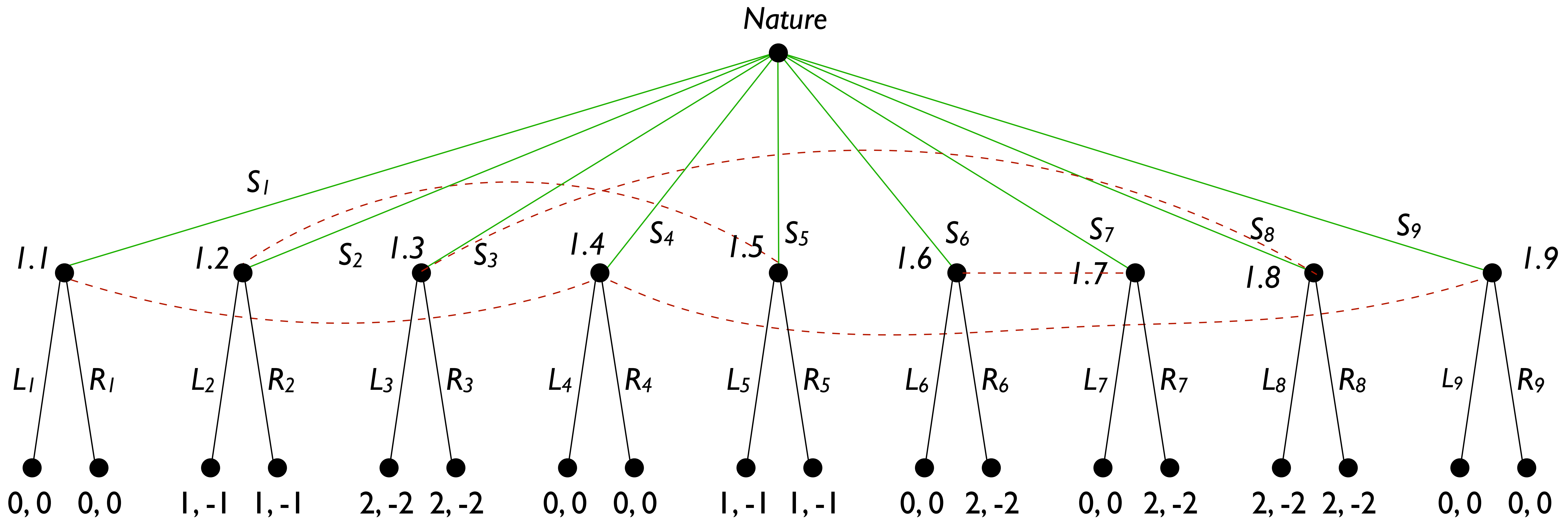
S_1, S_4, S_9

S_2, S_5

S_3, S_8

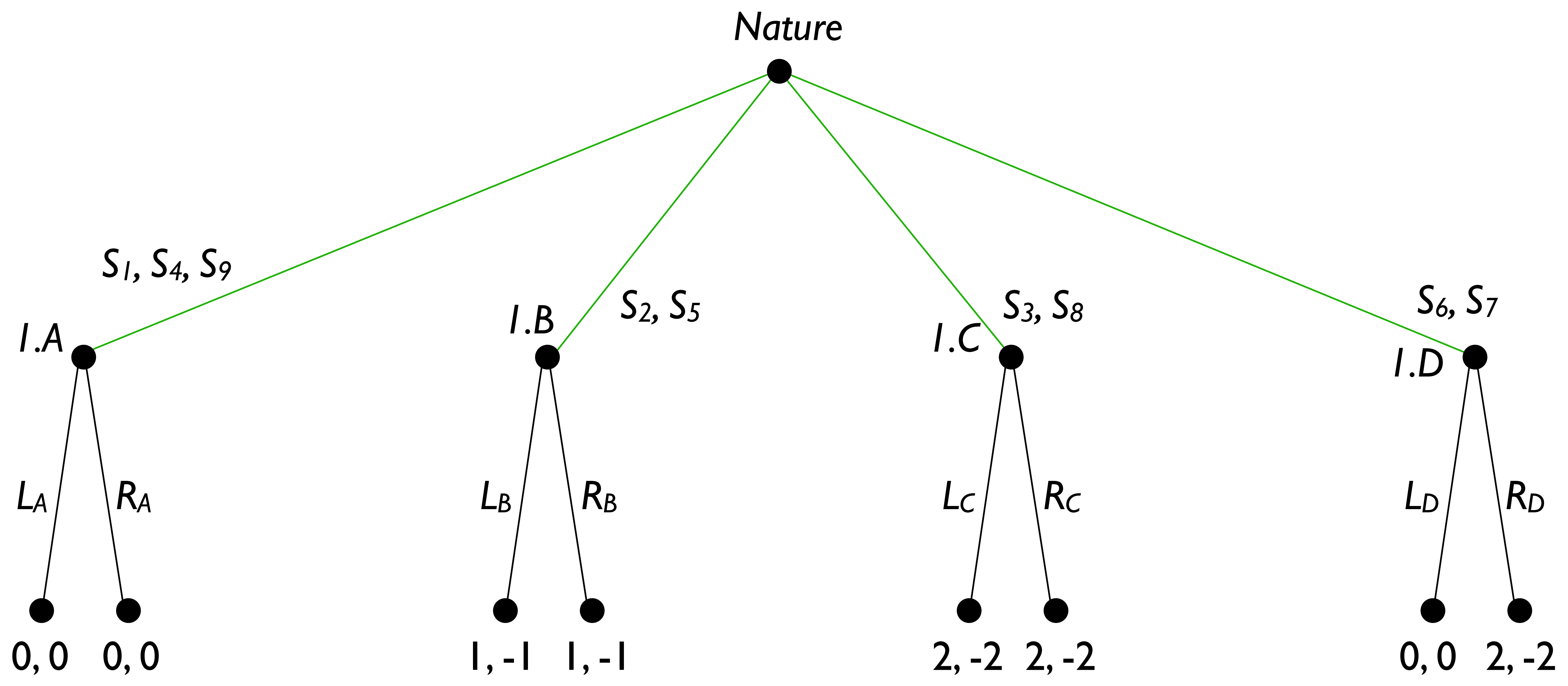
S_6, S_7

Merge two or more information sets if they lead to the same payoffs



Compressing the tree

Merge two or more information sets if they lead to the same payoffs



The original game tree and the compressed one are equivalent

State aggregation (without information loss)

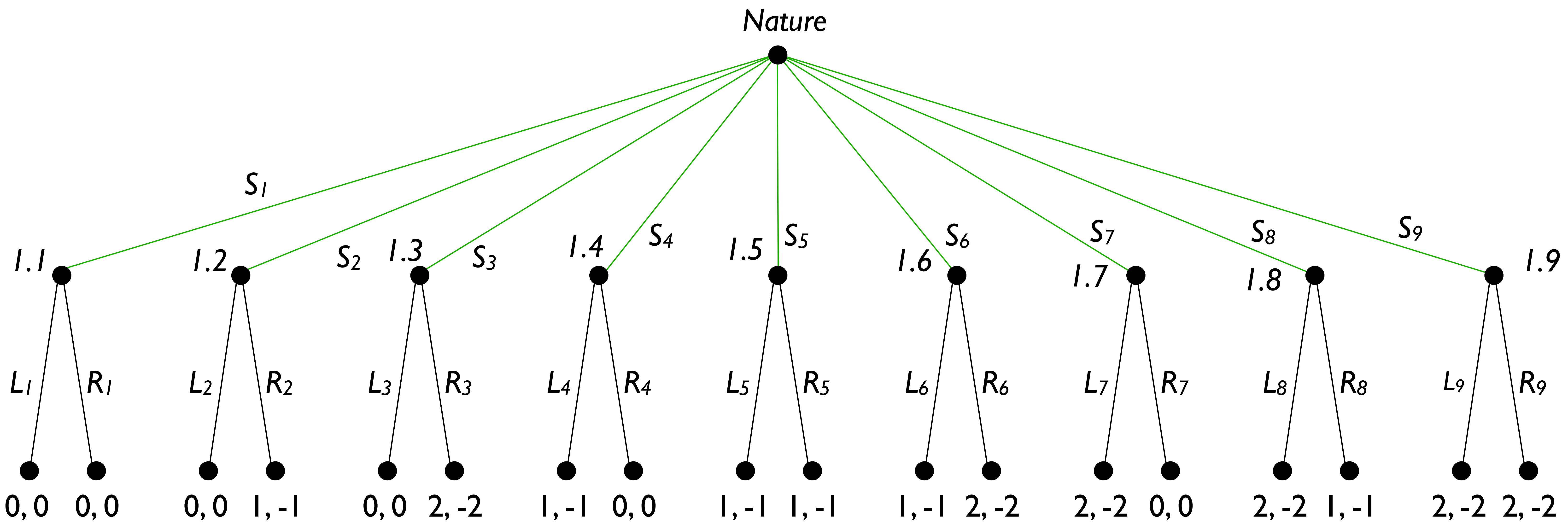
- State aggregation without loss of information allows to group few states
- The compressed game tree usually results intractable

Example in poker games

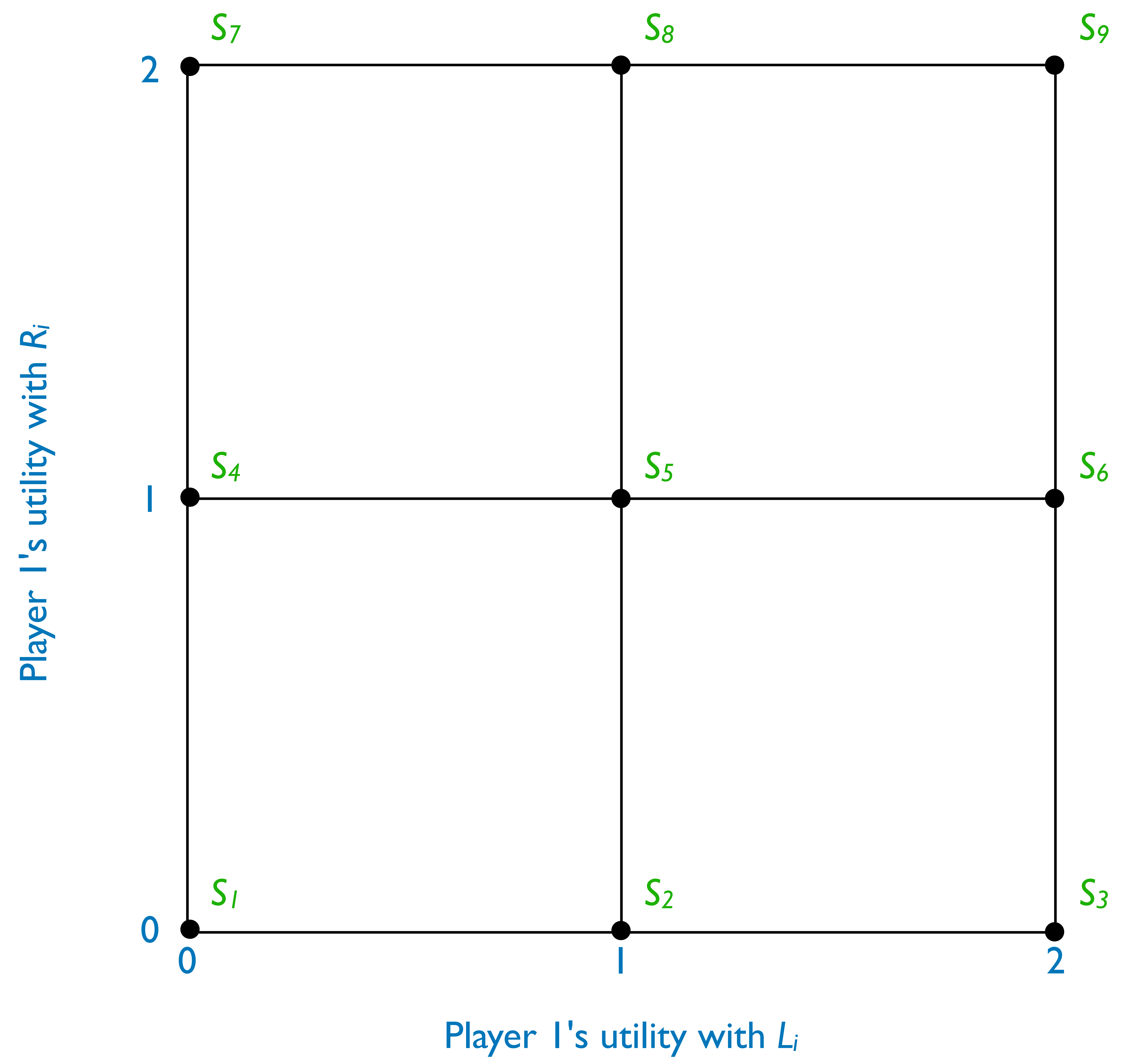
All these set of cards lead to the same payoffs

State aggregation (with information loss)

