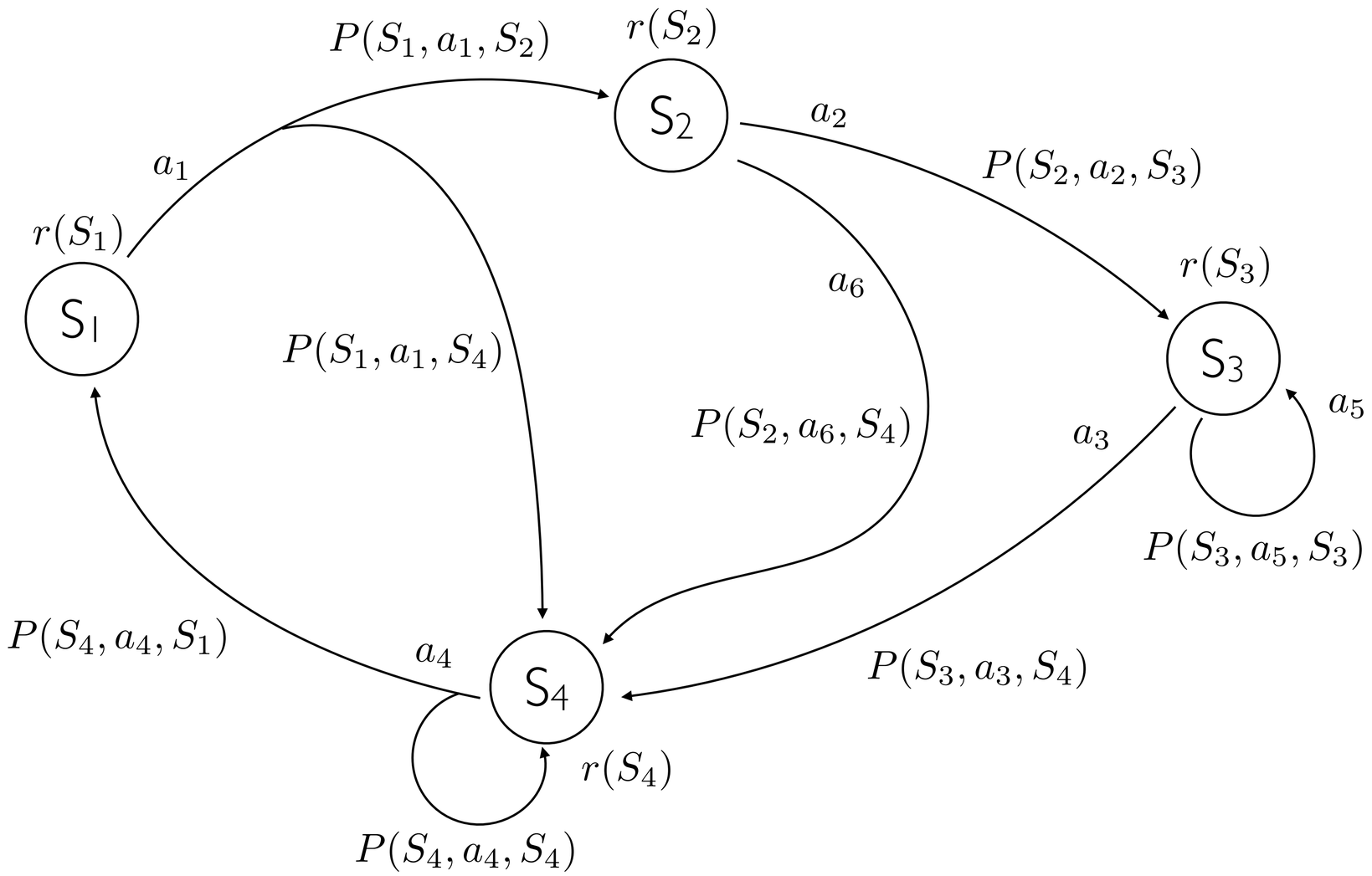
Stochastic games and multi-agent reinforcement learning (MARL) algorithms

POLITECNICO
MILANO 1863

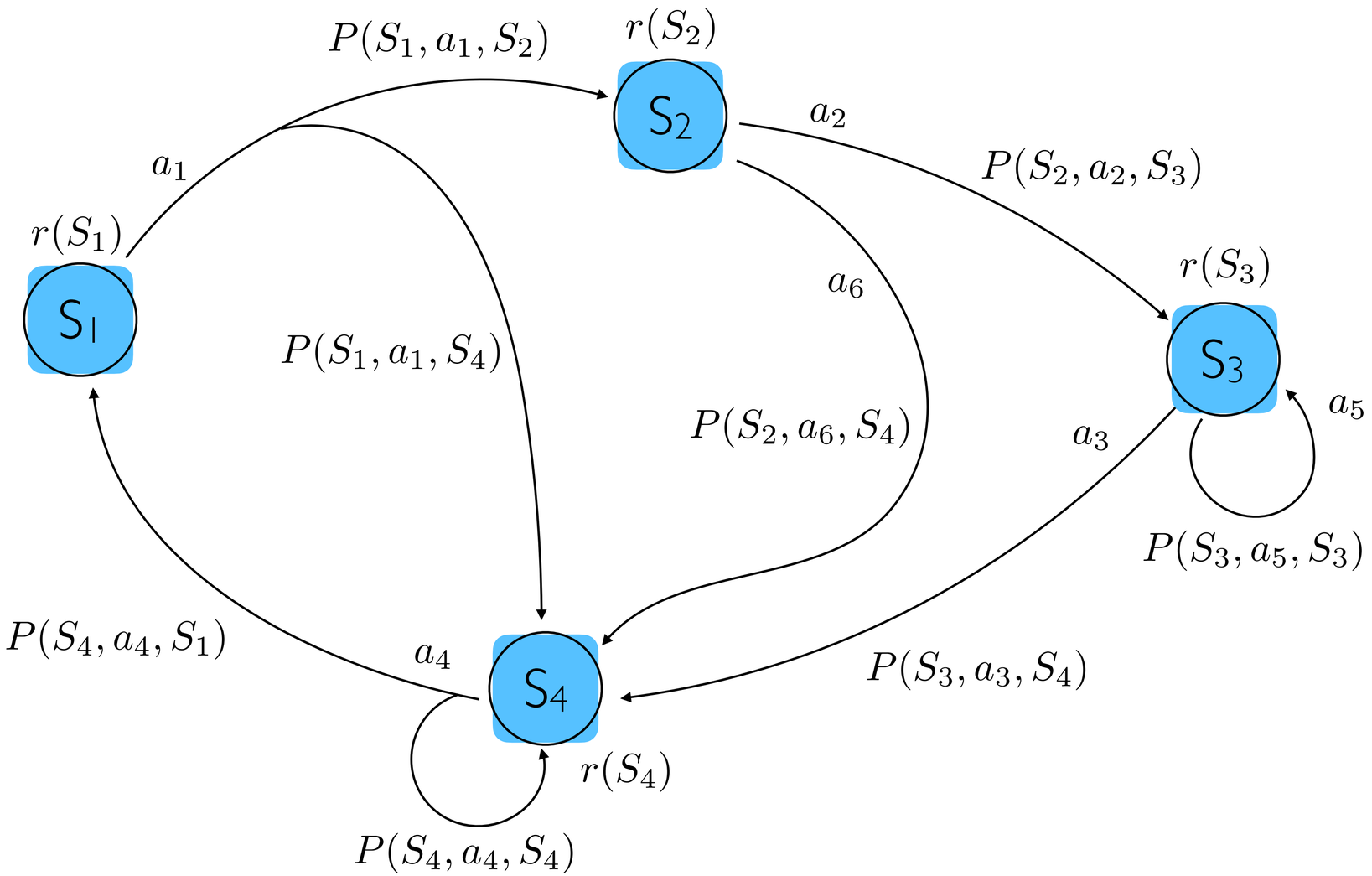
Markov Decision Processes (MDP)