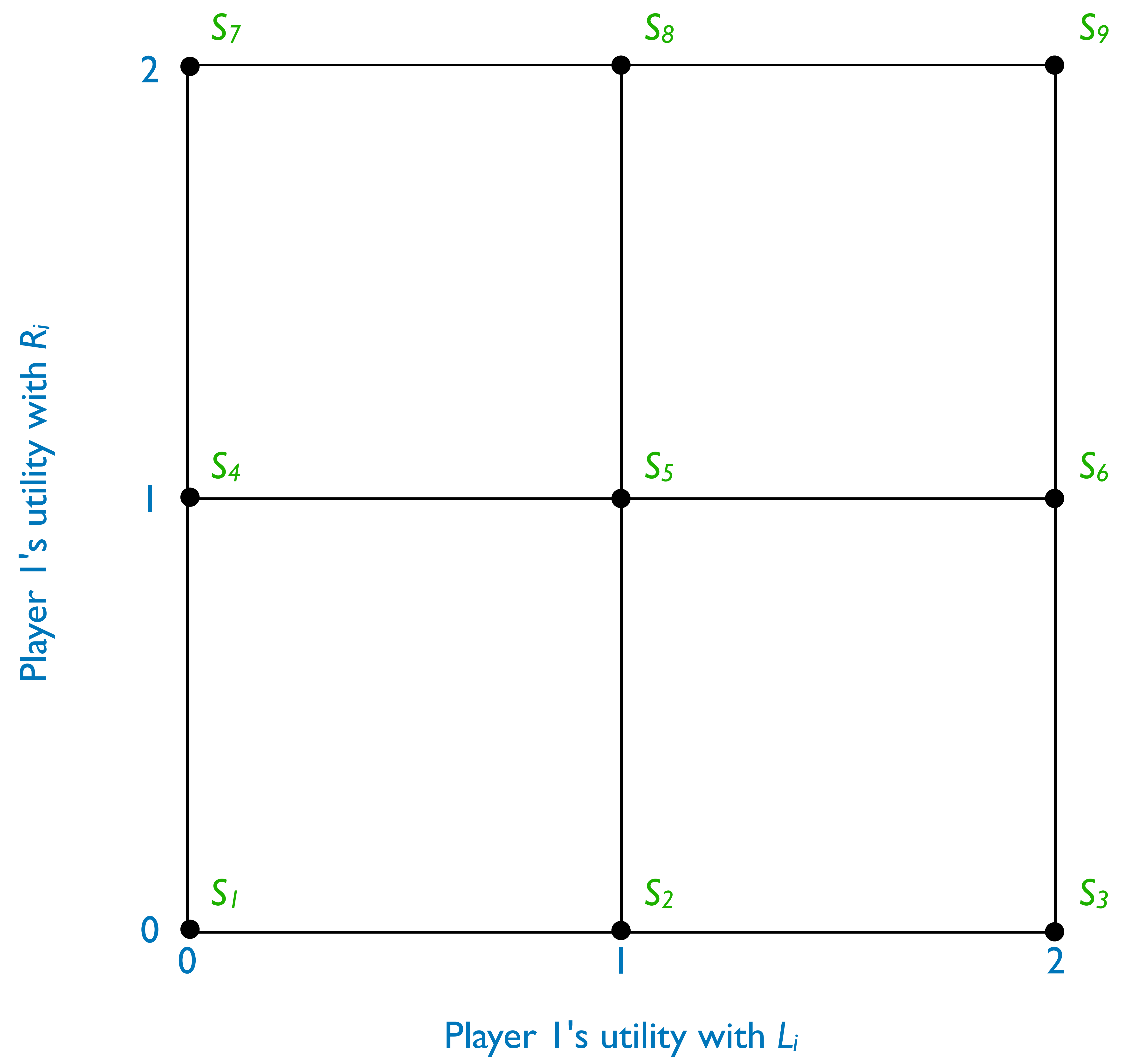
Merge two or more information sets if they lead to similar payoffs



Payoff space



Payoff space



We want to merge information sets even if they do not lead to the same payoffs

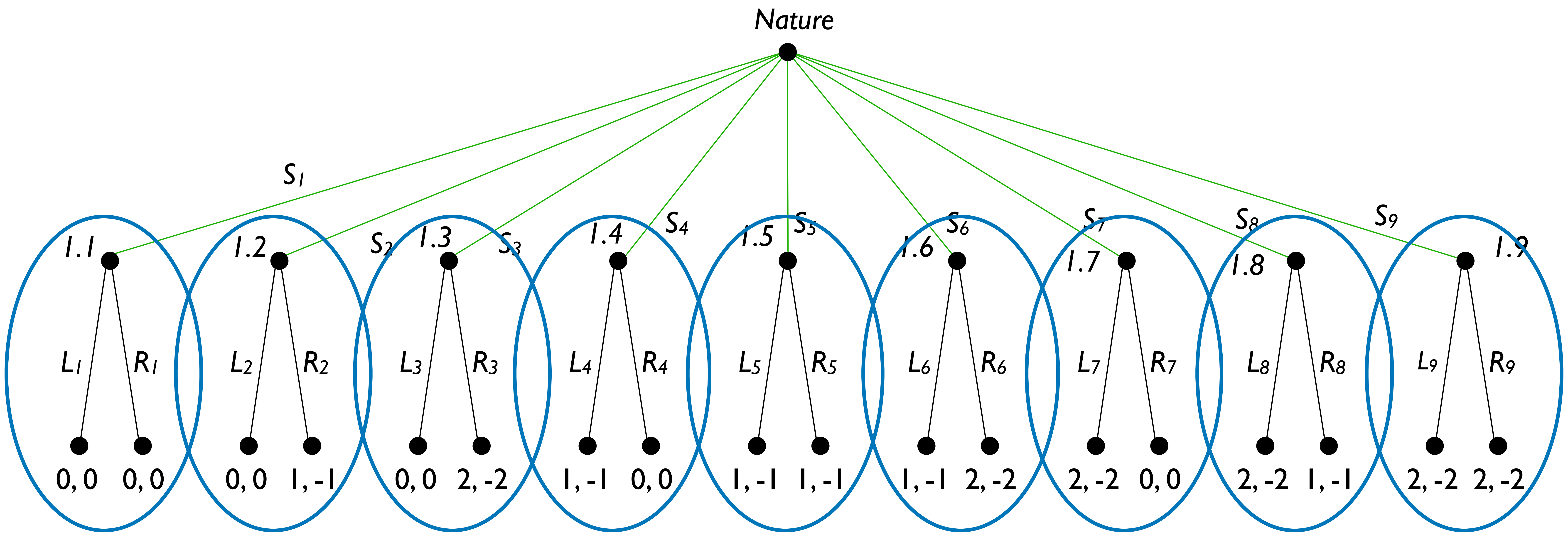
The aim is to group similar payoffs

Clustering problem

$X = \{x_1, \ldots, x_n\}$ Elements

Clustering problem

$X = \{x_1, \ldots, x_n\}$ **Elements**



Clustering problem

$X = \{x_1, \ldots, x_n\}$ Elements

$A = \{a_1, \ldots, a_m\}$ Attributes of the elements

Clustering problem

$X = \{x_1, \ldots, x_n\}$ Elements

$A = \{a_1, \ldots, a_m\}$ Attributes of the elements

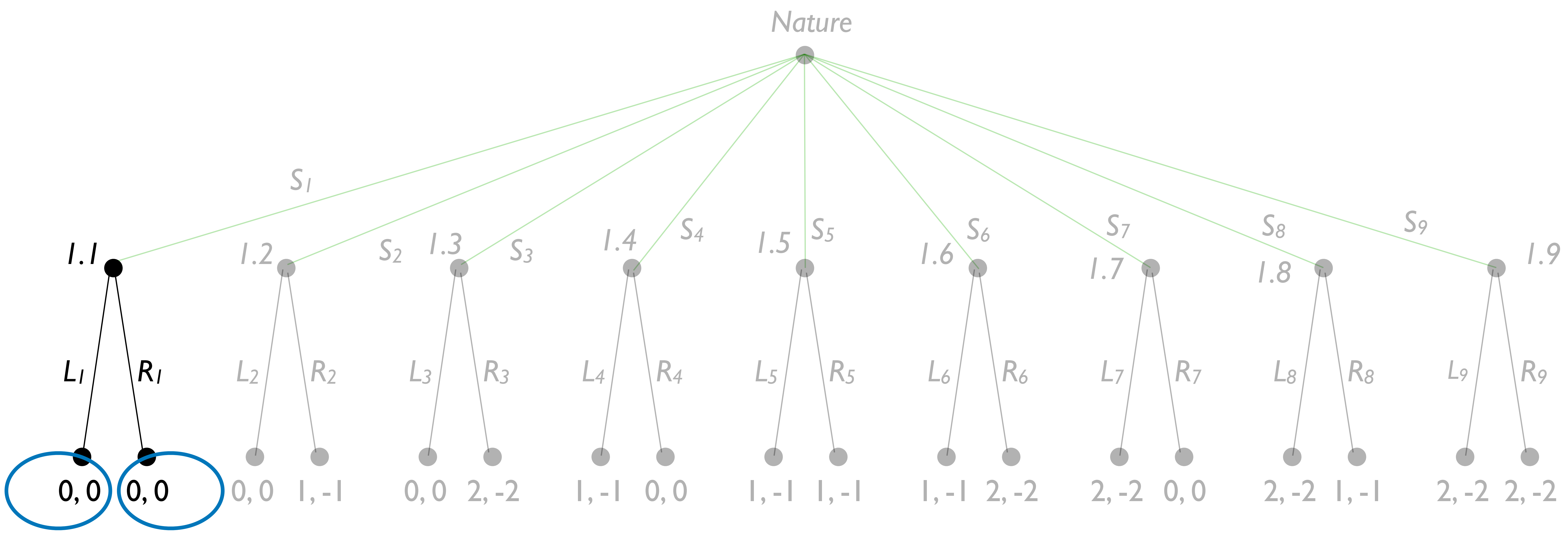
$u : X \to \mathbb{R}^m$ Attributes value of the elements

Clustering problem

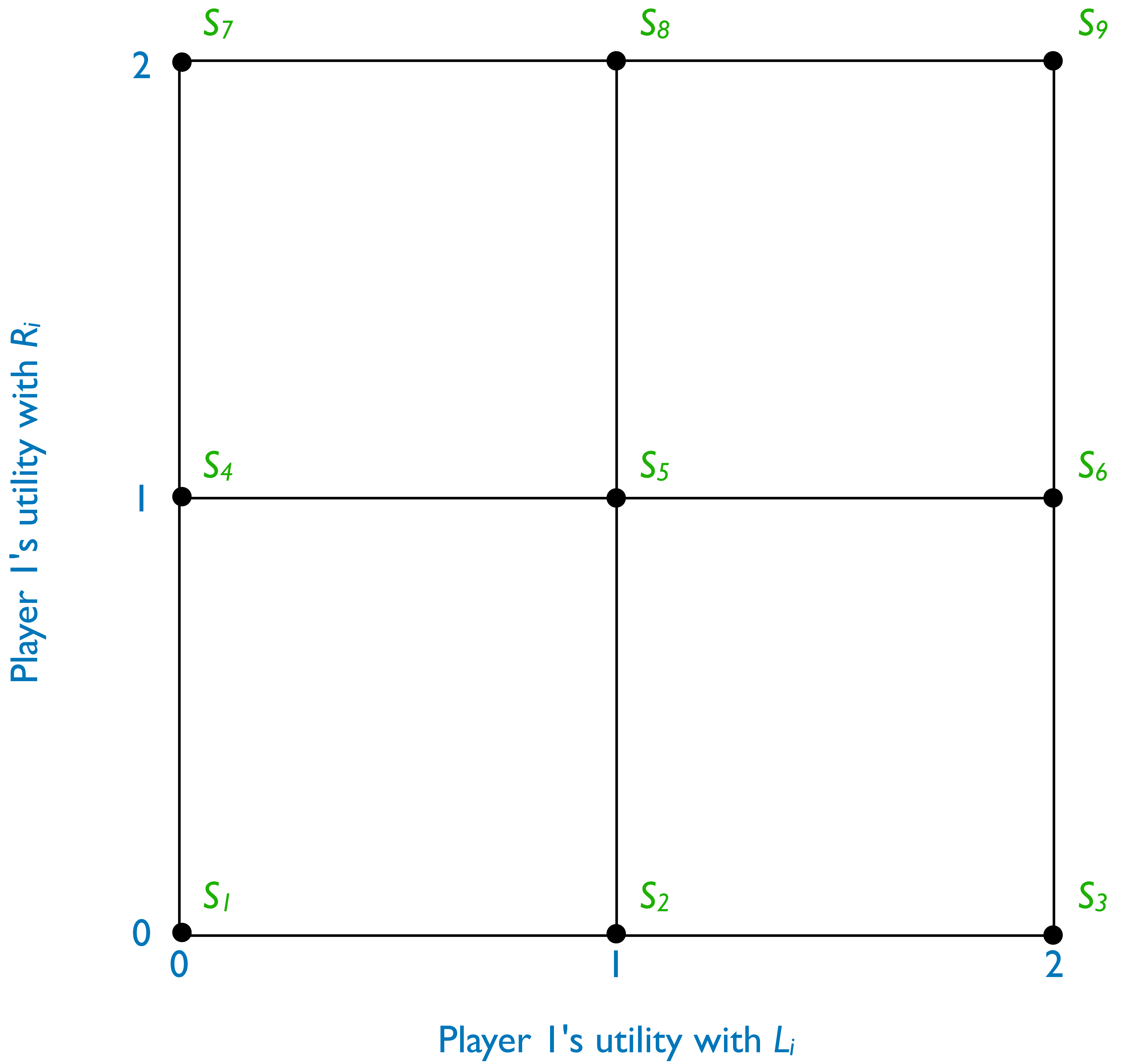
$X = \{x_1, \ldots, x_n\}$ — Elements

$A = \{a_1, \ldots, a_m\}$ — Attributes of the elements

$u : X \to \mathbb{R}^m$ — Attributes value of the elements



Attribute space (payoff space)



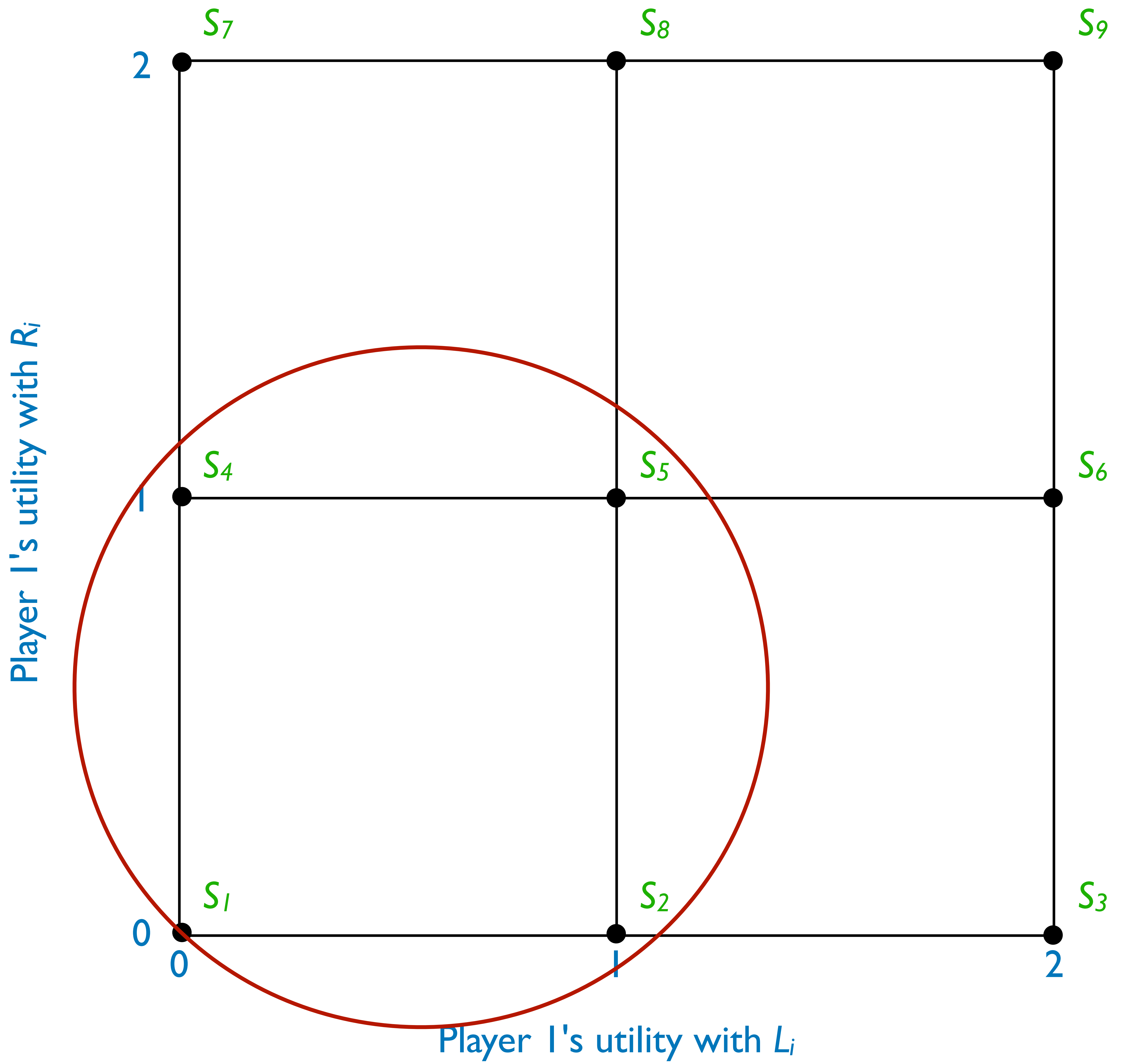
Clustering problem

$X = \{x_1, \ldots, x_n\}$	Elements
$A = \{a_1, \ldots, a_m\}$	Attributes of the elements
$u : X \to \mathbb{R}^m$	Attributes value of the elements
$X_i \subseteq X$	Cluster of elements

Clustering problem

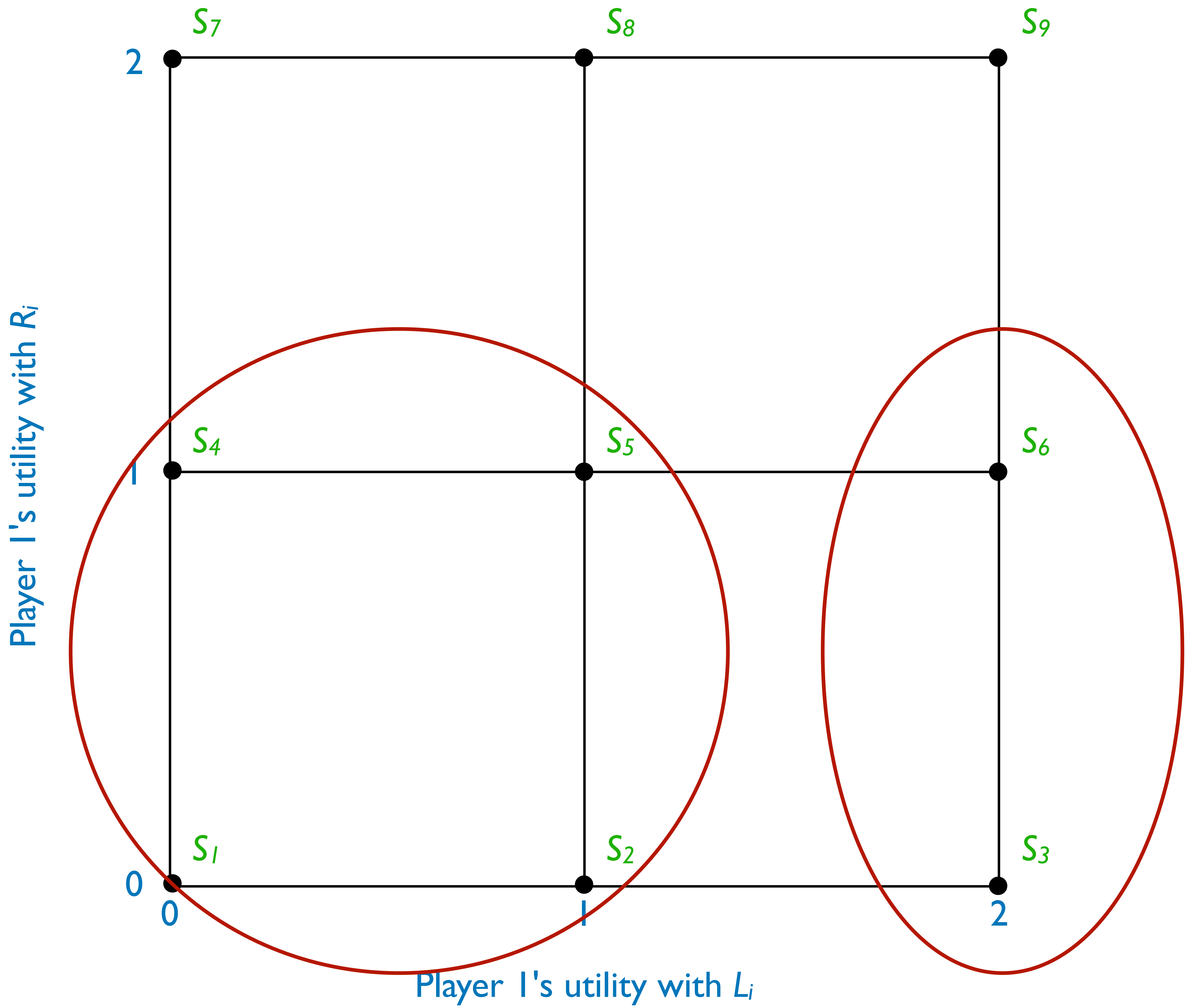
$X = \{x_1, \ldots, x_n\}$	Elements
$A = \{a_1, \ldots, a_m\}$	Attributes of the elements
$u : X \to \mathbb{R}^m$	Attributes value of the elements
$X_i \subseteq X$	Cluster of elements
$\mathcal{X} = \{X_1, X_2, \ldots, X_k\}$	Cluster partitioning

Cluster

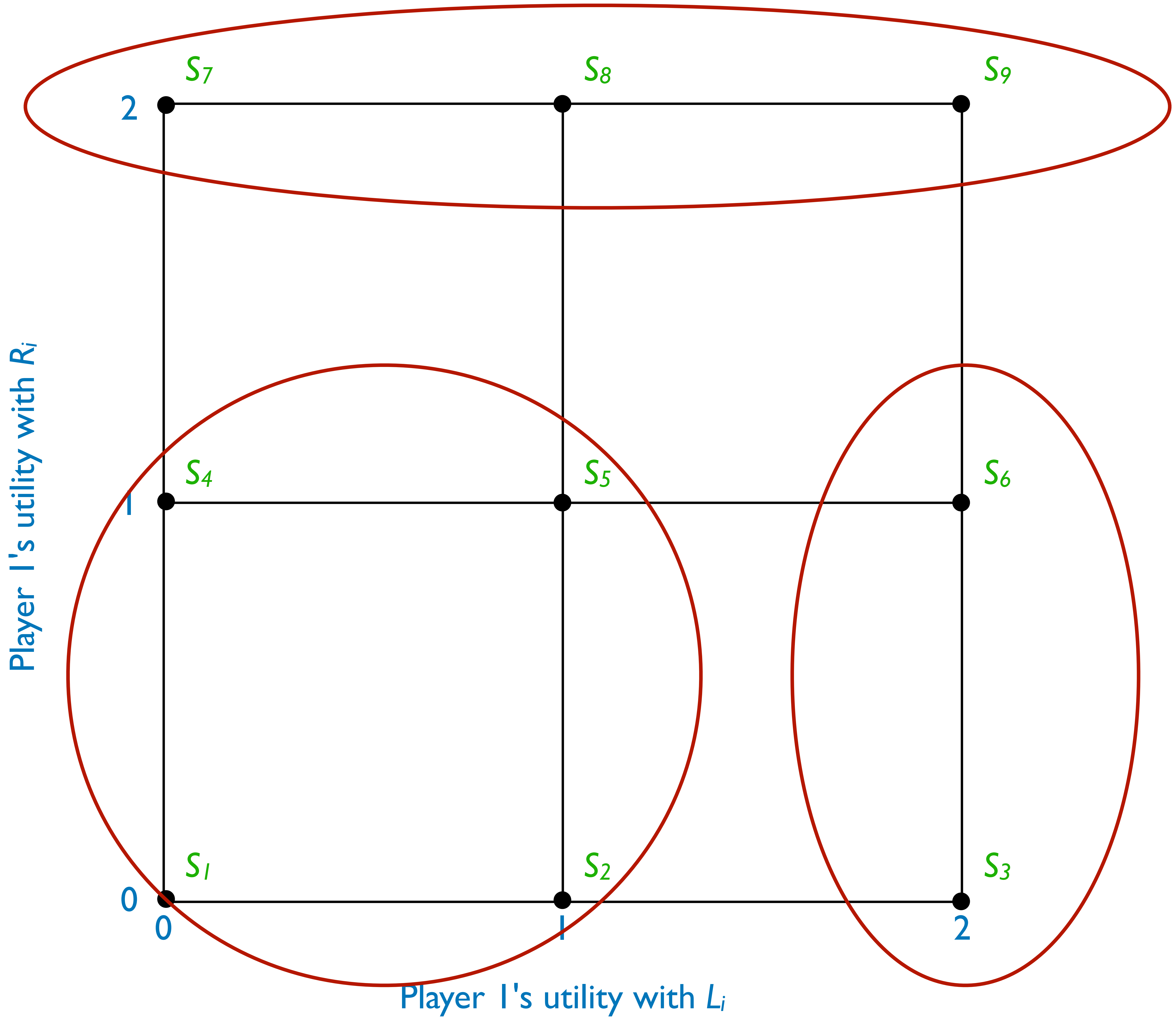


S_7
S_8
S_9
2
S_4
S_5
S_6
1
S_1
S_2
S_3
0
0
1
2
Player 1's utility with R_i
Player 1's utility with L_i

Cluster partitioning



Cluster partitioning



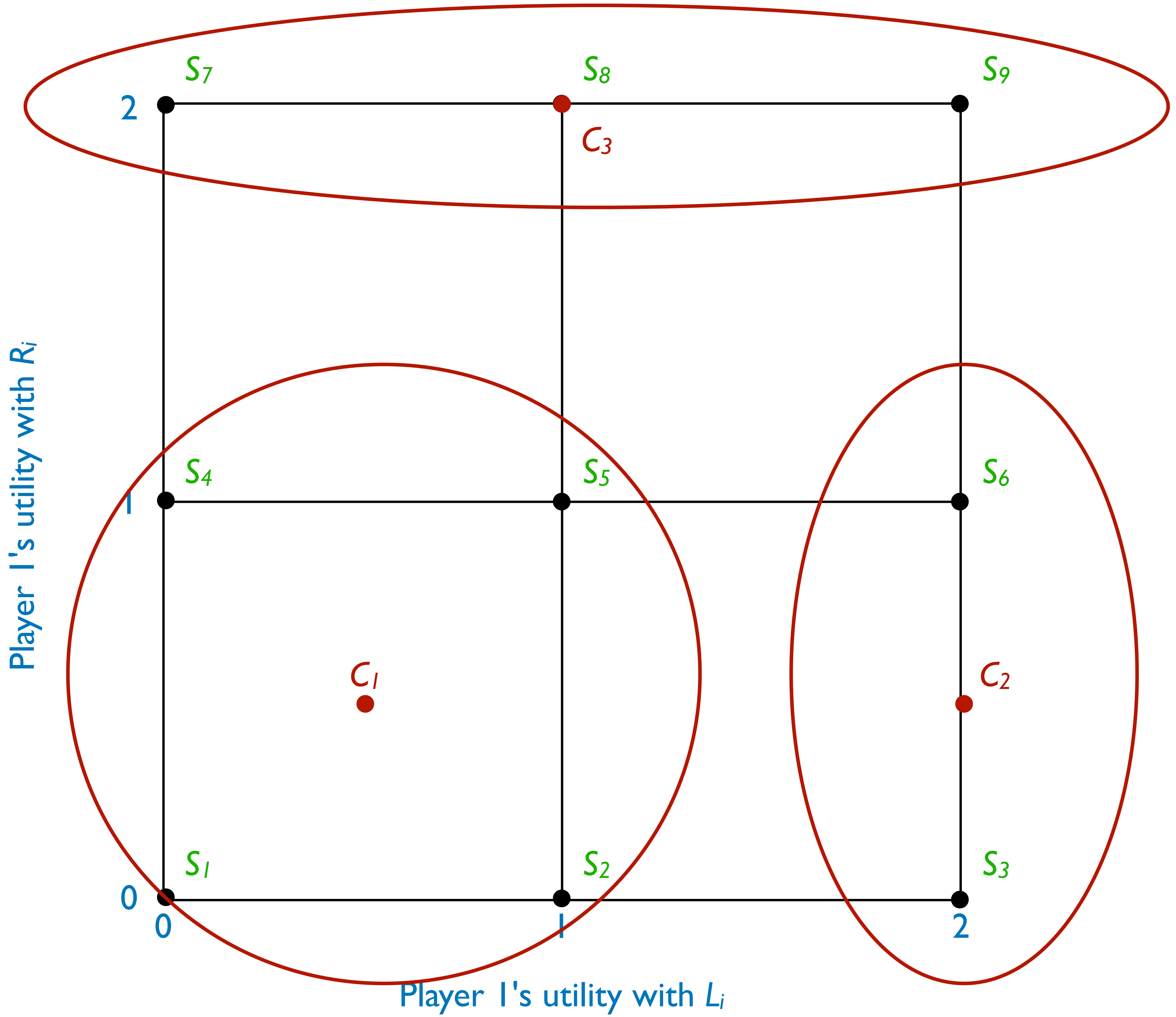
Clustering problem

$X = \{x_1, \ldots, x_n\}$	Elements
$A = \{a_1, \ldots, a_m\}$	Attributes of the elements
$u : X \to \mathbb{R}^m$	Attributes value of the elements
$X_i \subseteq X$	Cluster of elements
$\mathcal{X} = \{X_1, X_2, \ldots, X_k\}$	Cluster partitioning
C_i	Centroid of cluster *i*