MDP



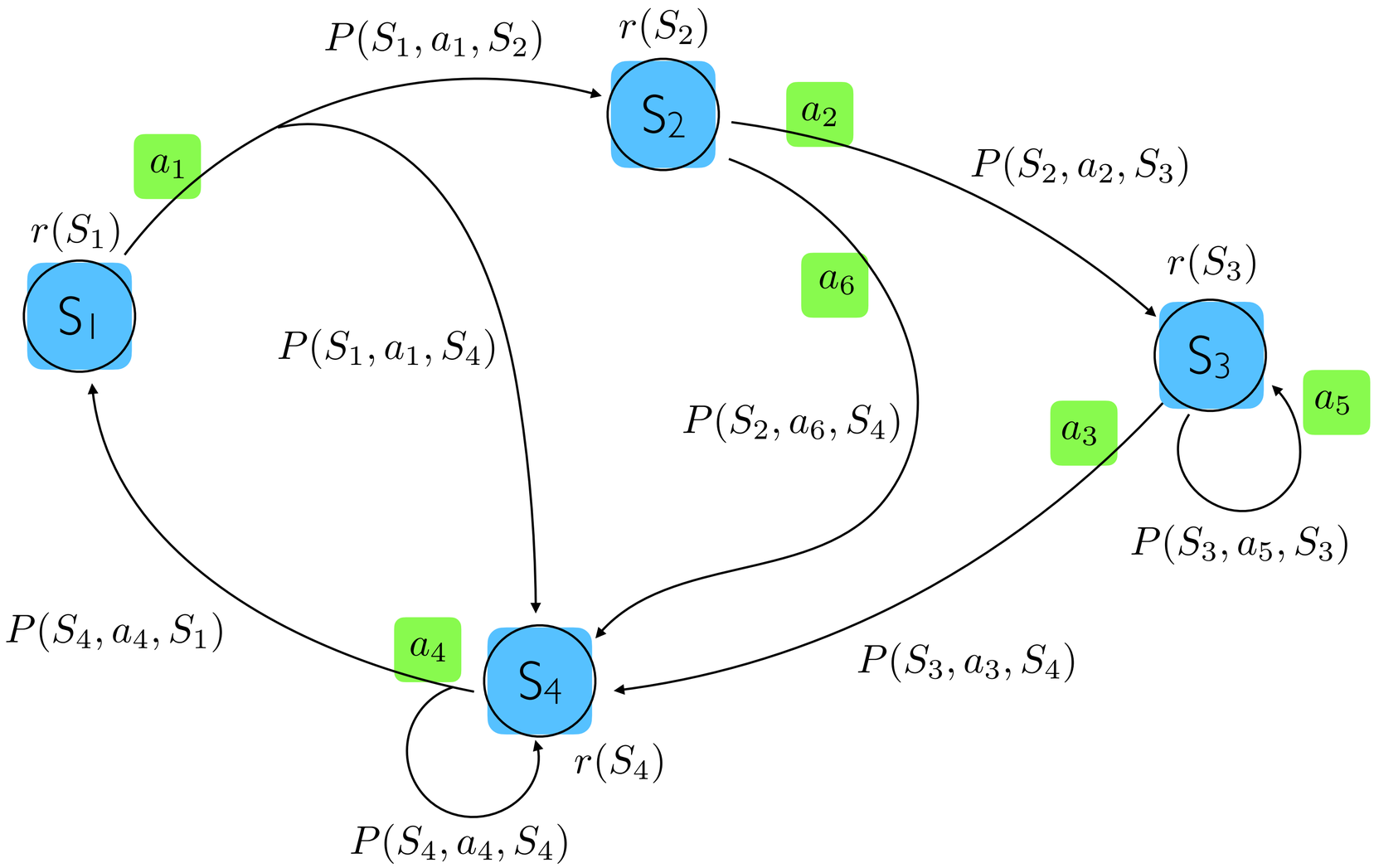
States Actions Transitions Rewards

MDP



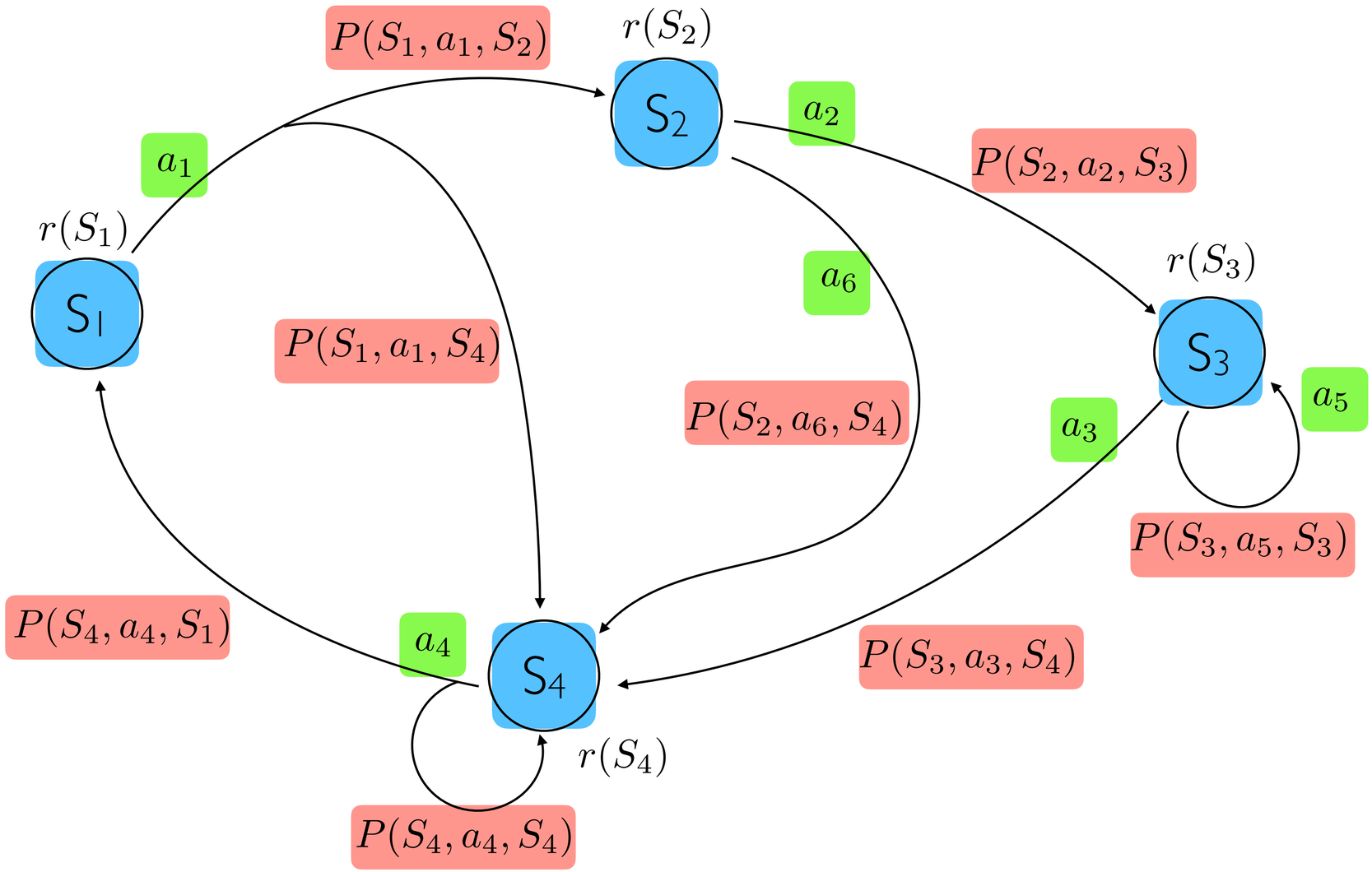
States Actions Transitions Rewards

MDP



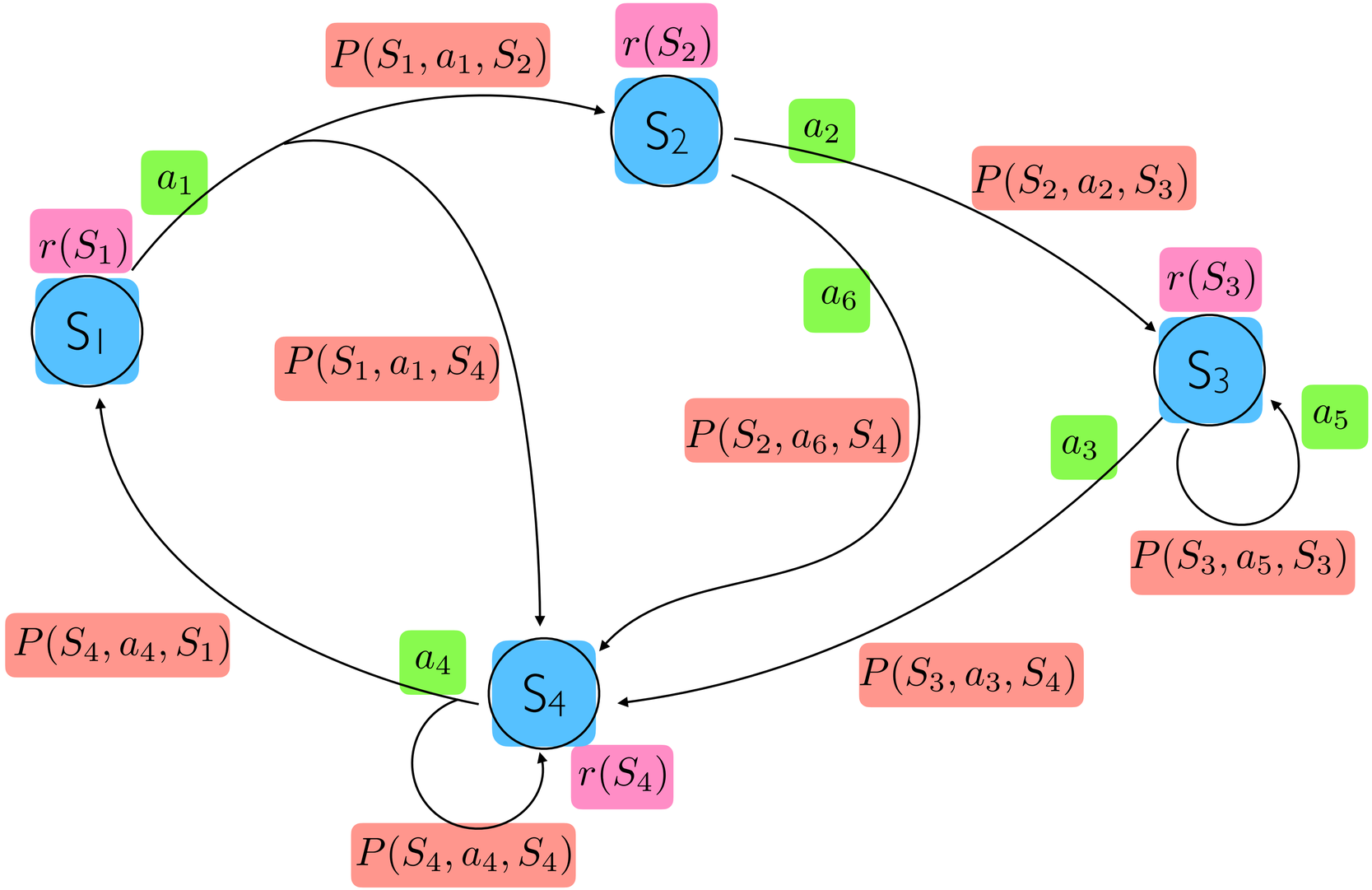
States Actions Transitions Rewards

MDP



States Actions Transitions Rewards

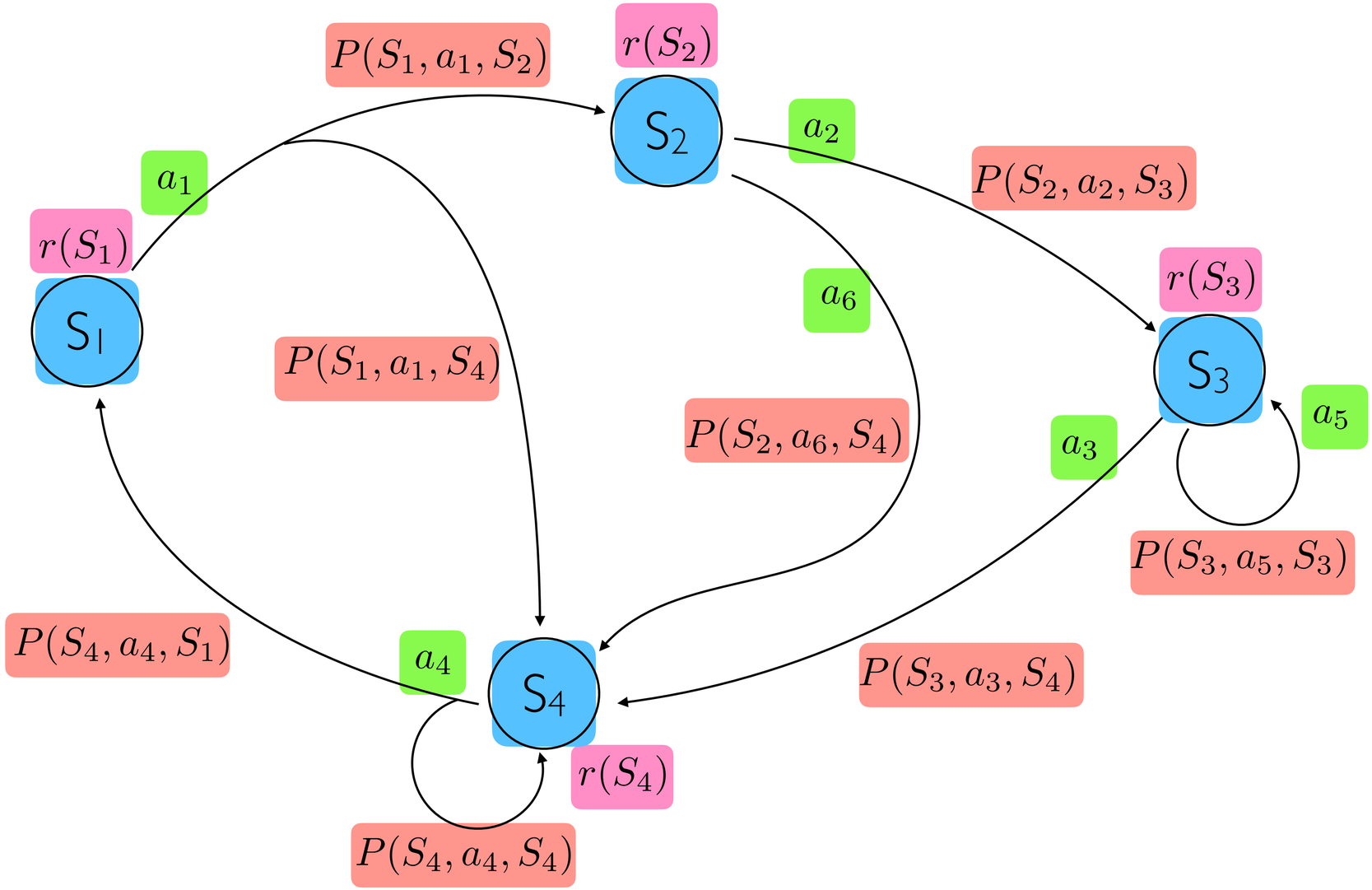
MDP



States Actions Transitions Rewards

MDP

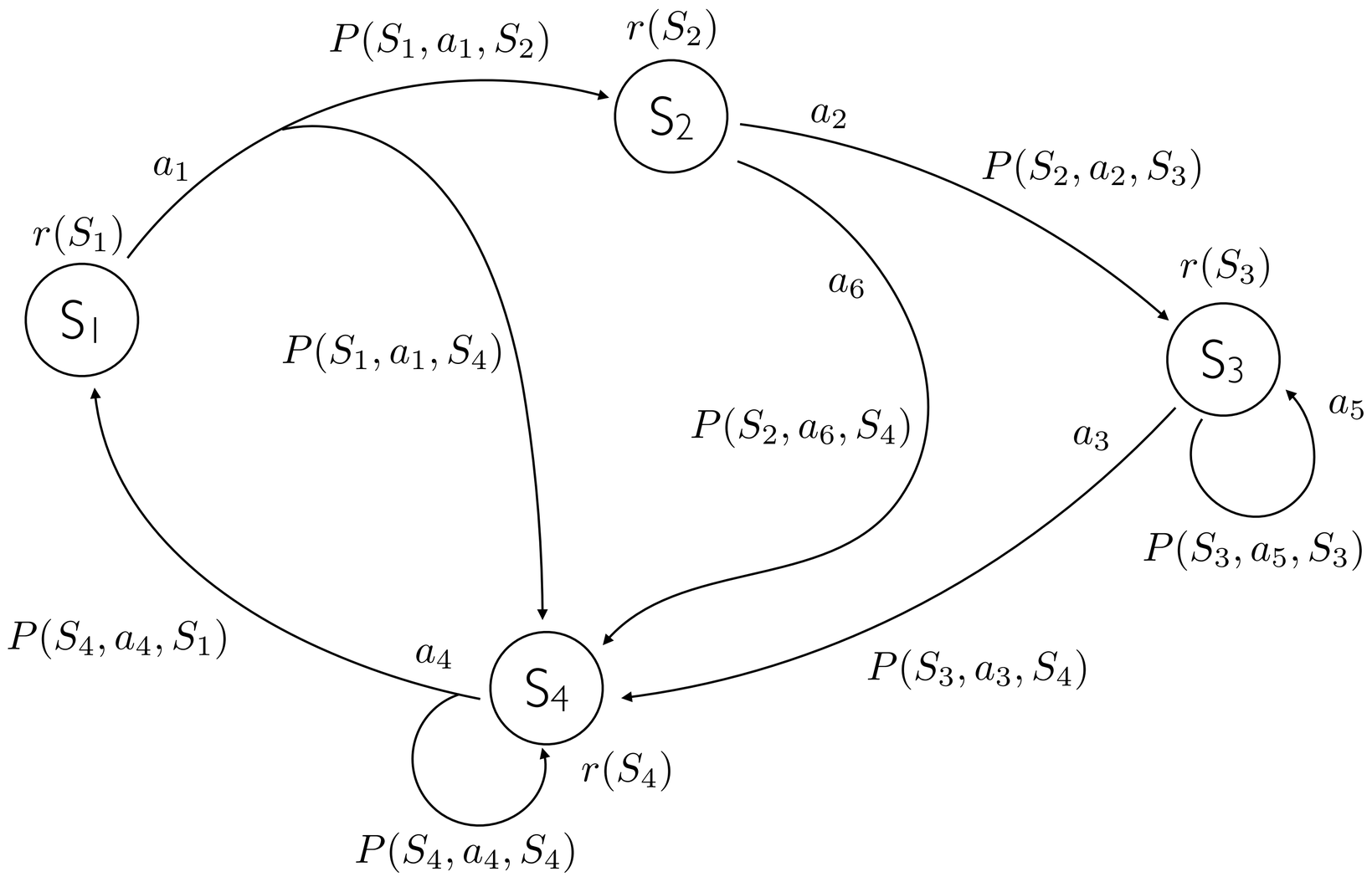
γ Discount factor



States Actions Transitions Rewards

MDP

γ


$P(S_1, a_1, S_2)$
$r(S_2)$
a_2
$P(S_2, a_2, S_3)$
a_1
$r(S_1)$
$r(S_3)$
a_6
$P(S_1, a_1, S_4)$
$P(S_2, a_6, S_4)$
a_5
a_3
$P(S_3, a_5, S_3)$
$P(S_4, a_4, S_1)$
a_4
$P(S_3, a_3, S_4)$
$r(S_4)$
$P(S_4, a_4, S_4)$
S1
S2
S3
S4

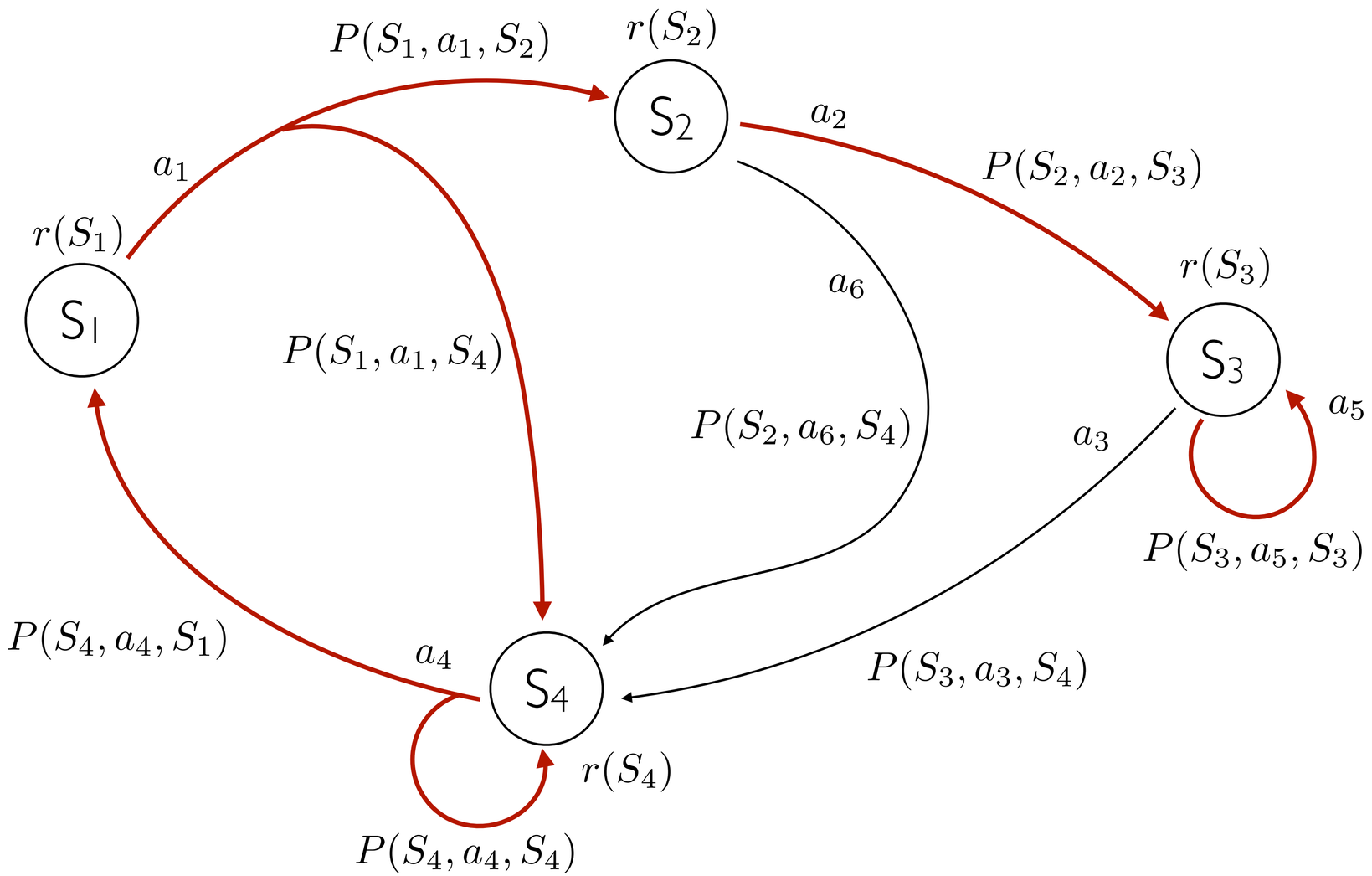