Clustering problem

$X = \{x_1, \ldots, x_n\}$	Elements
$A = \{a_1, \ldots, a_m\}$	Attributes of the elements
$u : X \to \mathbb{R}^m$	Attributes value of the elements
$X_i \subseteq X$	Cluster of elements
$\mathcal{X} = \{X_1, X_2, \ldots, X_k\}$	Cluster partitioning
C_i	Centroid of cluster *i*
$u(C_i) = \frac{1}{\lvert X_i \rvert} \sum_{x \in X_i} u(x)$	Attributes value of the elements of a centroid

Centroids


S7
S8
S9
2
C3
S4
S5
S6
1
C1
C2
S1
S2
S3
0
0
1
2
Player 1's utility with Ri
Player 1's utility with Li

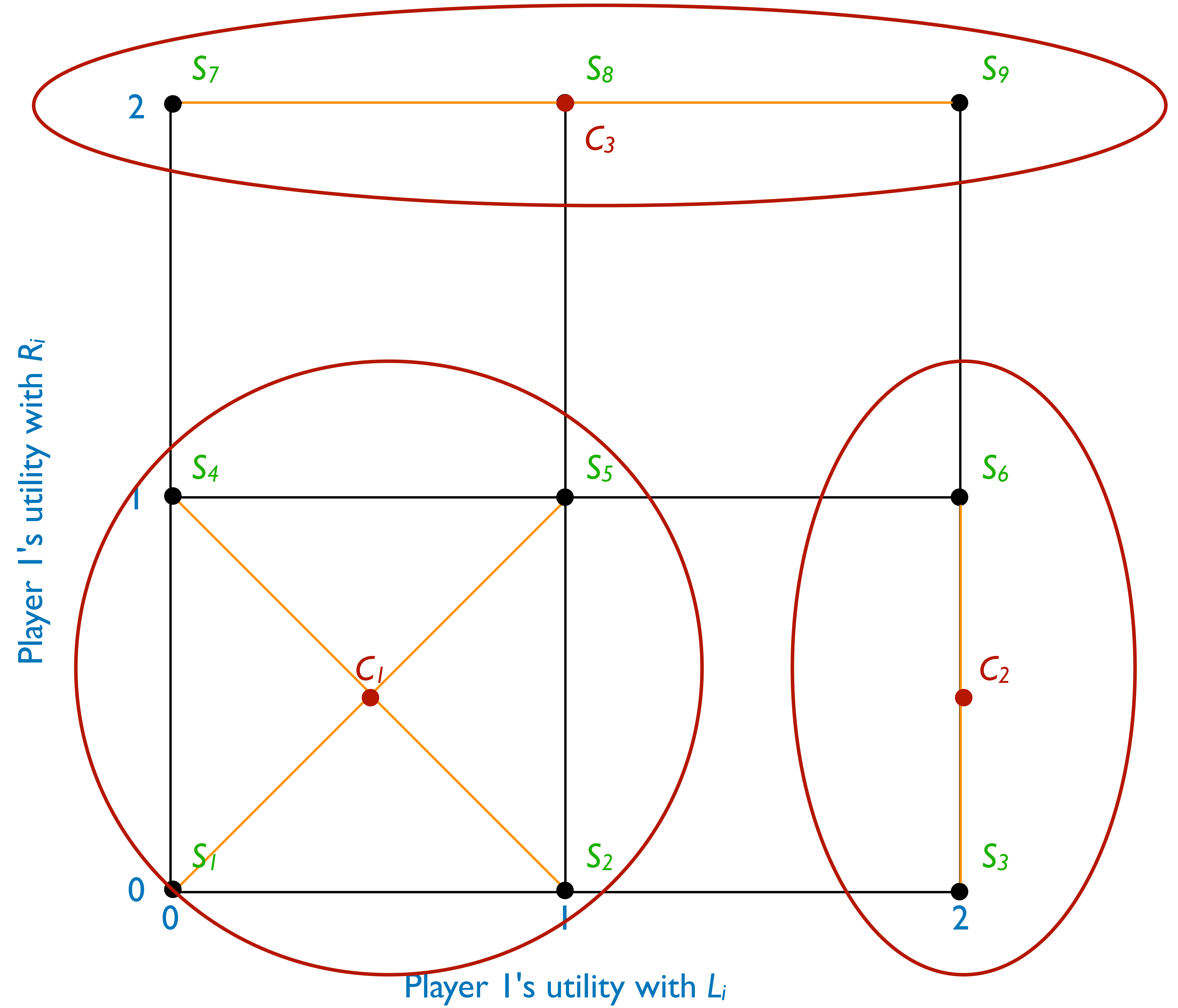
Clustering problem

$X = \{x_1, \ldots, x_n\}$	Elements
$A = \{a_1, \ldots, a_m\}$	Attributes of the elements
$u : X \to \mathbb{R}^m$	Attributes value of the elements
$X_i \subseteq X$	Cluster of elements
$\mathcal{X} = \{X_1, X_2, \ldots, X_k\}$	Cluster partitioning
C_i	Centroid of cluster *i*
$u(C_i) = \frac{1}{\|X_i\|} \sum_{x \in X_i} u(x)$	Attributes value of the elements of a centroid

Given *k*, the goal is finding the *k* centroids such that

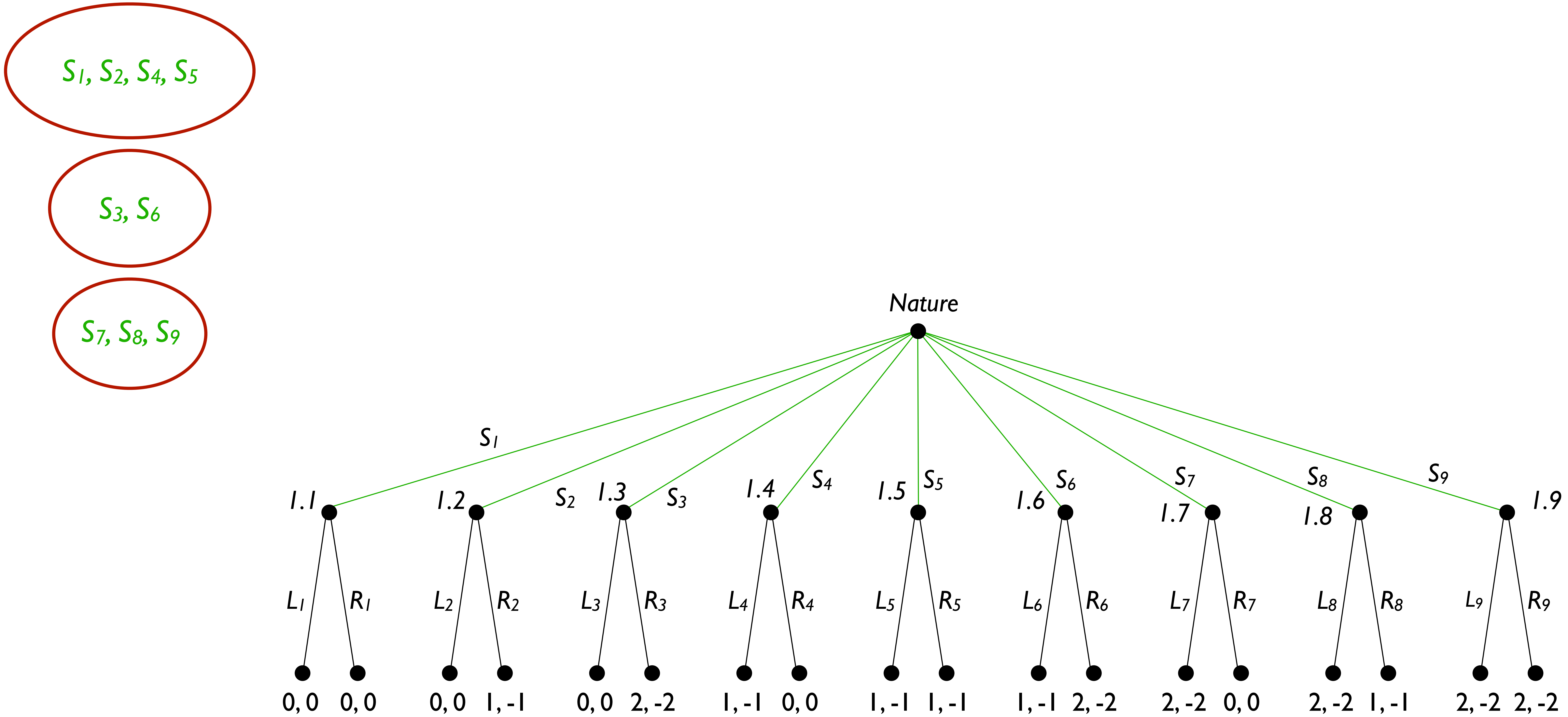
$$\arg \min_{C_1, \ldots, C_k} \sum_{i=1}^{k} \sum_{x \in X_i} ||u(x) - u(C_i)||^2$$

Centroids



S_7
S_8
S_9
2
C_3
Player 1's utility with R_i
S_4
S_5
S_6
1
C_1
C_2
S_1
S_2
S_3
0
0
1
2
Player 1's utility with L_i

State aggregation (with information loss)

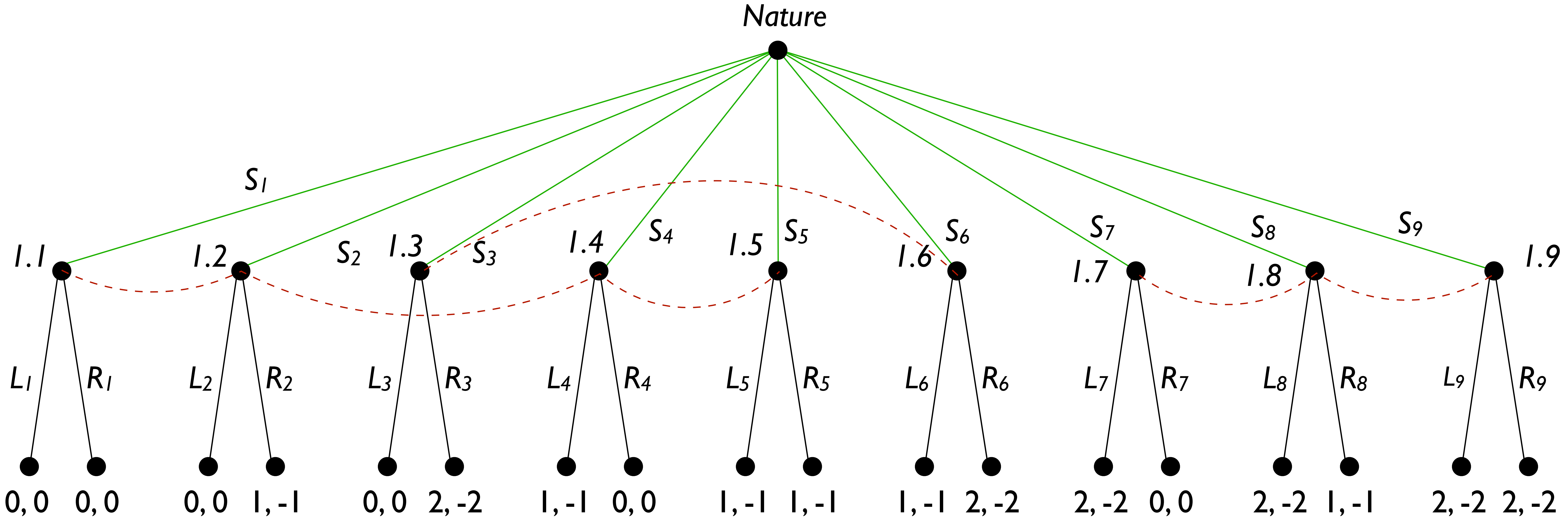


State aggregation (with information loss)

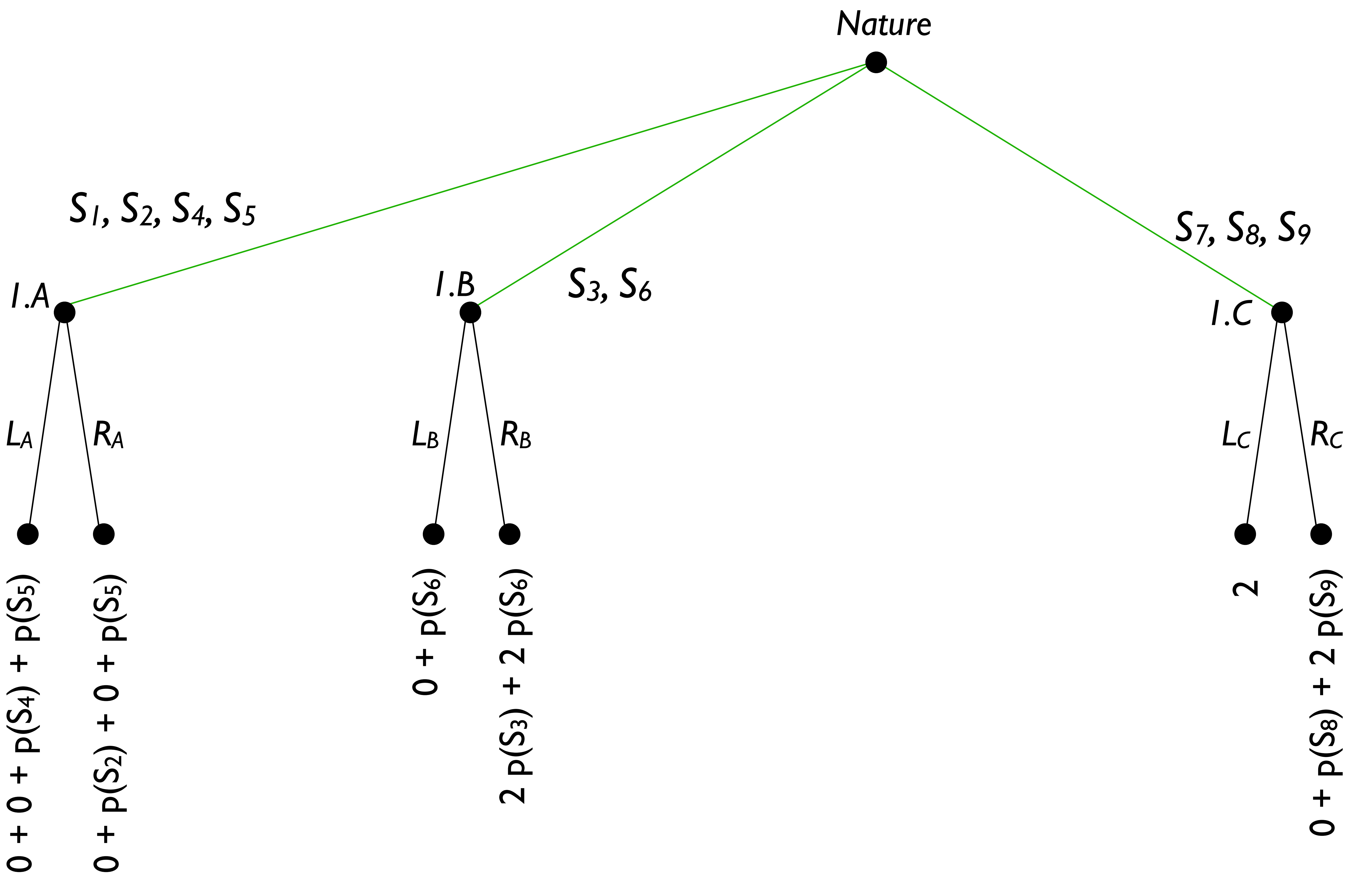
S_1, S_2, S_4, S_5

S_3, S_6

S_7, S_8, S_9



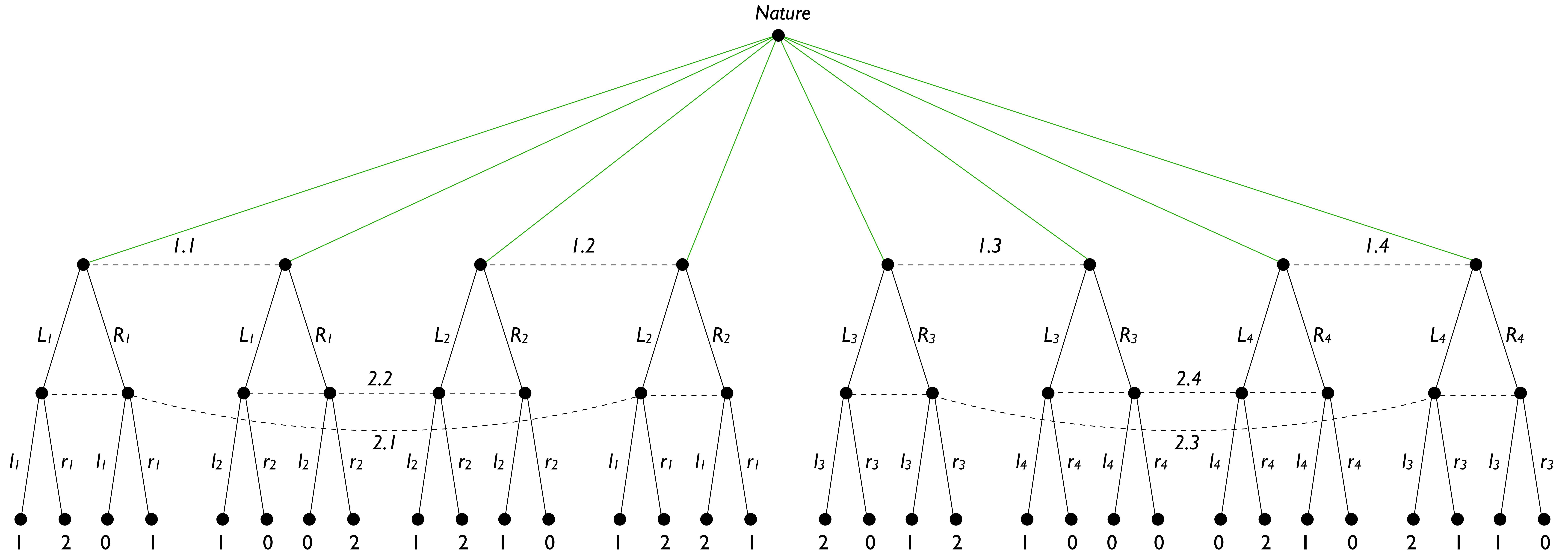
State aggregation (with information loss)



k-means algorithm

- *k*-means algorithm is one of the most used clustering algorithms
- It returns a loca minimum (for this reason, random restarts are used)
- Very fast in practice, but exponential time in the worst case

Example with 2 players



Aggregating player 1's states

1.1	1	2	0	1	1	0	0	2
1.2	1	2	1	0	1	2	2	1
1.3	2	0	1	2	1	0	0	0
1.4	0	2	1	0	2	1	1	0

Produce only 2 clusters
(the payoff space has 8 dimensions)

Aggregating player 2's states

2.1	1	2	0	1	1	2	2	1
2.2	1	0	0	2	1	2	1	0
2.3	2	0	1	2	2	1	1	0
2.4	1	0	0	0	0	2	1	0

Produce only 2 clusters
(the payoff space has 8 dimensions)

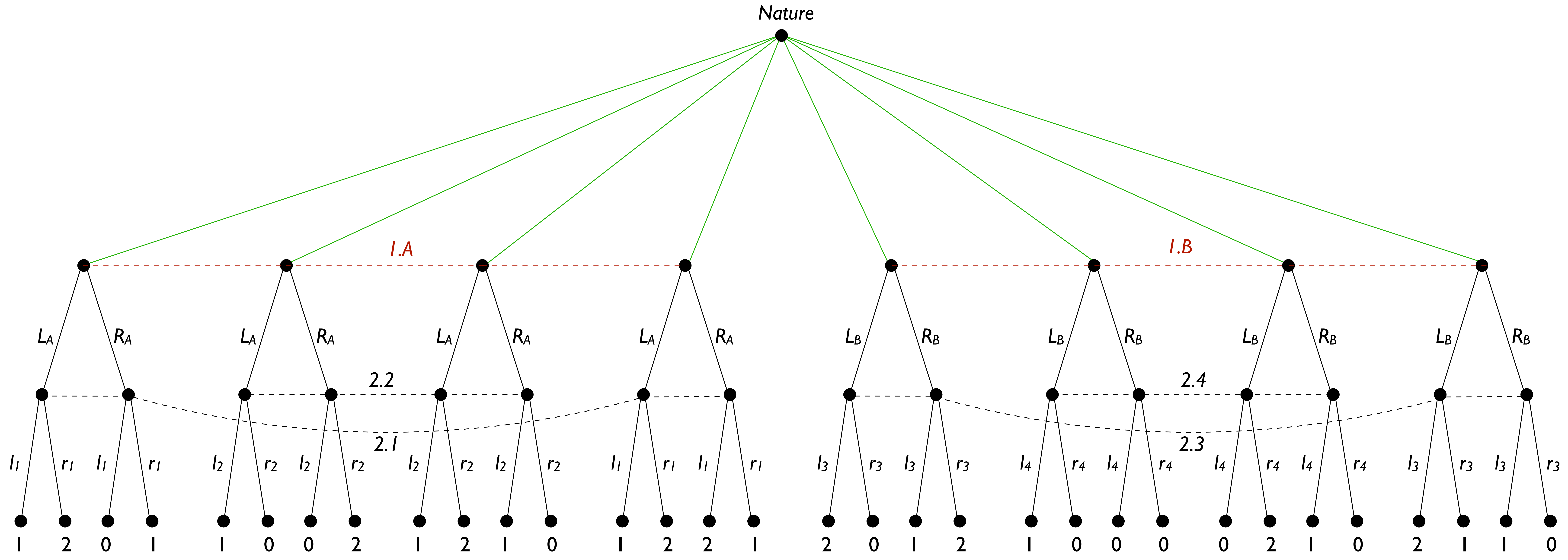
Complete game

	0	l_1	r_1	l_2	r_2	l_3	r_3	l_4	r_4
0									
L_1									
R_1									
L_2									
R_2									
L_3									
R_3									
L_4									
R_4									

Complete game

	0	l_1	r_1	l_2	r_2	l_3	r_3	l_4	r_4
0									
L_1		1/8	2/8	1/8	0				
R_1		0	1/8	0	2/8				
L_2		1/8	2/8	1/8	2/8				
R_2		2/8	1/8	1/8	0				
L_3						2/8	0	1/8	0
R_3						1/8	2/8	0	0
L_4						2/8	1/8	0	2/8
R_4						1/8	0	1/8	0

Partially abstracted game



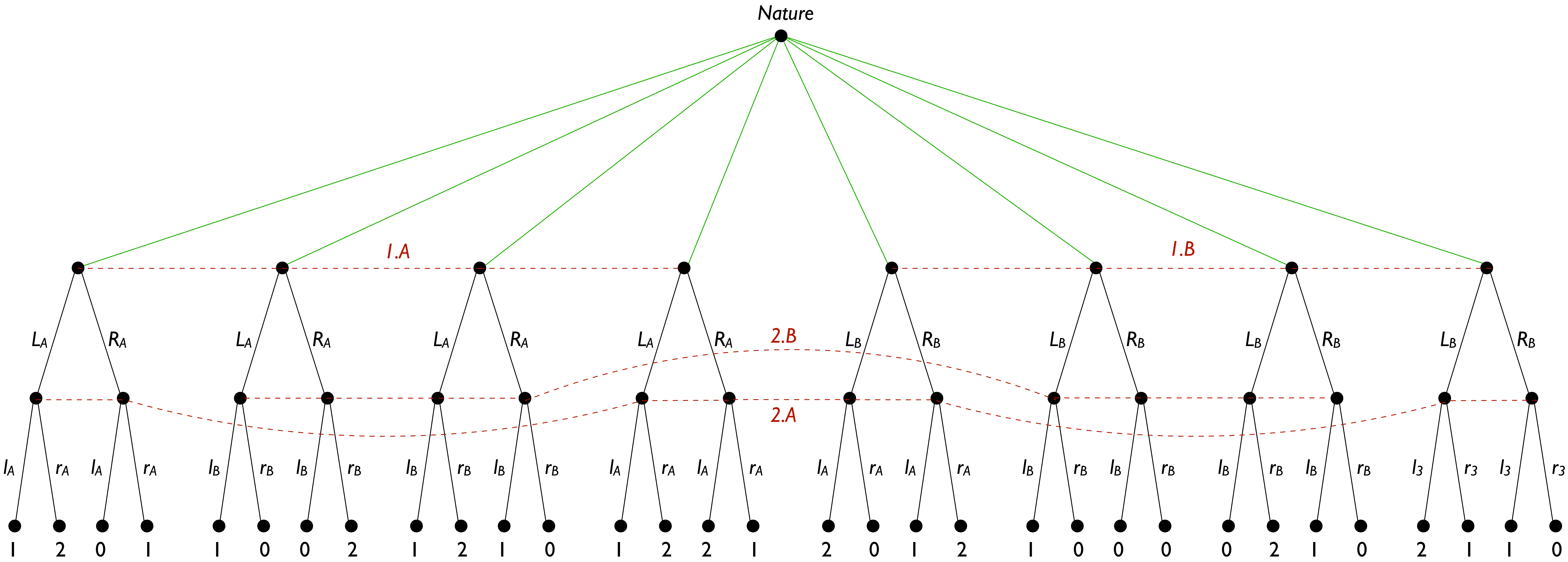
Partially abstracted game

	0	l_1	r_1	l_2	r_2	l_3	r_3	l_4	r_4
0									
L_A									
R_A									
L_B									
R_B									

Partially abstracted game

	0	l_1	r_1	l_2	r_2	l_3	r_3	l_4	r_4
0									
L_A		2/8	4/8	2/8	2/8				
R_A		2/8	2/8	1/8	2/8				
L_B						4/8	1/8	1/8	2/8
R_B						2/8	2/8	1/8	0