Q-learning

Solving an MDP without information

- Solving an MDP means finding the best policy, mapping each state to a specific action, and maximizing the revenue to the agent

MDP

γ


$P(S_1, a_1, S_2)$
$r(S_2)$
a_2
$P(S_2, a_2, S_3)$
a_1
$r(S_1)$
$r(S_3)$
a_6
$P(S_1, a_1, S_4)$
$P(S_2, a_6, S_4)$
a_5
a_3
$P(S_3, a_5, S_3)$
$P(S_4, a_4, S_1)$
a_4
$P(S_3, a_3, S_4)$
$r(S_4)$
$P(S_4, a_4, S_4)$
S1
S2
S3
S4

Solving an MDP without information

- Solving an MDP means finding the best policy, mapping each state to a specific action, and maximizing the revenue to the agent

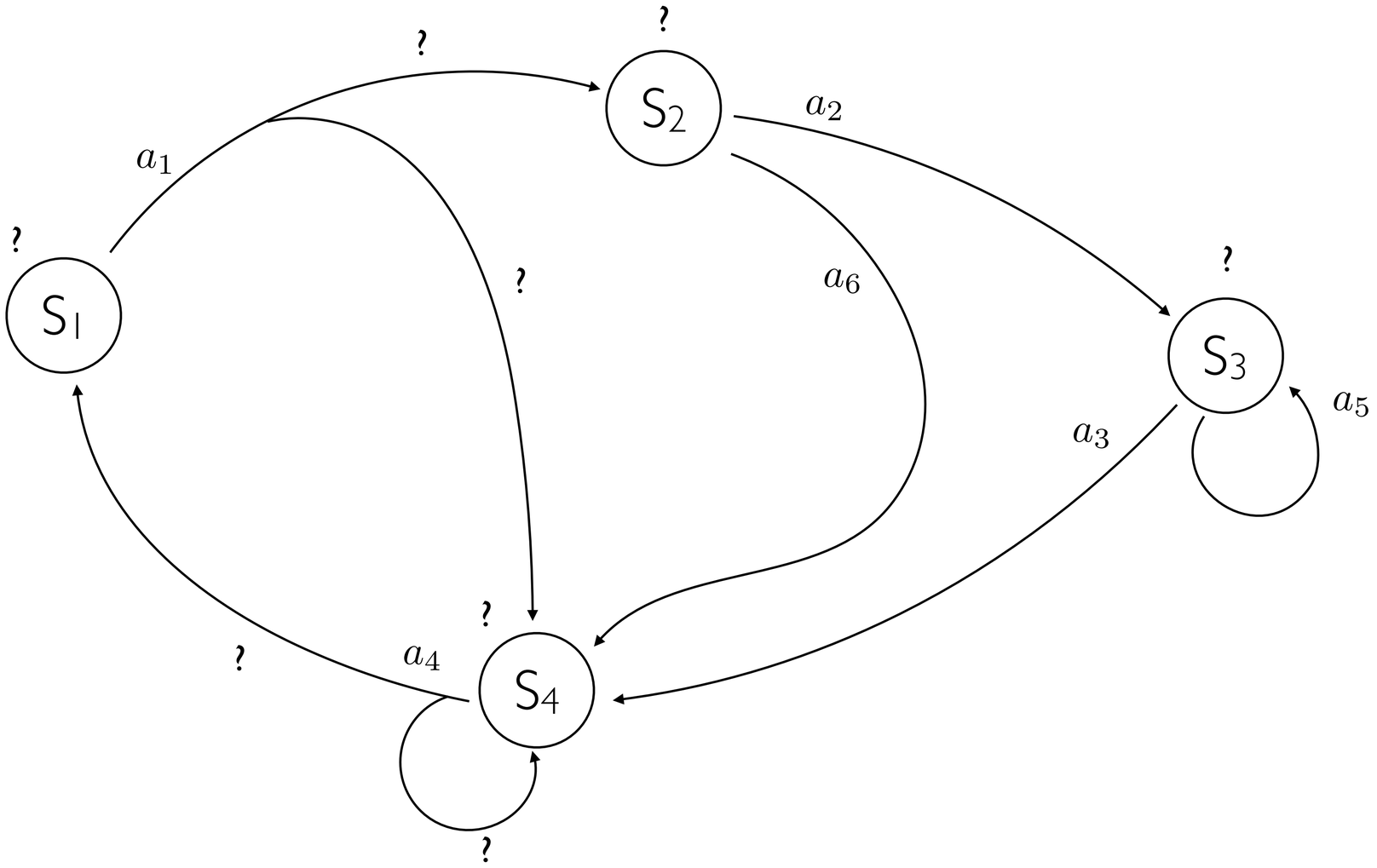
- When the value of all the parameters are known, MDP can be easily solved in polynomial time with linear programming or dynamic programming