Completely abstracted game



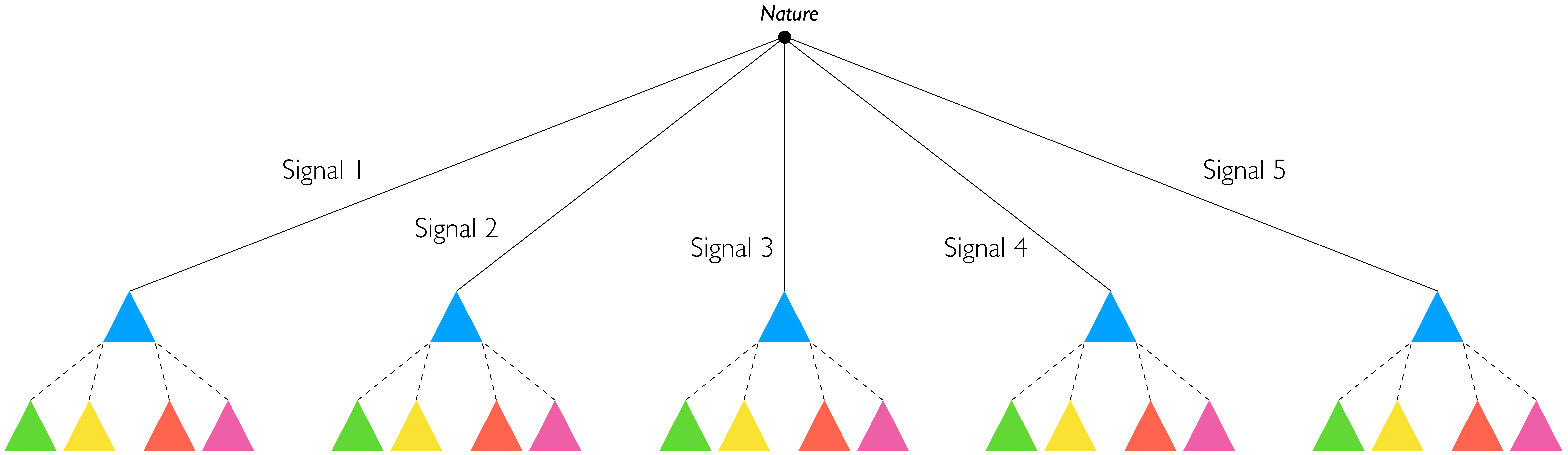
Completely abstracted game

	0	l_A	r_A	l_B	r_B
0					
L_A					
R_A					
L_B					
R_B					

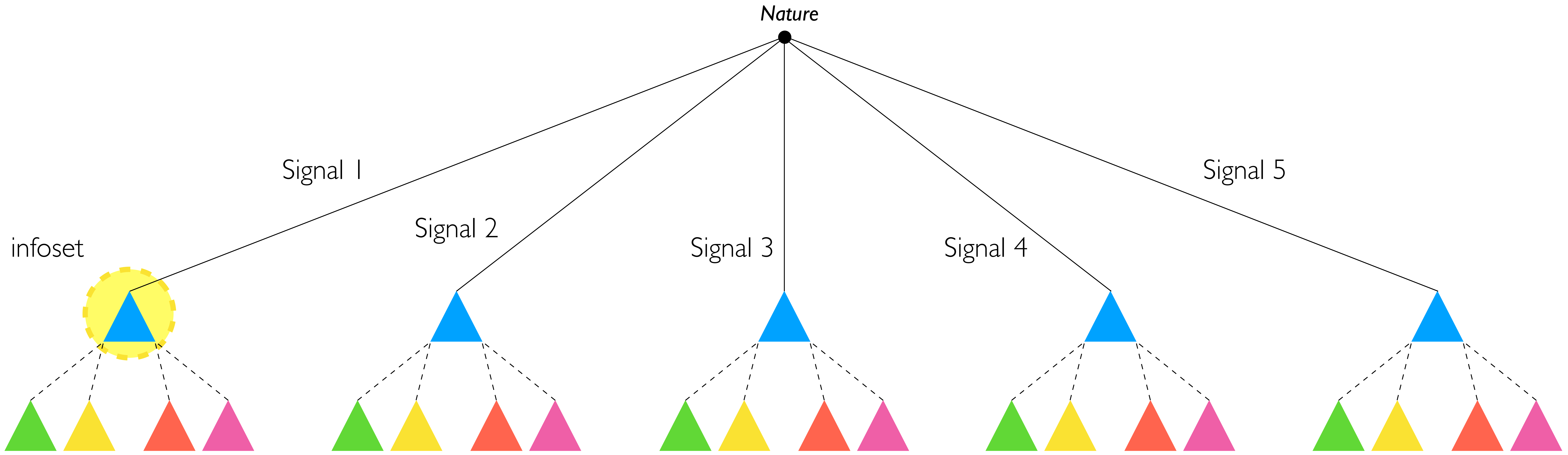
Completely abstracted game

	0	l_A	r_A	l_B	r_B
0					
L_A		2/8	4/8	2/8	2/8
R_A		2/8	2/8	1/8	2/8
L_B		4/8	1/8	1/8	2/8
R_B		2/8	2/8	1/8	0

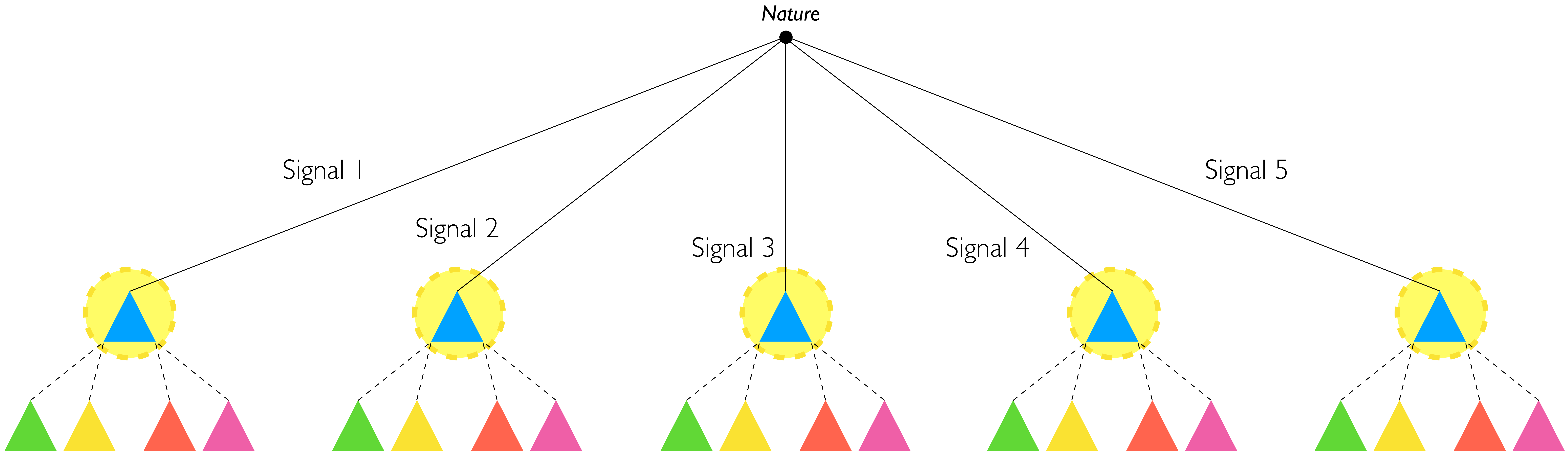
Games where players traverse multiple infosets



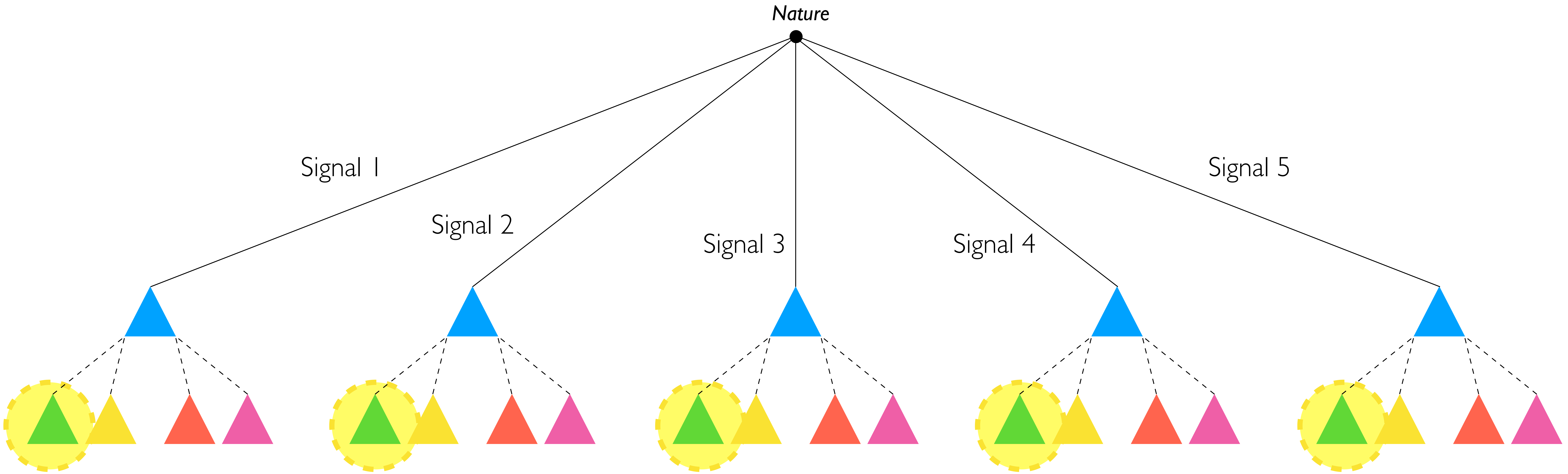
Games where players traverse multiple infosets



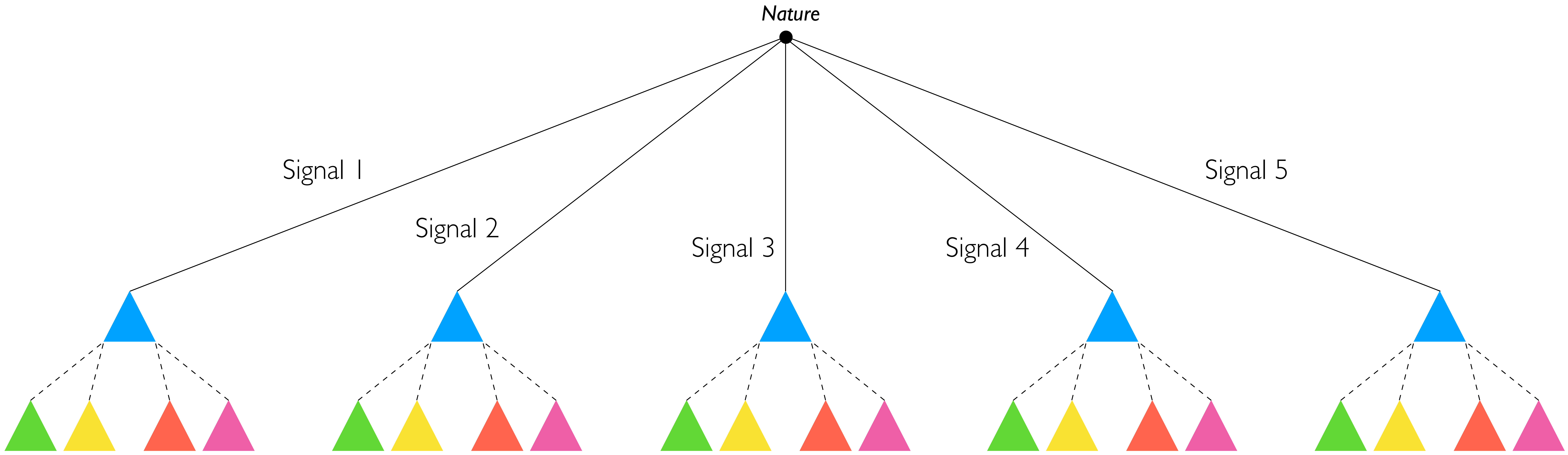
Games where players traverse multiple infosets



Games where players traverse multiple infosets

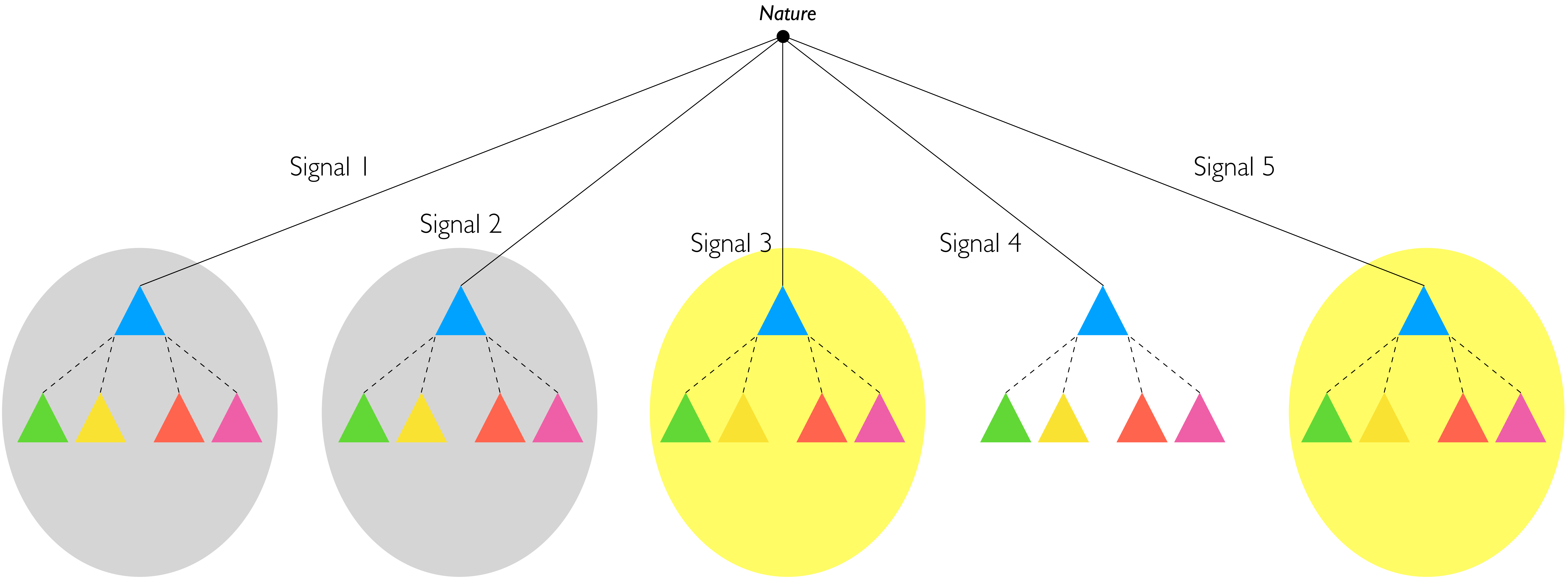


How to construct an abstraction?

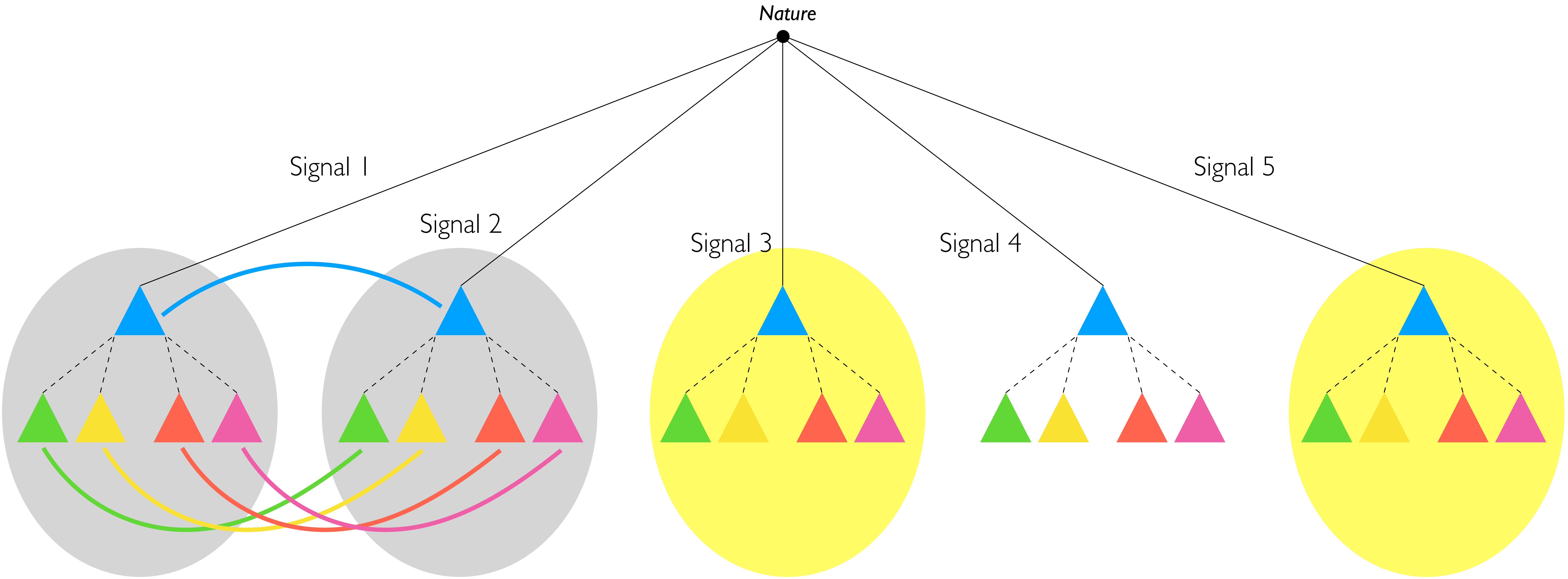


Some possibilities

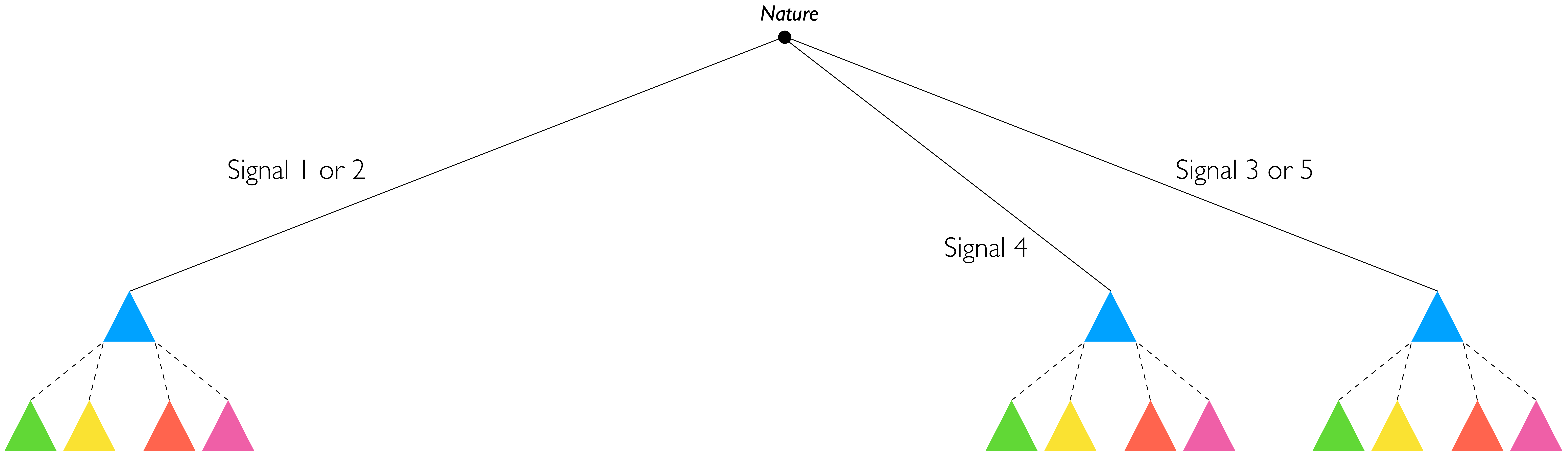
How to construct an abstraction?



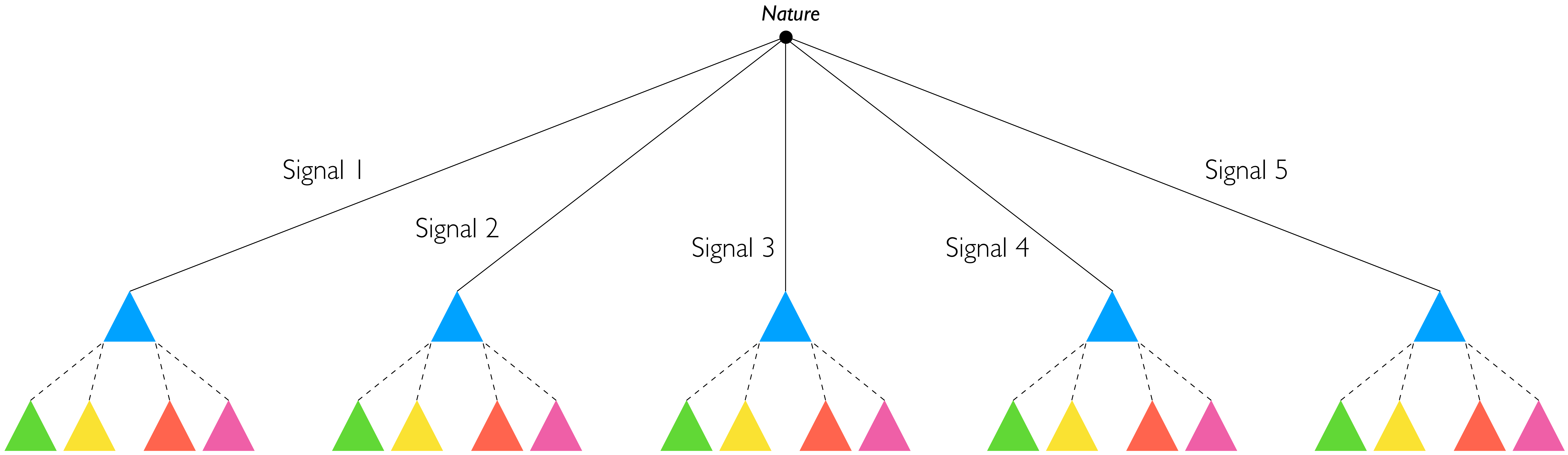
How to construct an abstraction?



How to construct an abstraction?

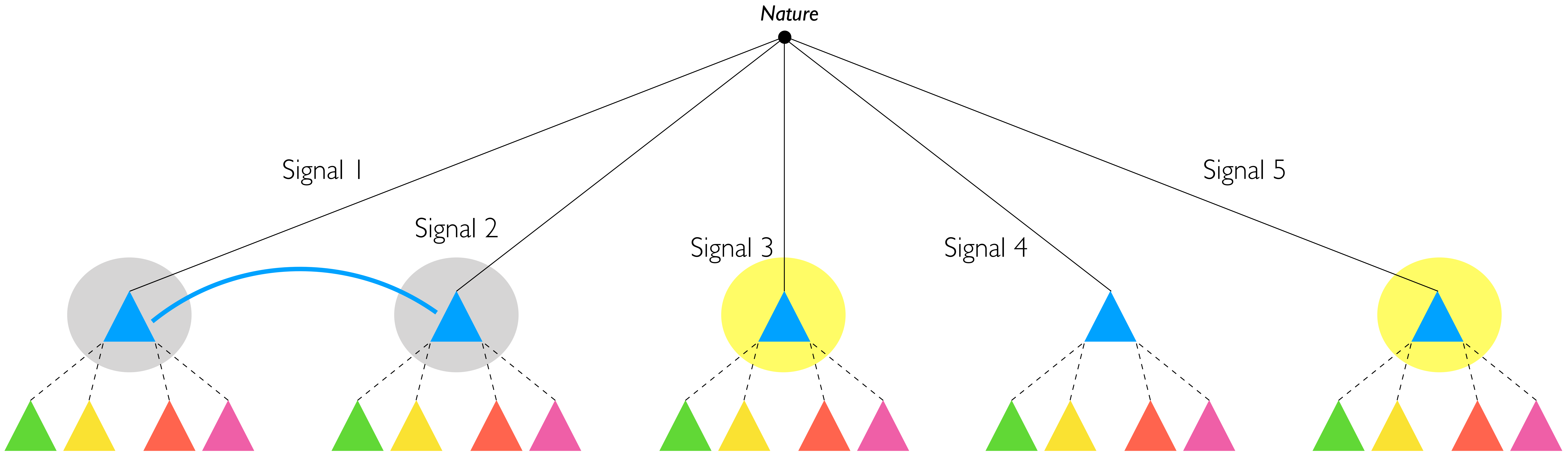


How to construct an abstraction?



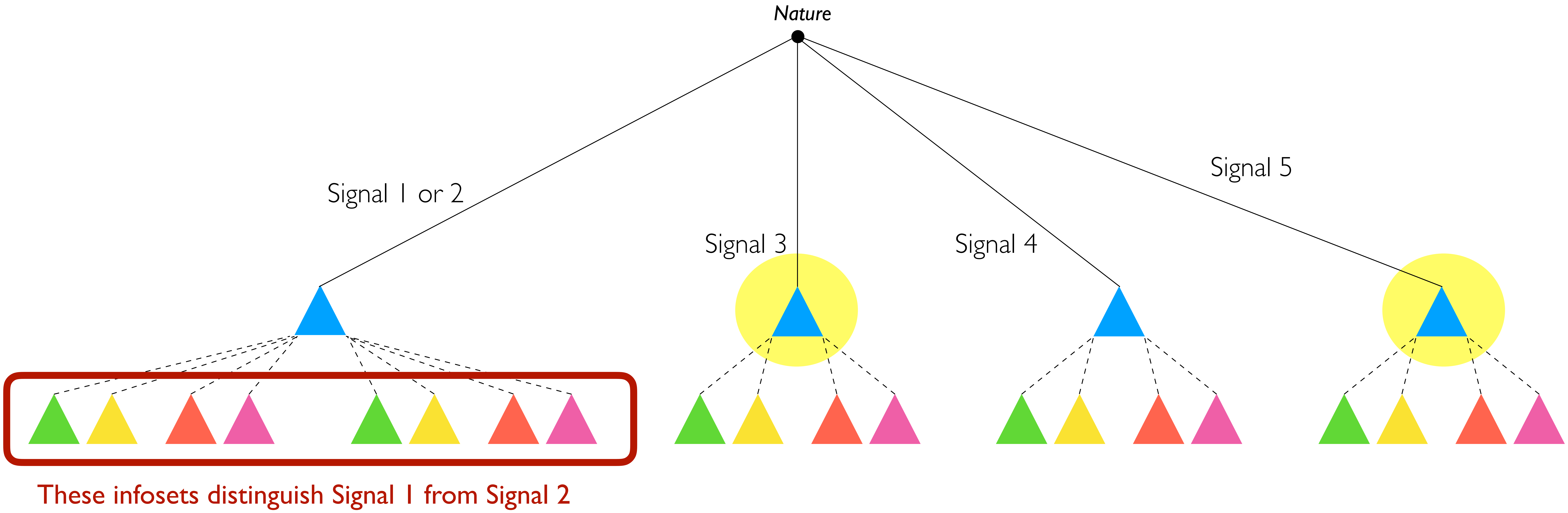
Some possibilities

How to construct an abstraction?

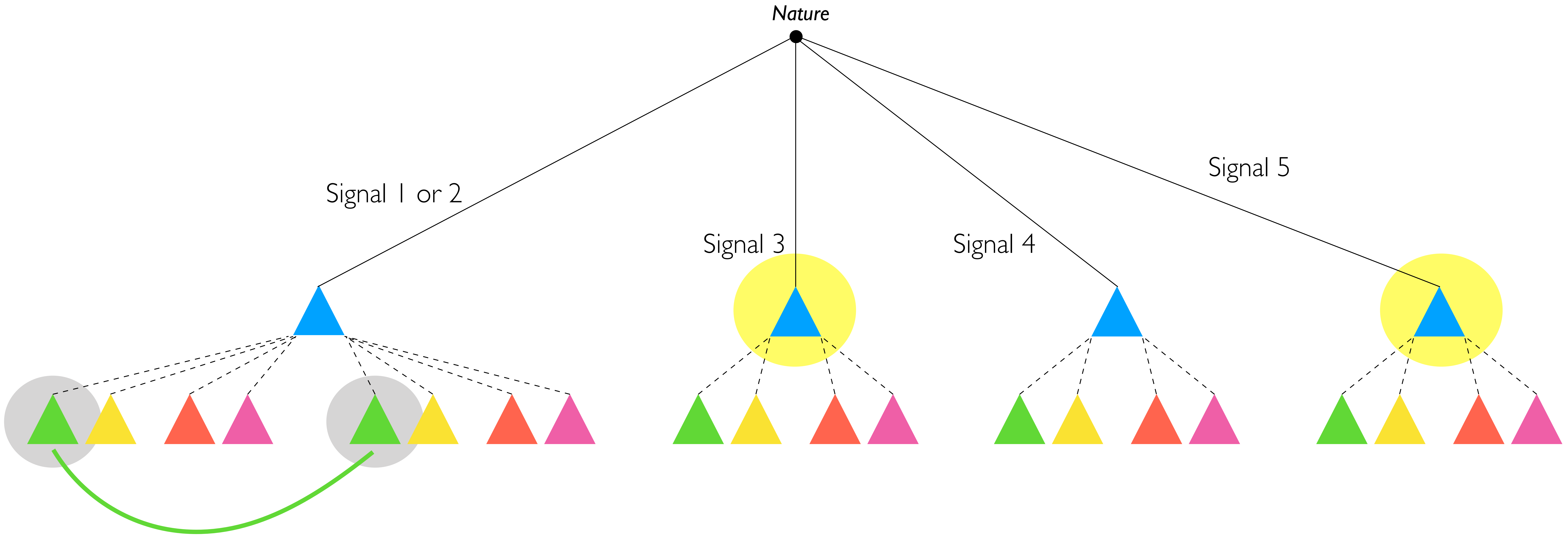


In the construction, all the terminal outcomes achievable from the merged infests must be considered