Solving an MDP without information

- Solving an MDP means finding the best policy, mapping each state to a specific action, and maximizing the revenue to the agent
- When the value of all the parameters are known, MDP can be easily solved in polynomial time with linear programming or dynamic programming
- When some (or all) parameters are unknown (e.g., rewards and/or transition probabilities), the Q-learning algorithm can be used

MDP

γ



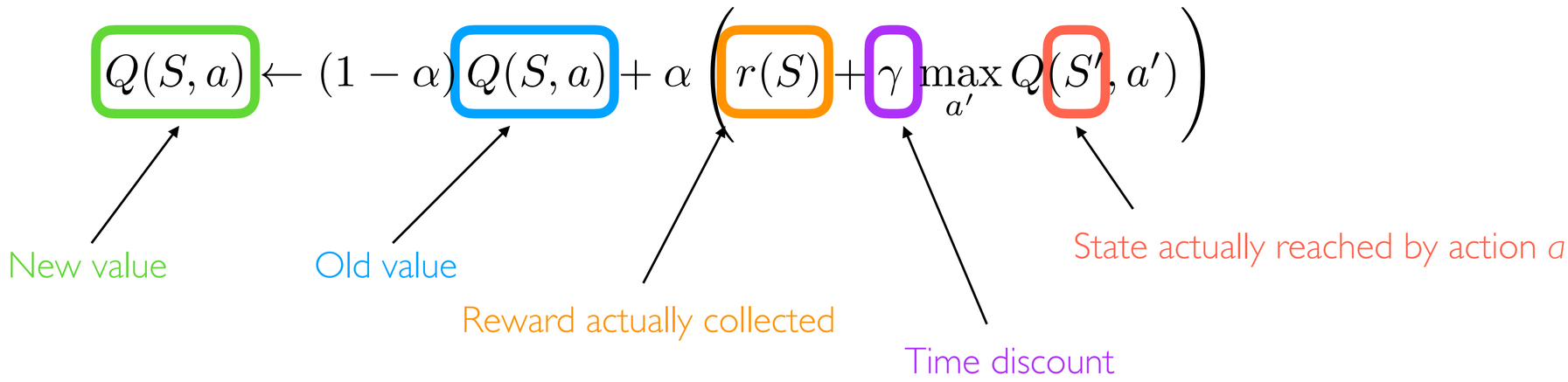
Q-learning algorithm (1)

- The Q-learning algorithm builds a Q-table in which storing the Q-values, corresponding to the estimate of the values the agent expects to gain from playing a given action in a given state

Q-learning algorithm (1)

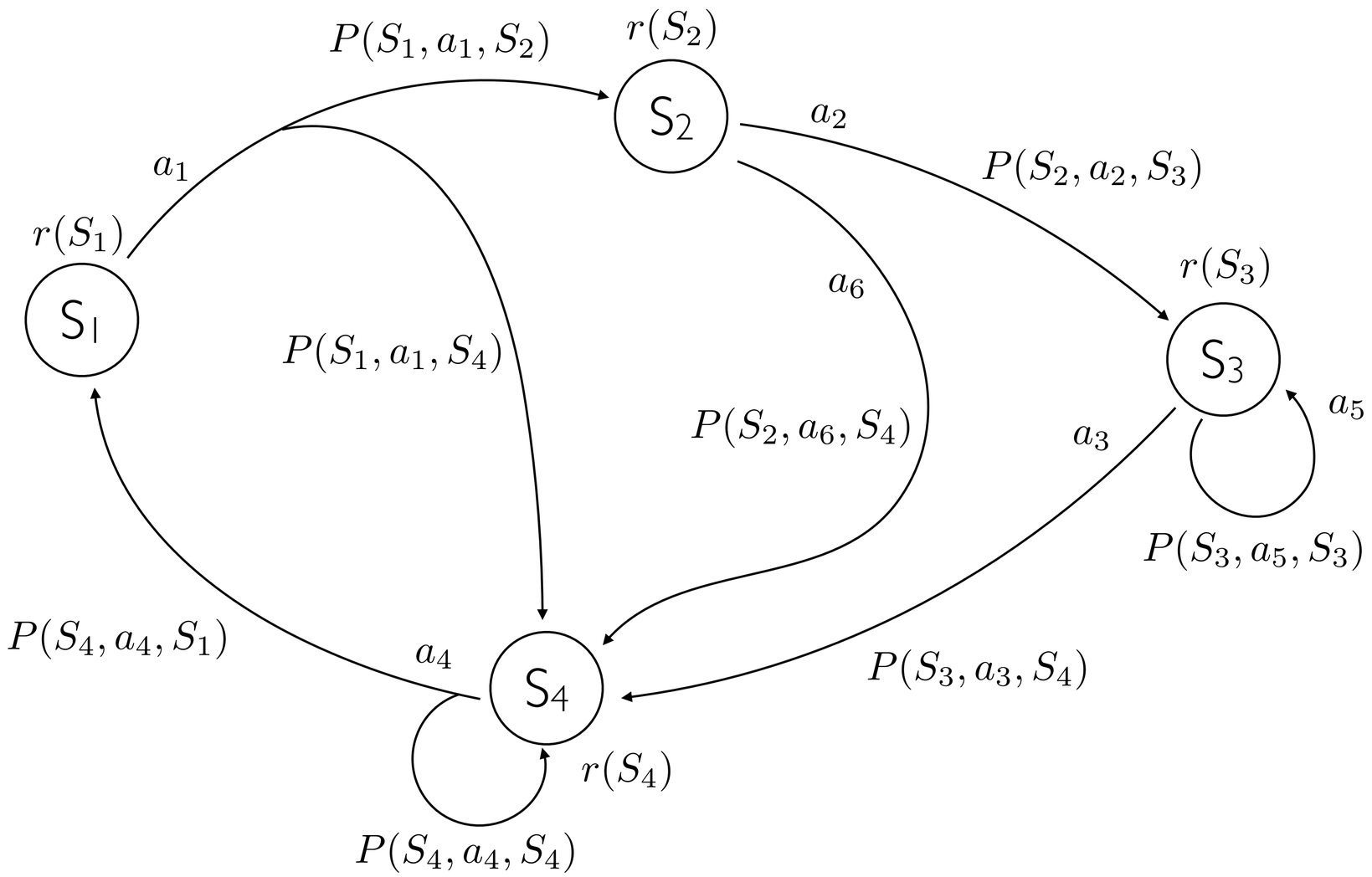
- The Q-learning algorithm builds a Q-table in which storing the Q-values, corresponding to the estimate of the values the agent expects to gain from playing a given action in a given state

- The agent moves over the MDP and Q-values are updated according to its experience as follows:



MDP

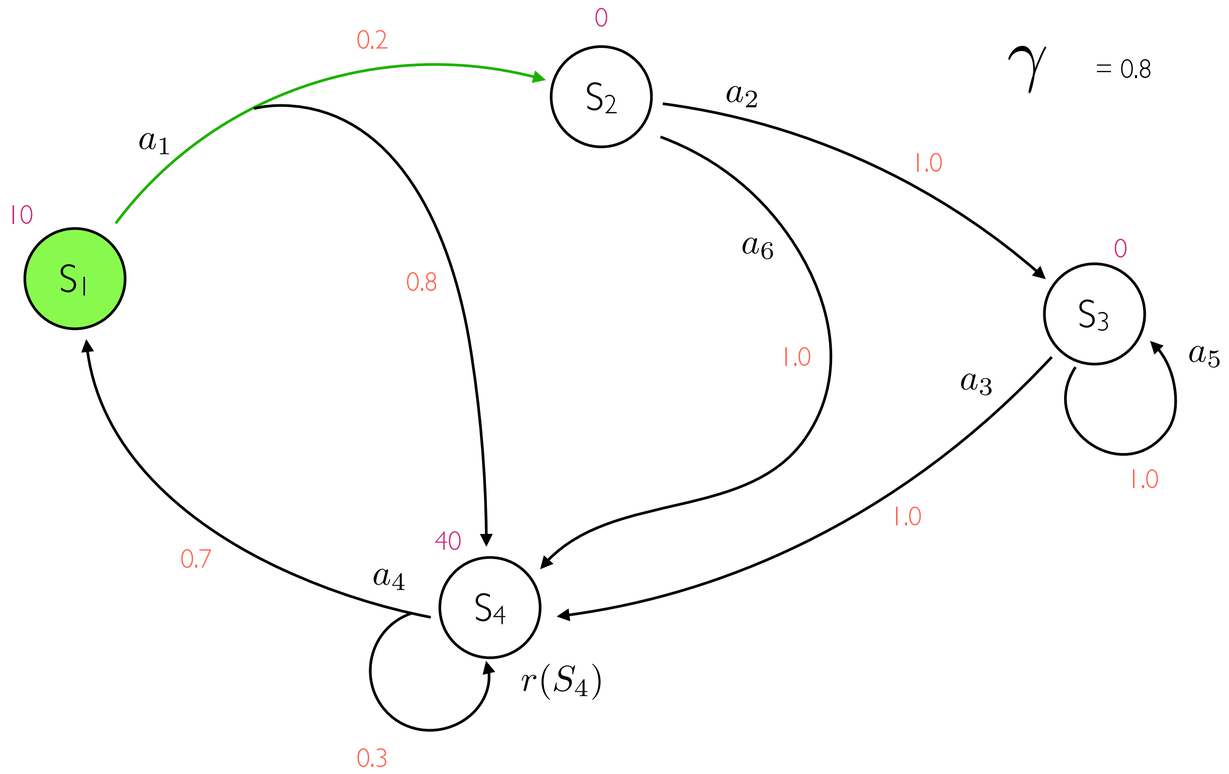
γ


$P(S_1, a_1, S_2)$
$r(S_2)$
a_2
$P(S_2, a_2, S_3)$
a_1
$r(S_1)$
$r(S_3)$
a_6
$P(S_1, a_1, S_4)$
$P(S_2, a_6, S_4)$
a_5
a_3
$P(S_3, a_5, S_3)$
$P(S_4, a_4, S_1)$
a_4
$P(S_3, a_3, S_4)$
$r(S_4)$
$P(S_4, a_4, S_4)$
S1
S2
S3
S4

Q-table

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	0					
	S2		0				0
	S3			0		0	
	S4				0		