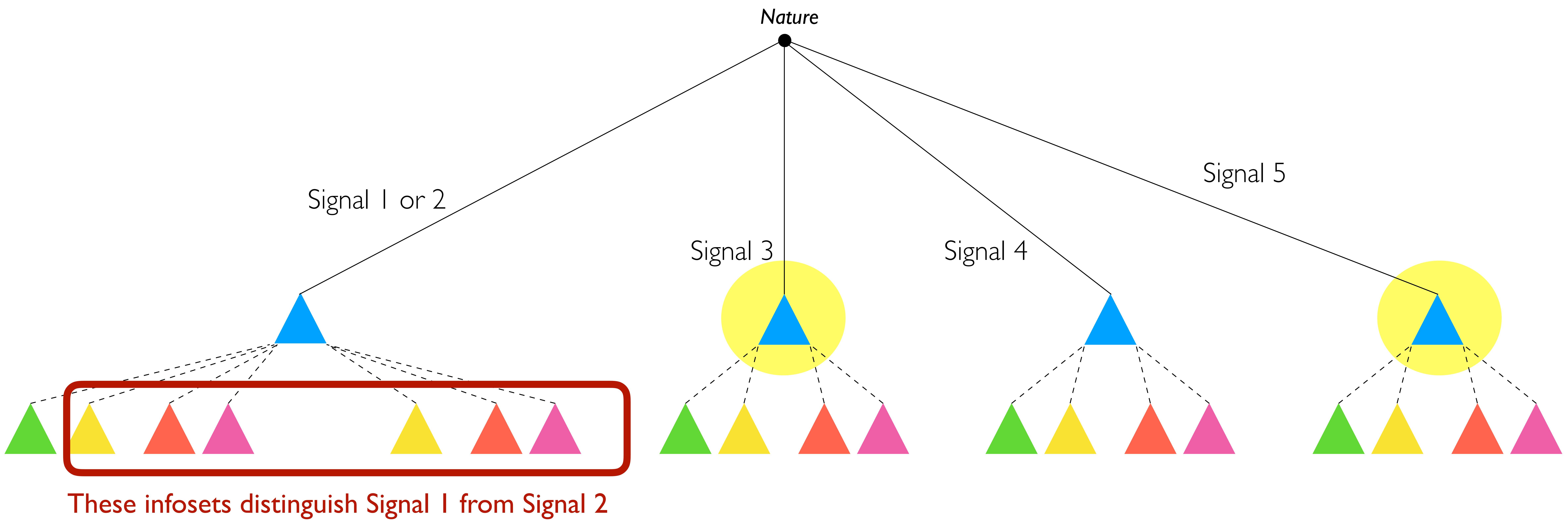
How to construct an abstraction?



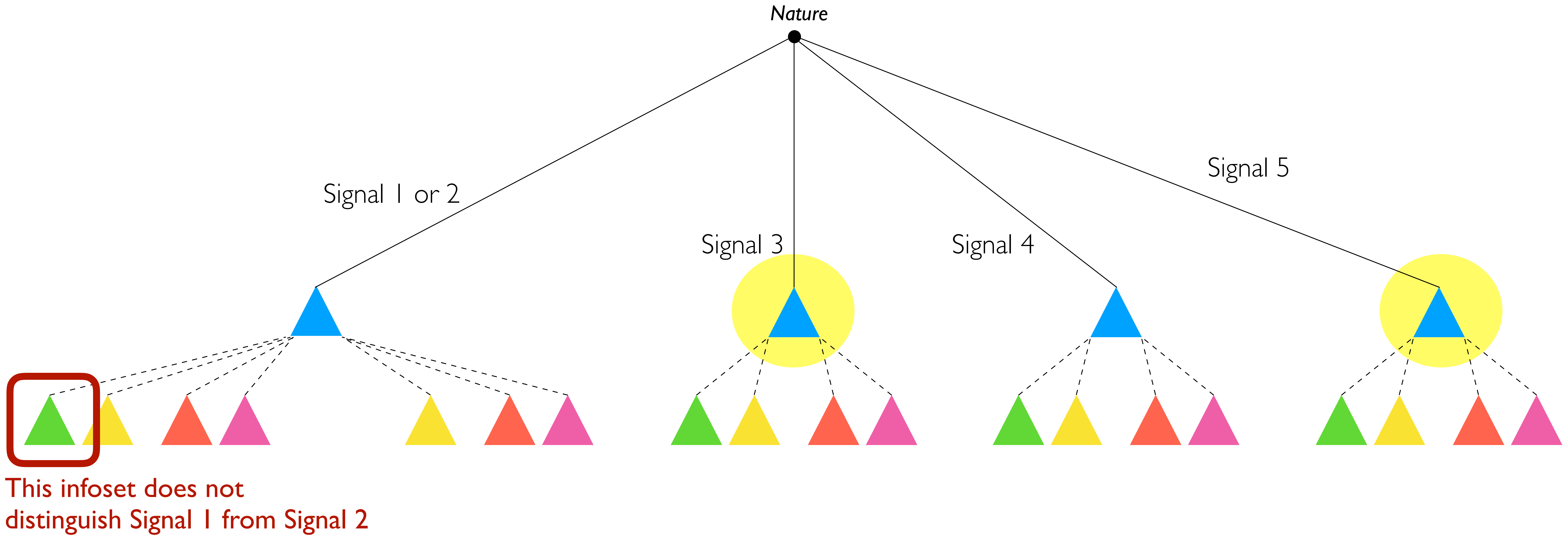
How to construct an abstraction?



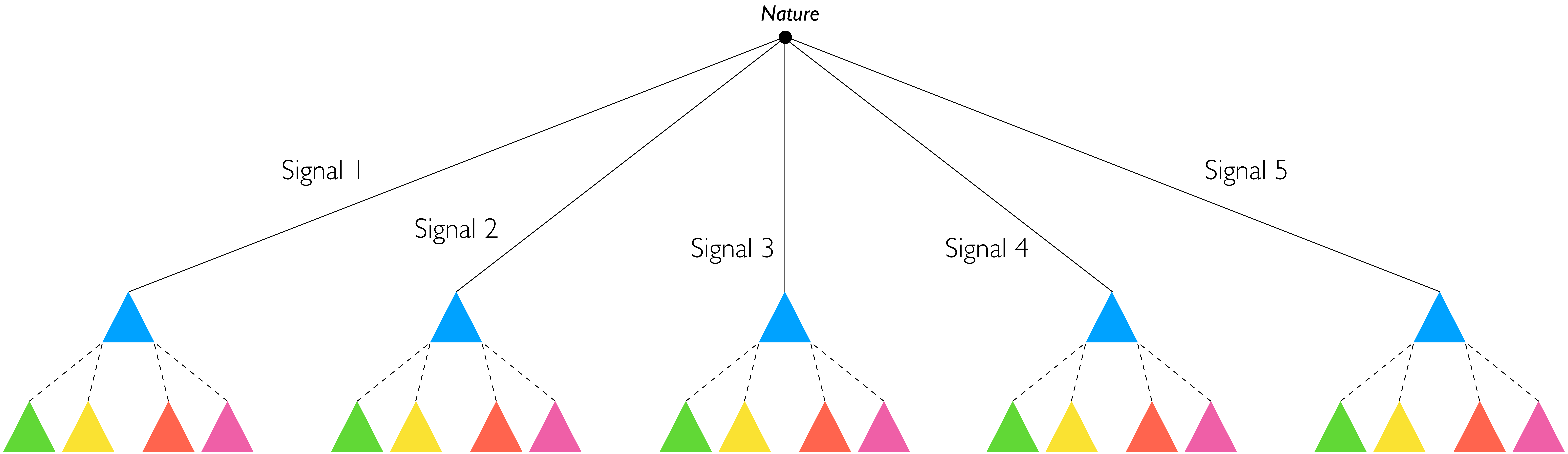
How to construct an abstraction?



How to construct an abstraction?

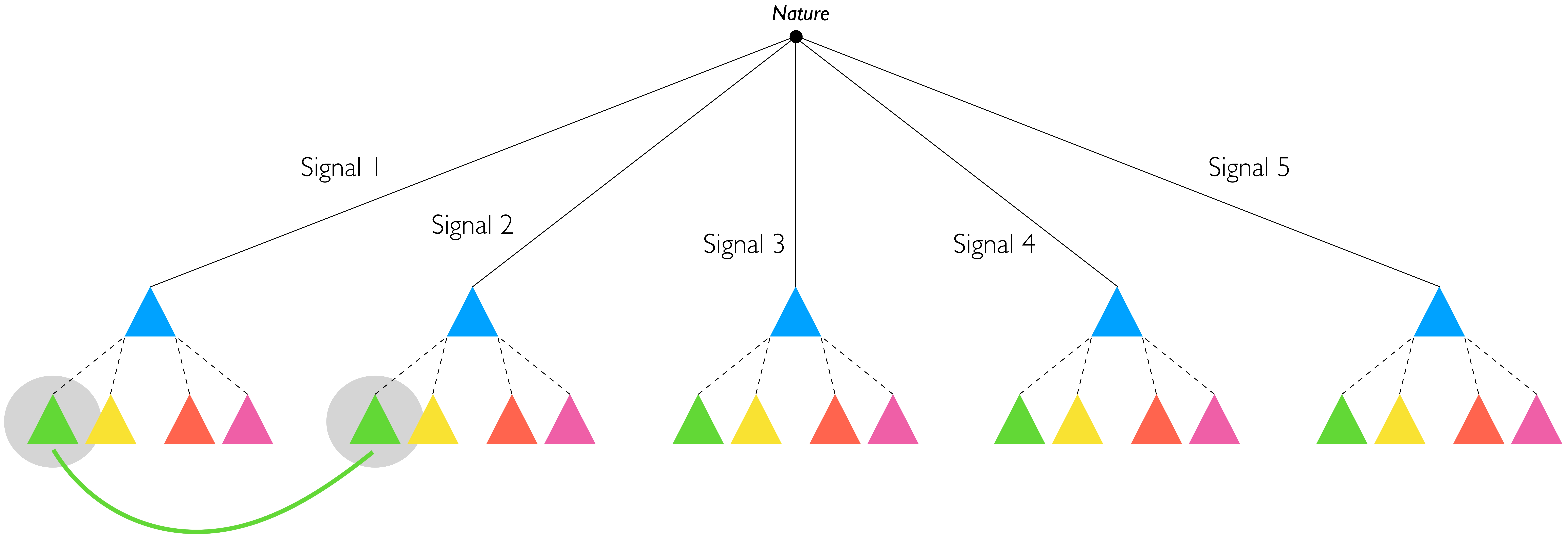


How to construct an abstraction?

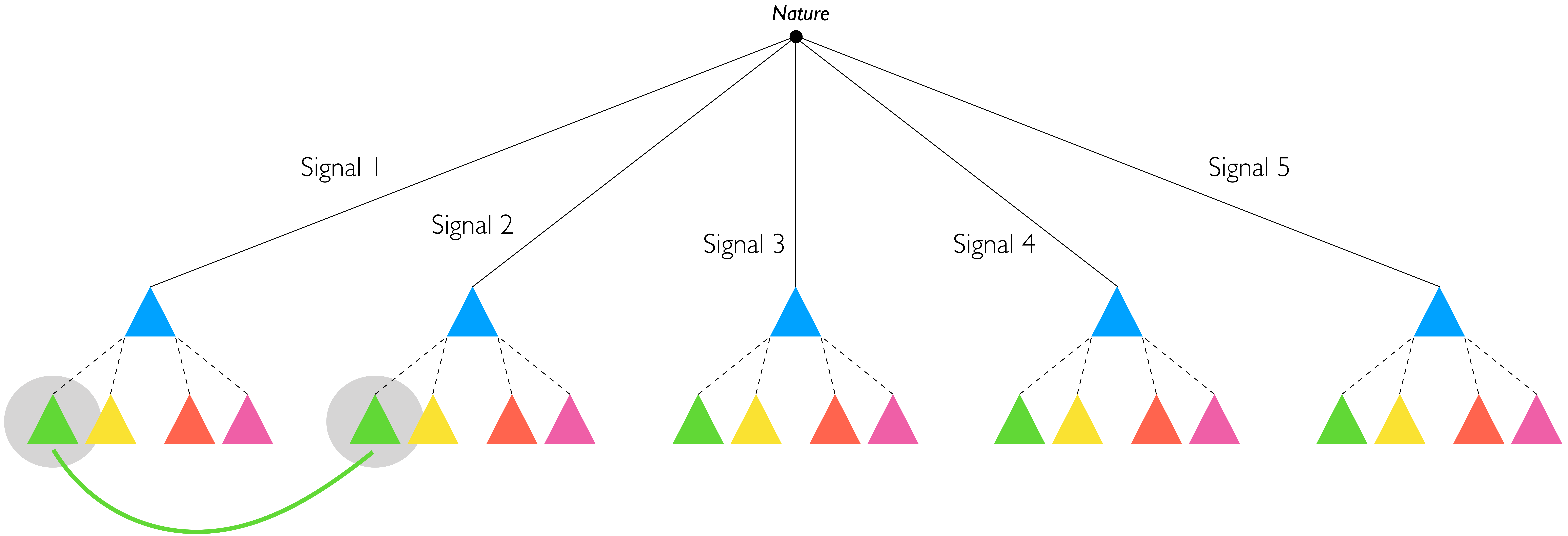


Some possibilities

How to construct an abstraction?



How to construct an abstraction?



Such an abstraction is with imperfect recall

A possible algorithms

- Working top-down level by level
- Aggregating above to enable an aggregation below
- In principle, different approaches for the aggregation