MDP



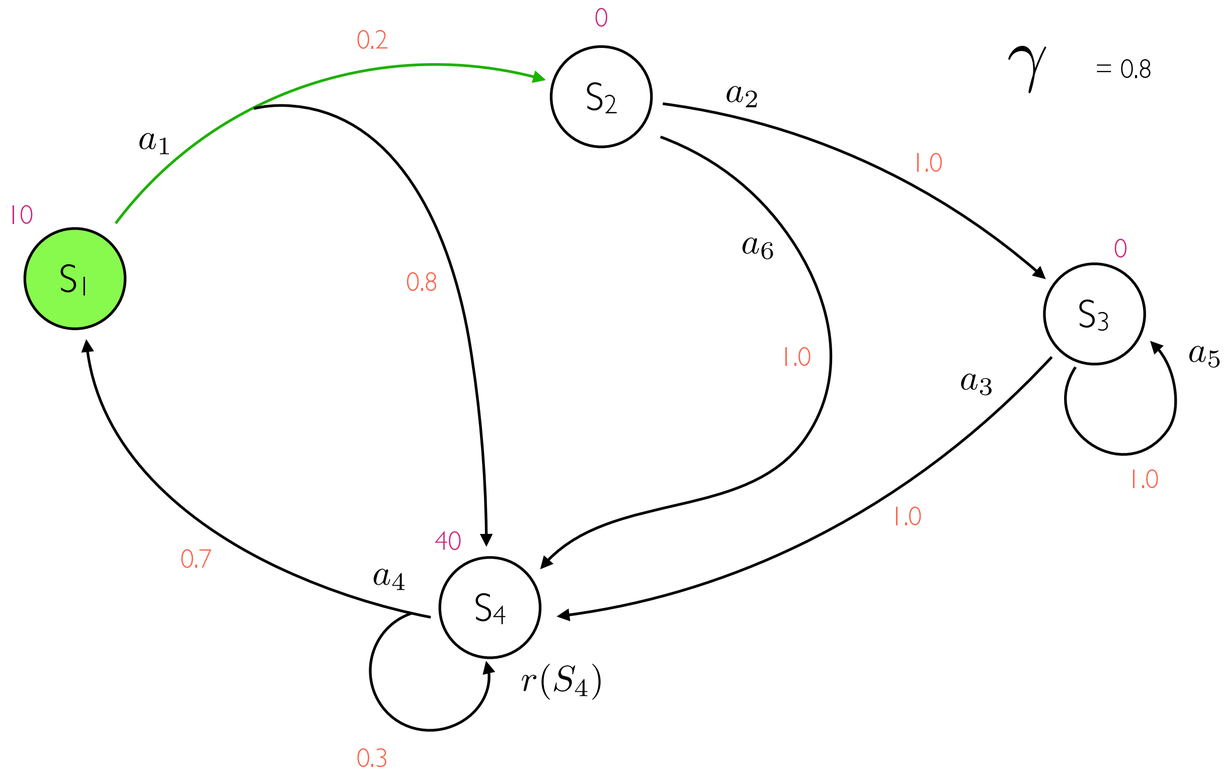
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	0					
	S2		0				0
	S3			0		0	
	S4				0		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

5 0.5 0.0 0.5 10 0.8 0.0

MDP



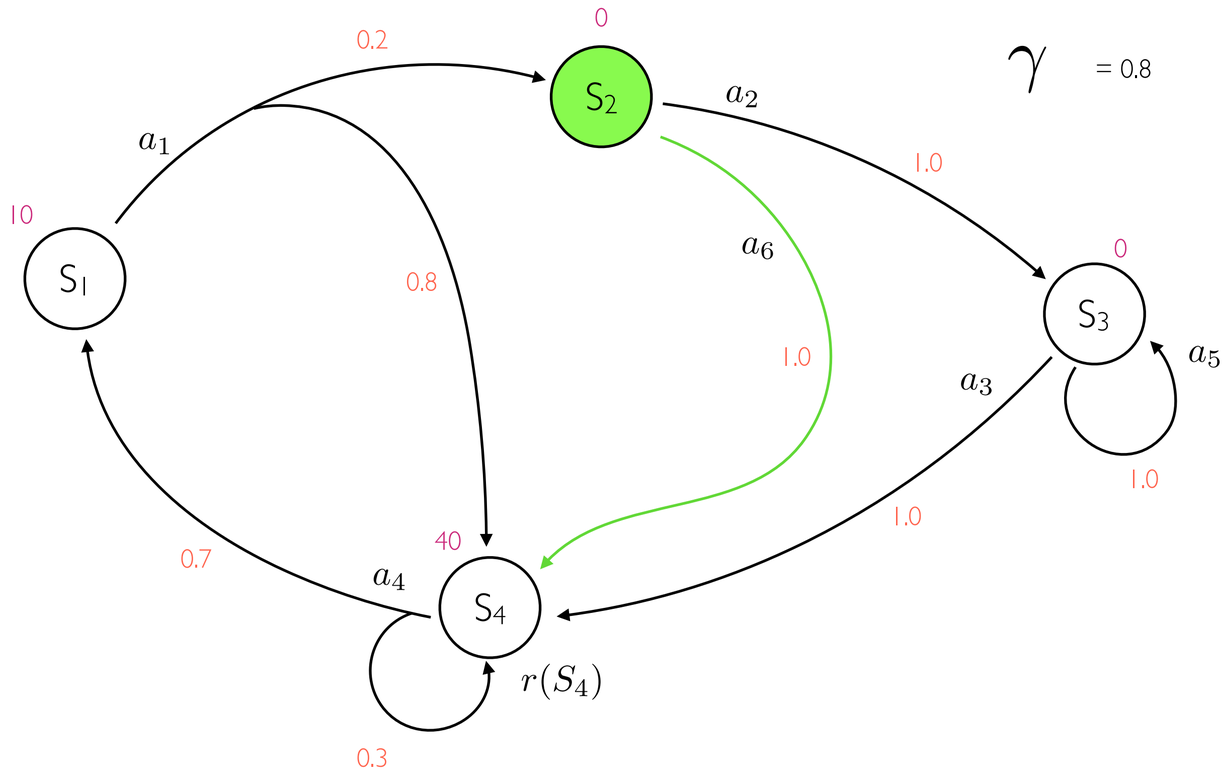
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	**5**					
	S2		0				0
	S3			0		0	
	S4				0		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

5 0.5 0.0 0.5 10 0.8 0.0

MDP



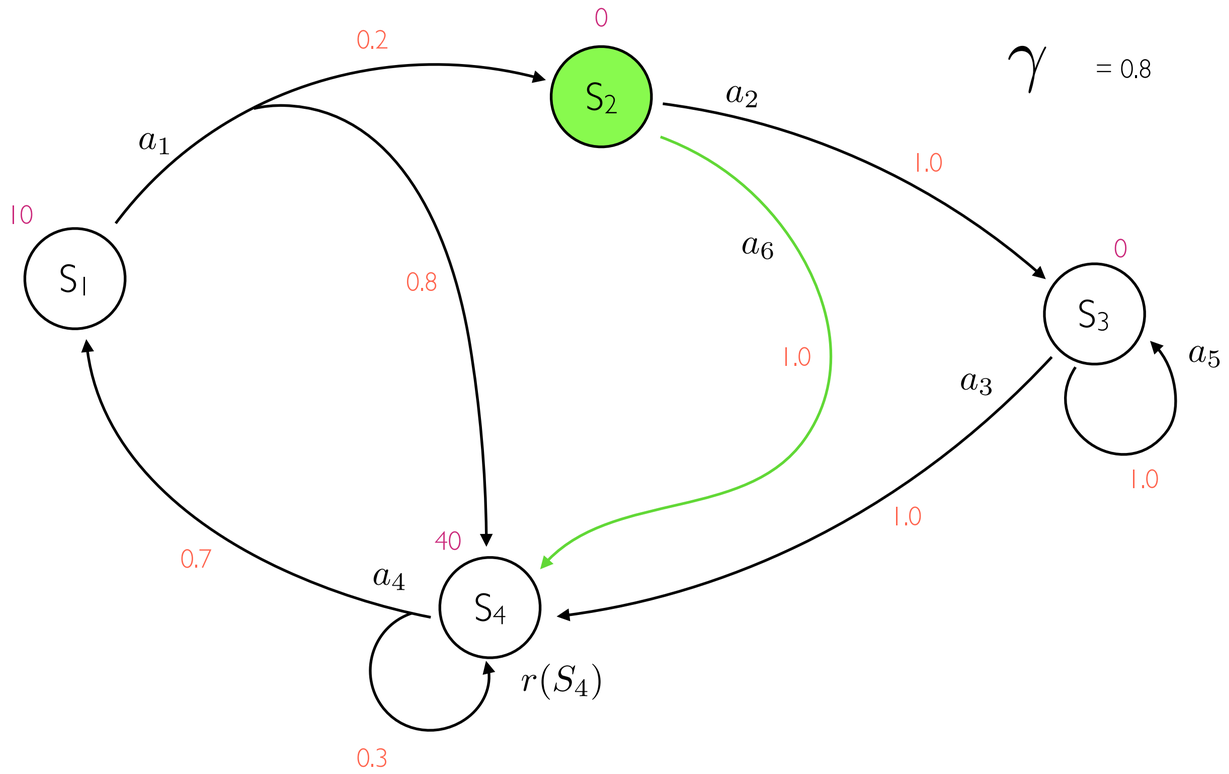
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	5					
	S2		0				0
	S3			0		0	
	S4				0		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

0 0.5 0.0 0.5 0 0.8 0.0

MDP



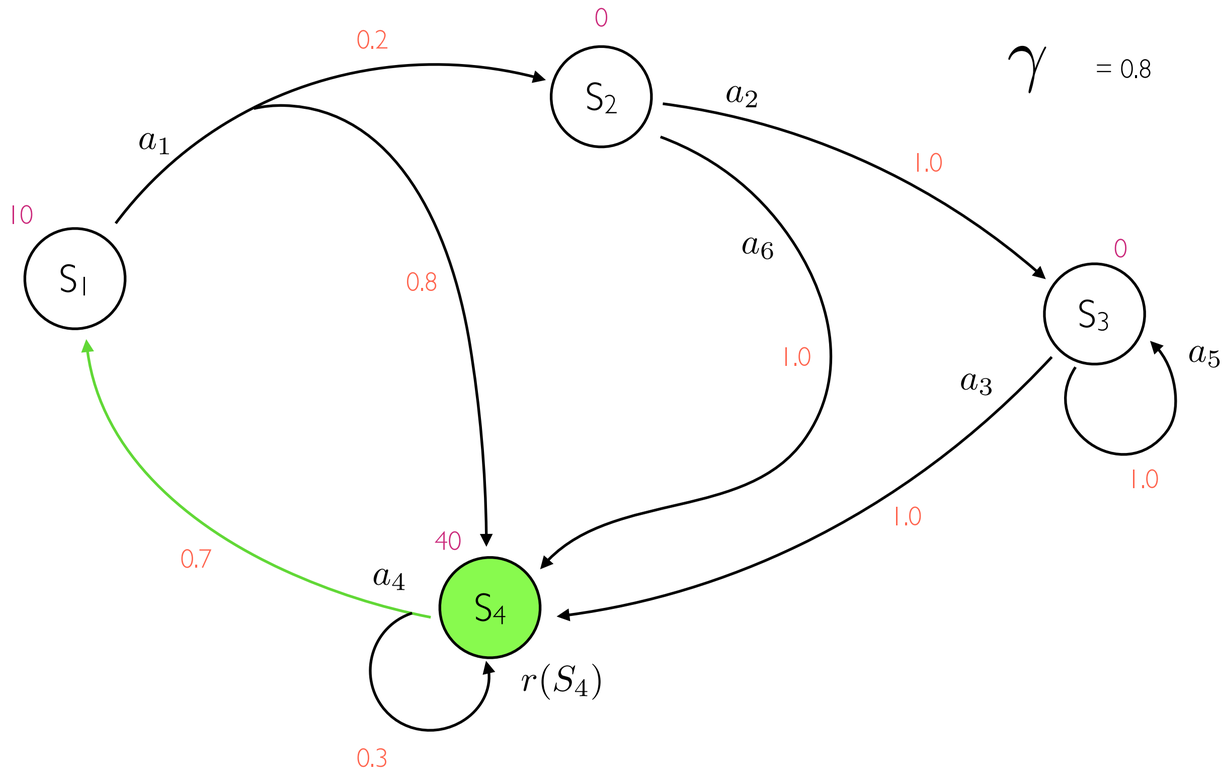
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	5					
	S2		0				**0**
	S3			0		0	
	S4				0		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

0 0.5 0.0 0.5 0 0.8 0.0

MDP



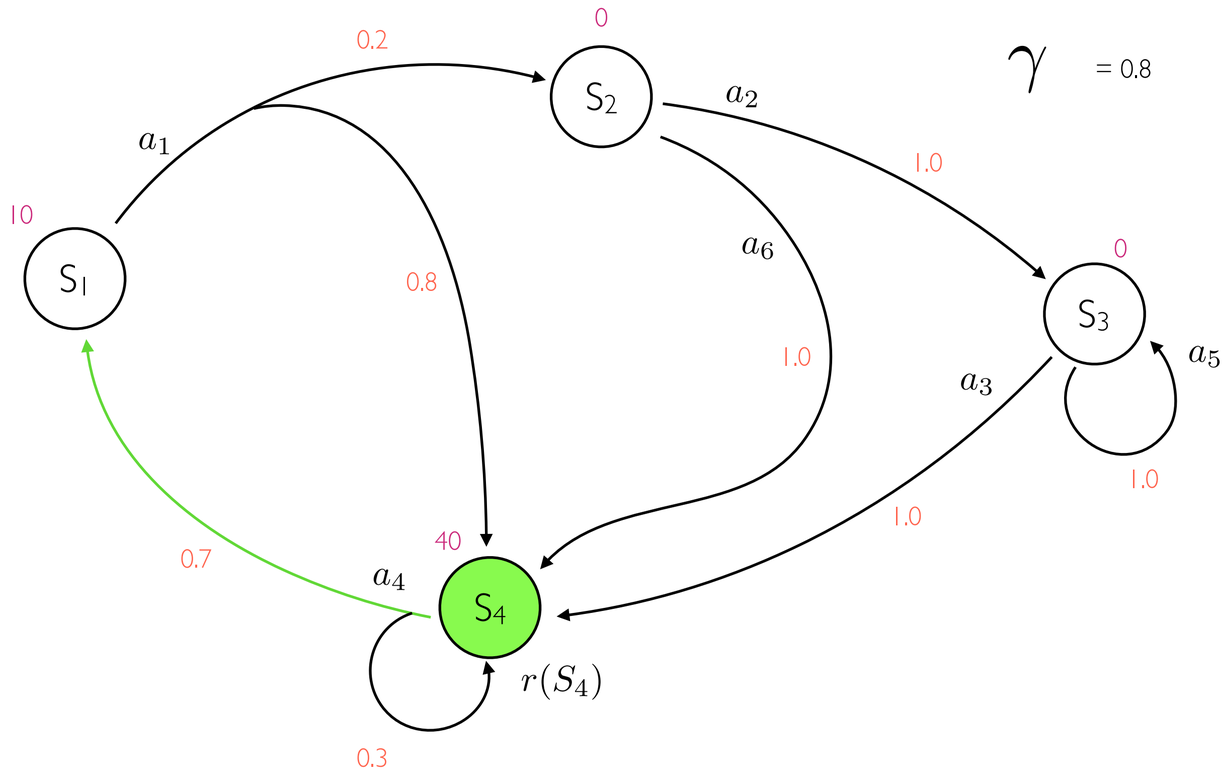
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	5					
	S2		0				0
	S3			0		0	
	S4				0		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

22 0.5 0.0 0.5 40 0.8 5

MDP



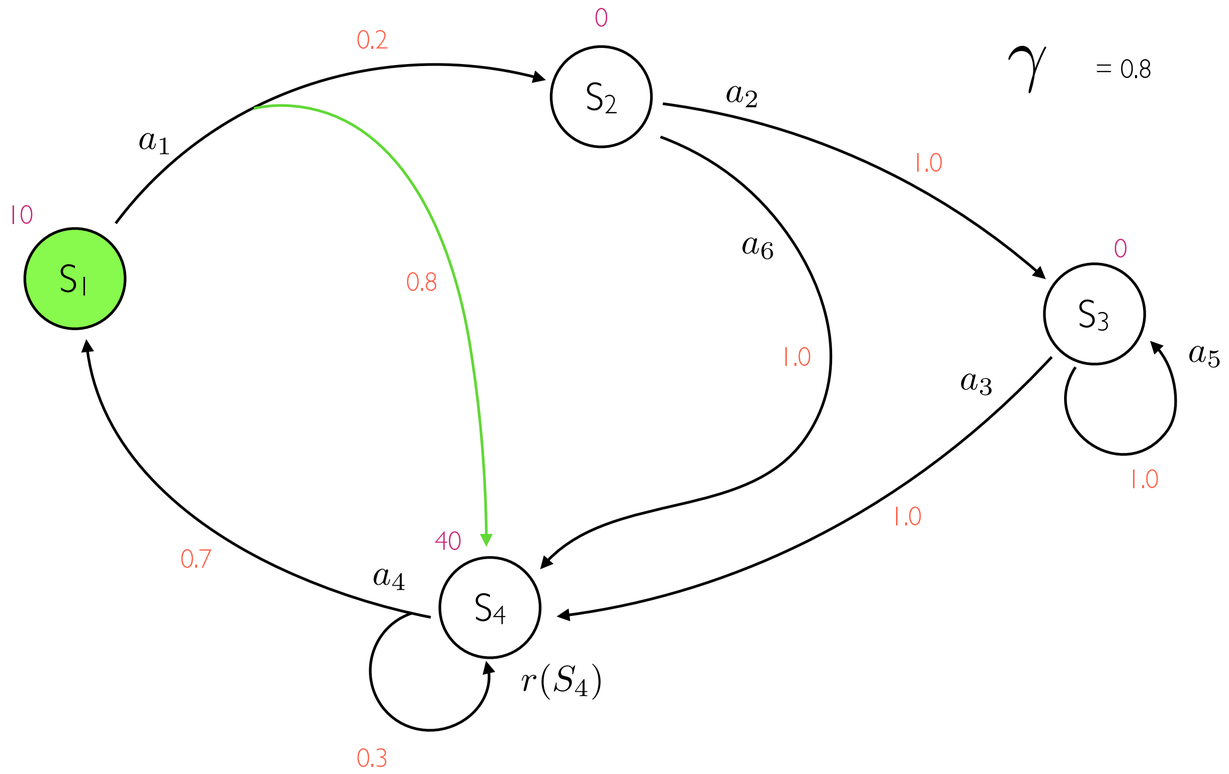
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	5					
	S2		0				0
	S3			0		0	
	S4				**22**		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

22 0.5 0.0 0.5 40 0.8 5

MDP



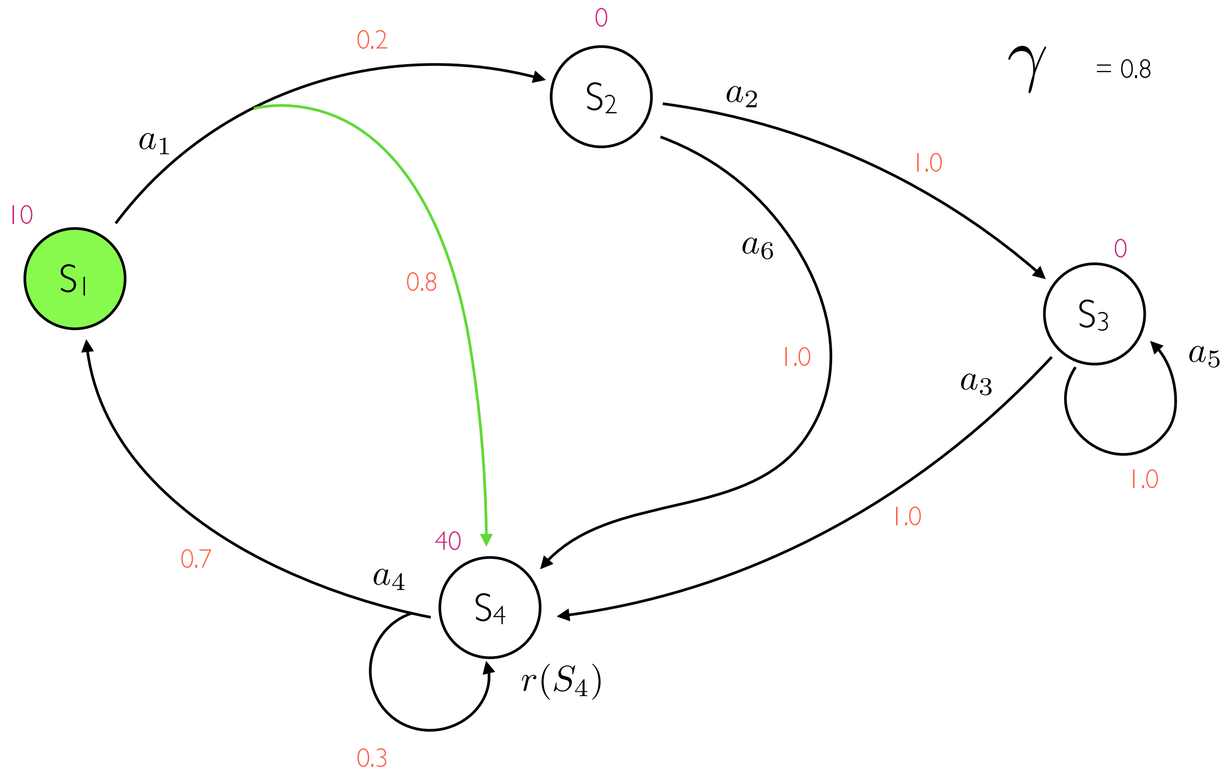
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	5					
	S2		0				0
	S3			0		0	
	S4				22		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

16.3 0.5 5 0.5 5 0.8 22

MDP



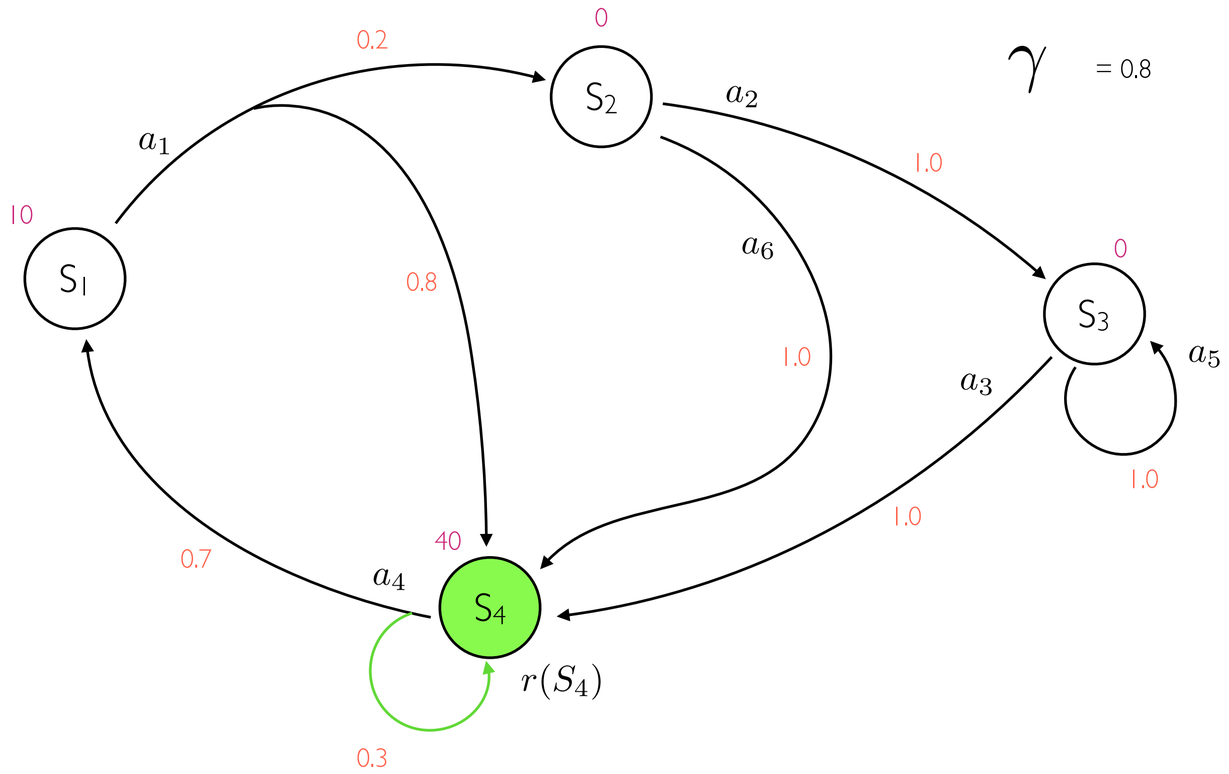
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	**16.3**					
	S2		0				0
	S3			0		0	
	S4				22		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

16.3 0.5 5 0.5 5 0.8 22

MDP



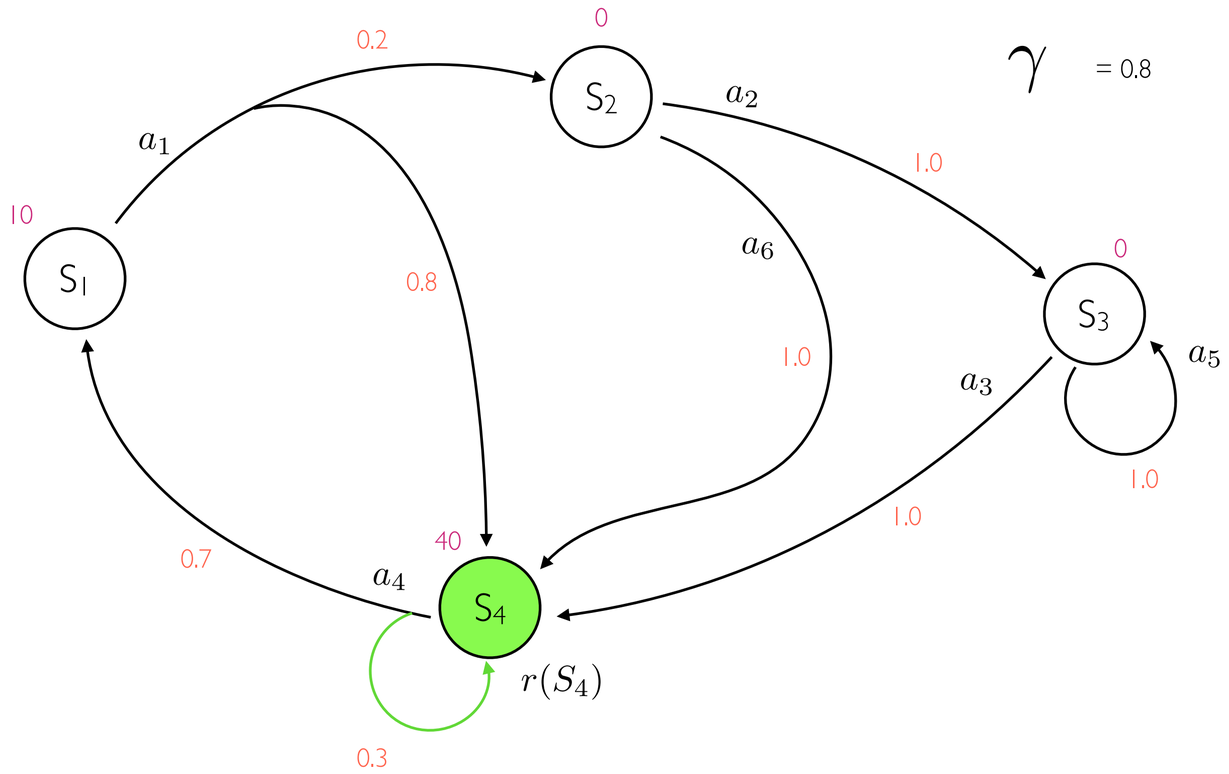
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	15.5					
	S2		0				0
	S3			0		0	
	S4				22		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

39.8 0.5 5 0.5 40 0.8 22

MDP



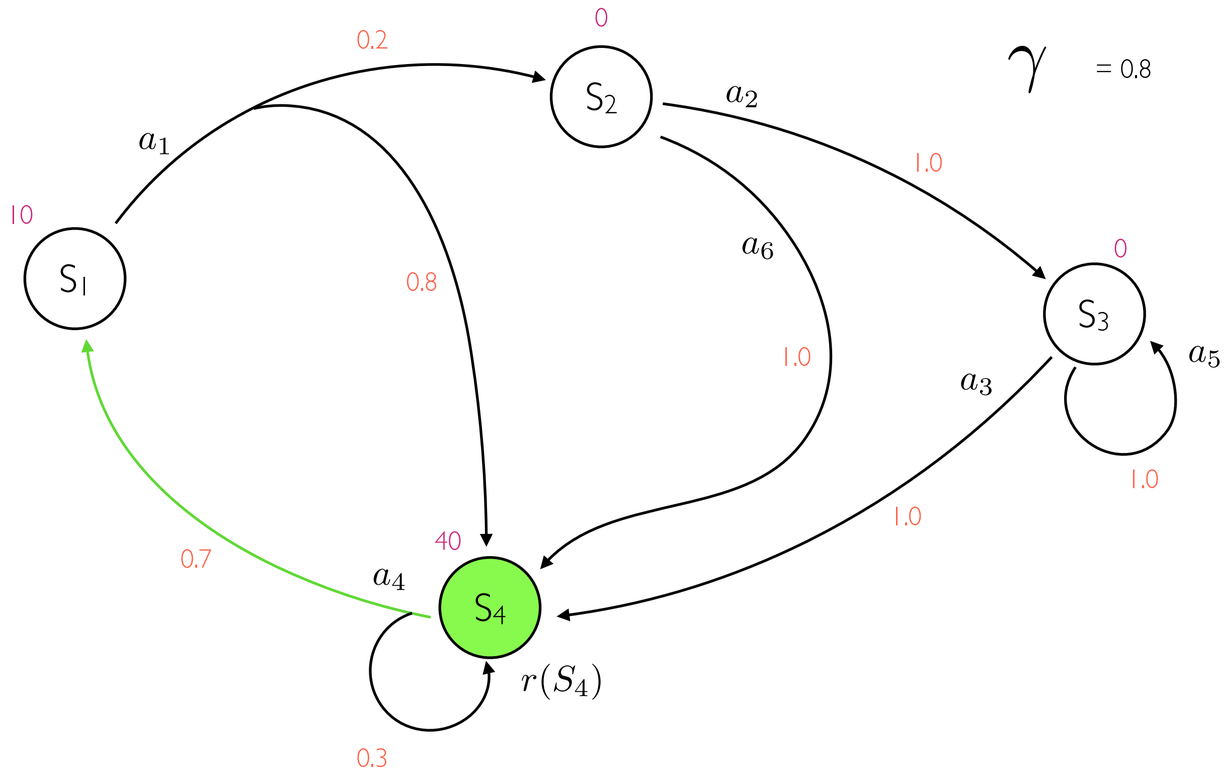
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	15.5					
	S2		0				0
	S3			0		0	
	S4				**39.8**		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

39.8 0.5 5 0.5 40 0.8 22

MDP



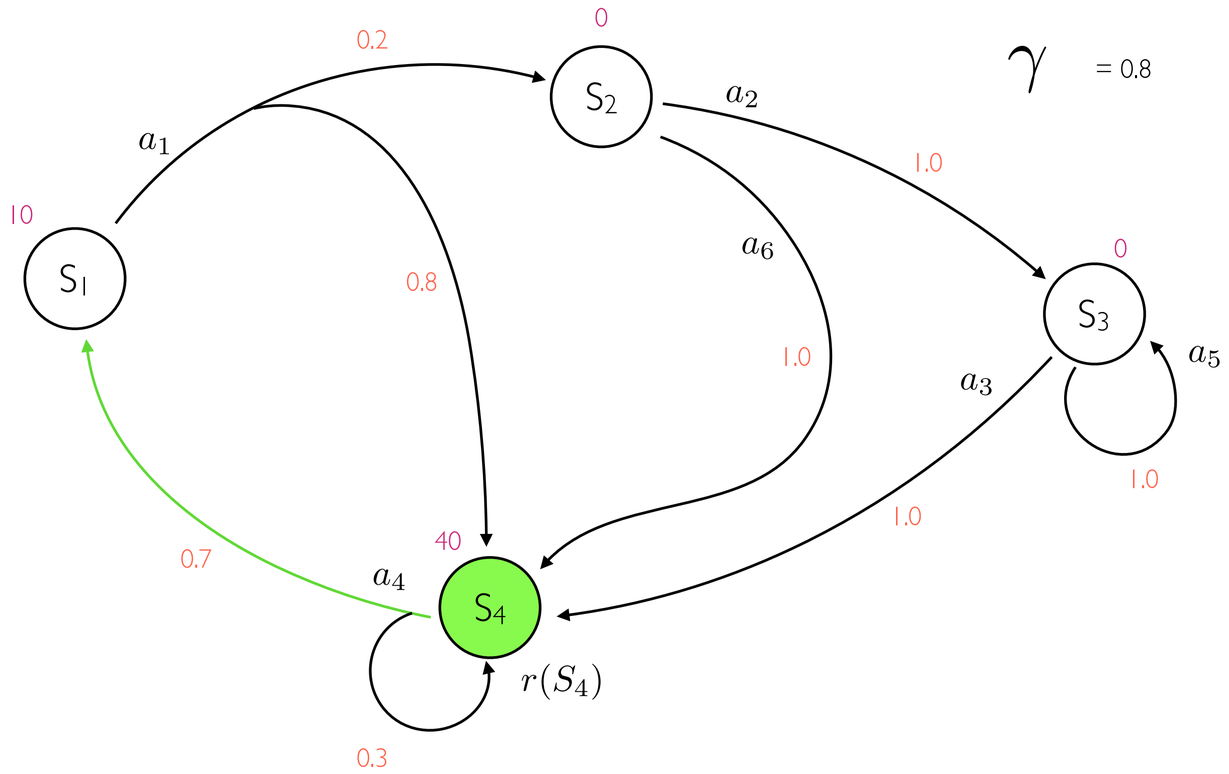
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	15.5					
	S2		0				0
	S3			0		0	
	S4				39.8		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

46.1 0.5 39.8 0.5 40 0.8 15.5

MDP



γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	15.5					
	S2		0				0
	S3			0		0	
	S4				**46.1**		

$$Q(S,a) \leftarrow (1-\alpha)\, Q(S,a) + \alpha \left(r(S) + \gamma \max_{a'} Q(S',a') \right)$$

46.1 0.5 39.8 0.5 40 0.8 15.5

Q-learning algorithm (2)

- We can follow several criteria to choose the action to play in every state multiple actions are available

Q-learning algorithm (2)

- We can follow several criteria to choose the action to play in every state multiple actions are available

  - **Greedy** $a \in \arg\max_{a'} Q(S, a')$

Q-learning algorithm (2)

- We can follow several criteria to choose the action to play in every state multiple actions are available
  - **Greedy** $a \in \arg\max_{a'} Q(S, a')$
  - **Epsilon-greedy** $a \in \begin{cases} \arg\max_{a'} Q(S, a') & \text{with probability } 1 - \epsilon \\ \text{a random action} & \text{with probability } \epsilon \end{cases}$

Q-learning algorithm (2)

- We can follow several criteria to choose the action to play in every state multiple actions are available

  - **Greedy** $a \in \arg\max_{a'} Q(S, a')$

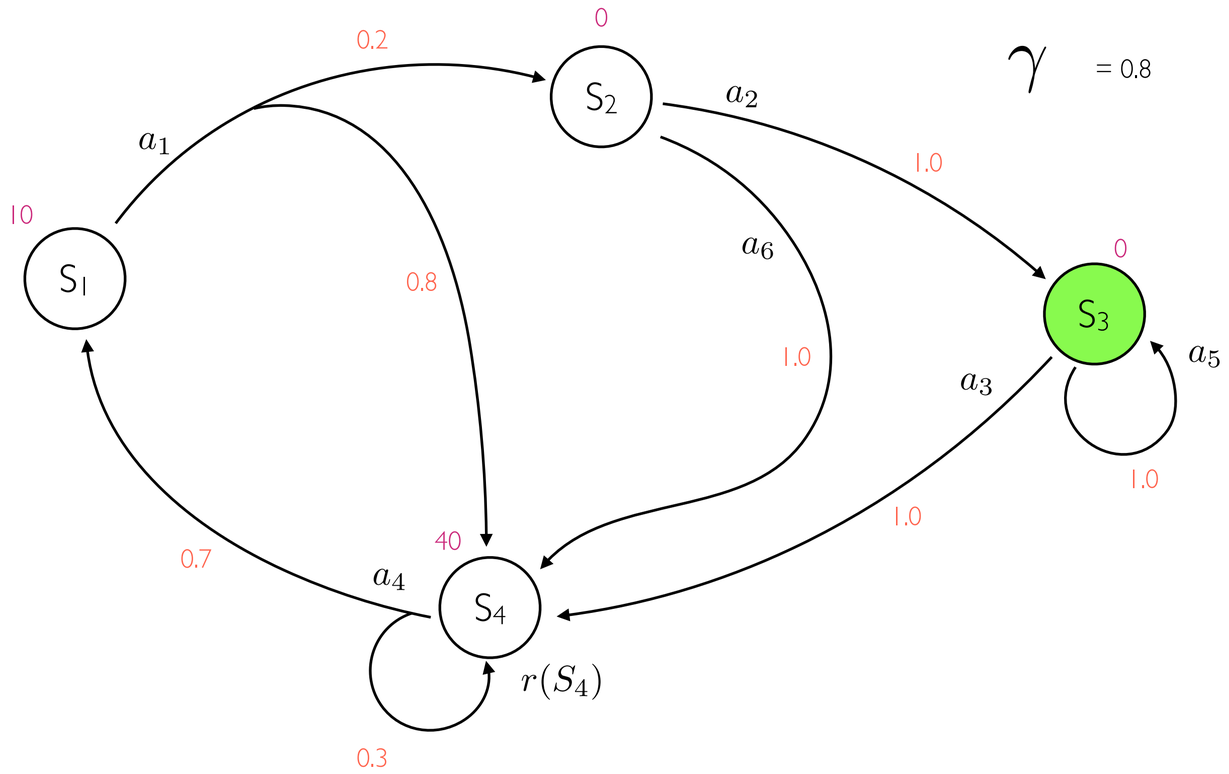
  - **Epsilon-greedy** $a \in \begin{cases} \arg\max_{a'} Q(S, a') & \text{with probability } 1 - \epsilon \\ \text{a random action} & \text{with probability } \epsilon \end{cases}$

  - **Boltzmann exploration** $a \sim \frac{\exp\left(\frac{Q(s,a)}{\tau}\right)}{\sum_{a'} \exp\left(\frac{Q(s,a')}{\tau}\right)}$

    Tau is the temperature, zero corresponds to greedy, infinity to the random policy

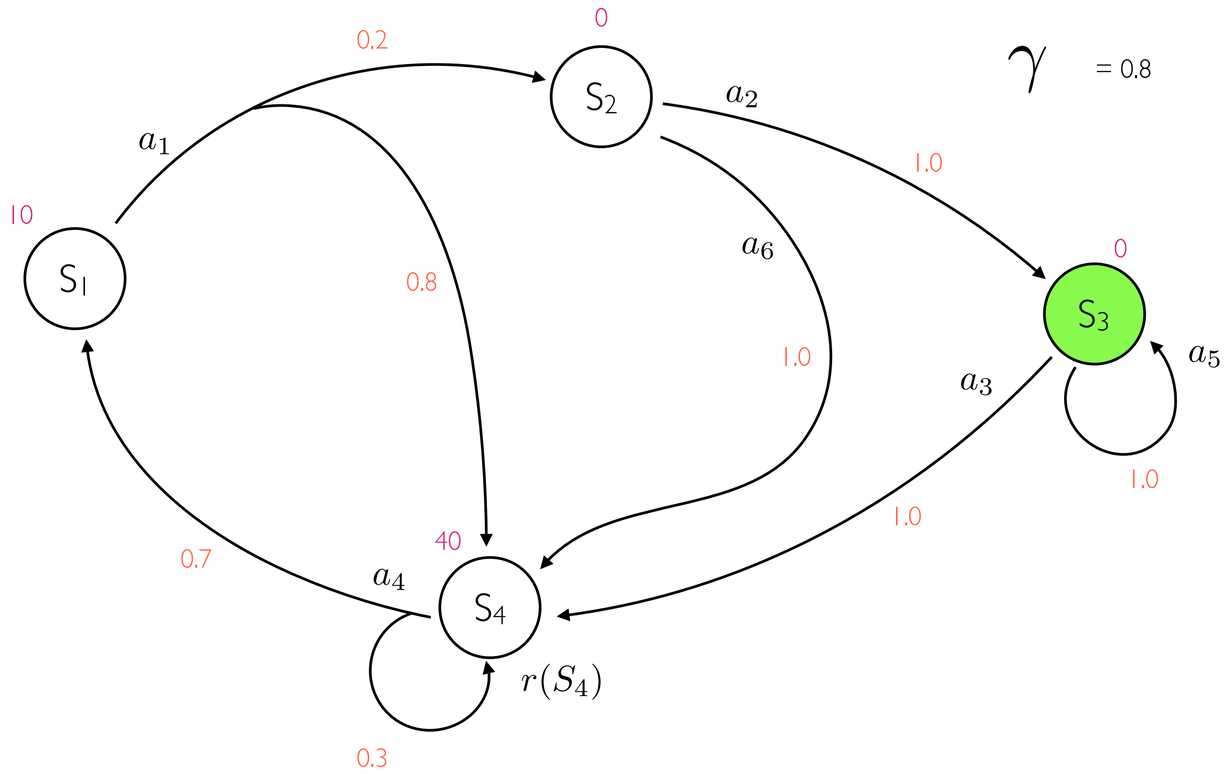
    Usually a schedule is adopted to reduce tau as the time increases

MDP



Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	25					
	S2		30				22
	S3			10		13	
	S4				21		

MDP

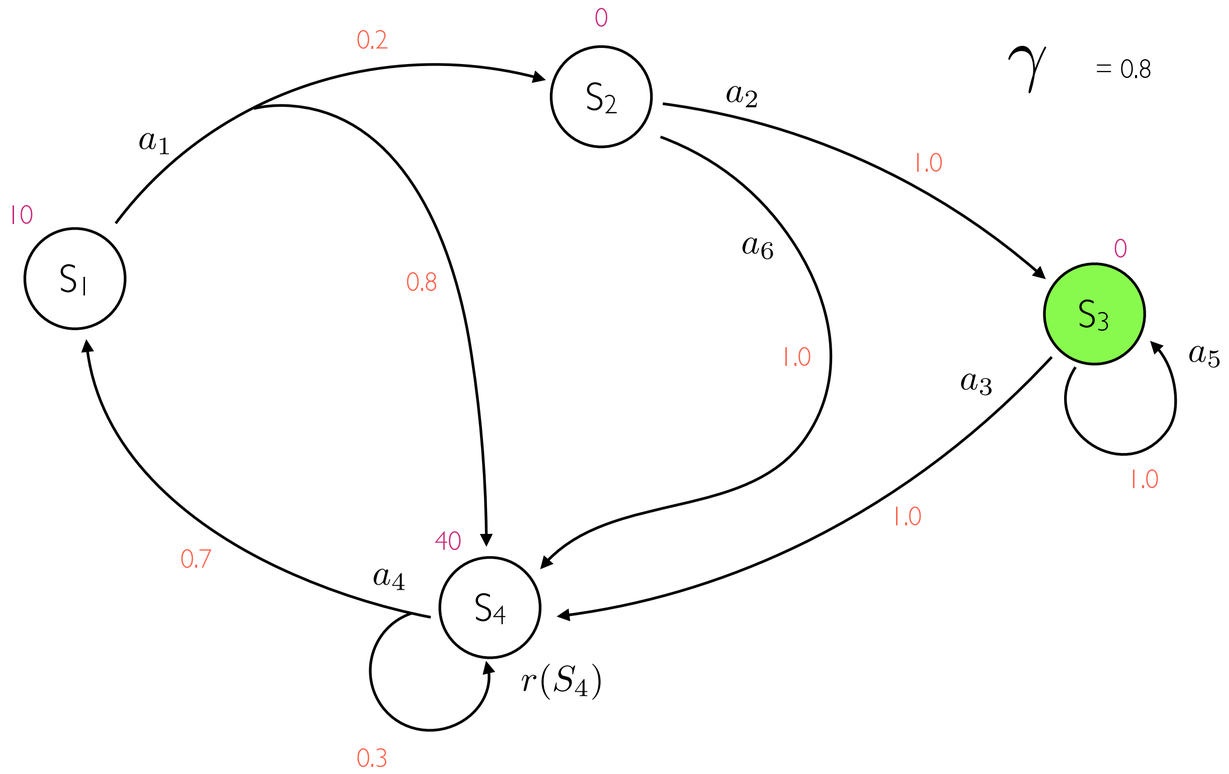


γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	25					
	S2		30				22
	S3			10		13	
	S4				21		

Greedy $\begin{cases} a_3 & 0.0 \\ a_5 & 1.0 \end{cases}$

MDP



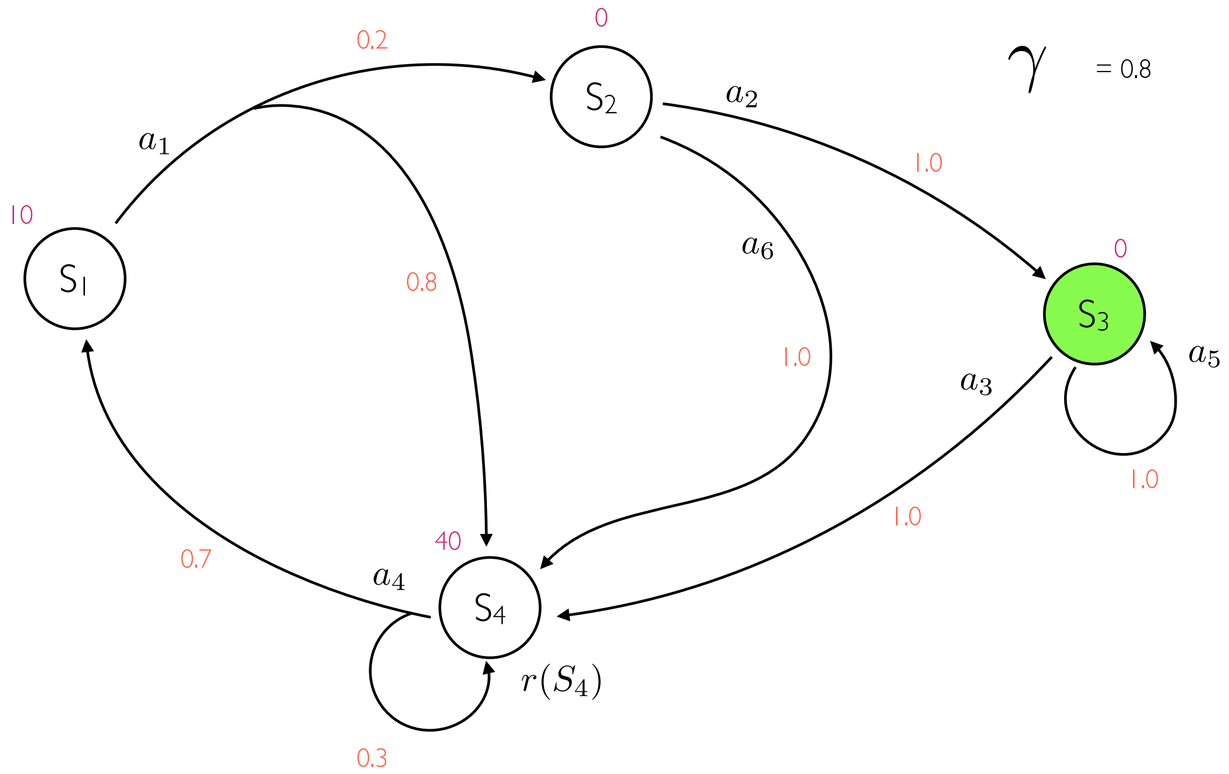
γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	25					
	S2		30				22
	S3			10		13	
	S4				21		

Greedy $\begin{cases} a_3 & 0.0 \\ a_5 & 1.0 \end{cases}$

Epsilon-Greedy $\begin{cases} a_3 & \epsilon \\ a_5 & 1.0 - \epsilon \end{cases}$

MDP



γ = 0.8

Q-Table		Actions					
		a1	a2	a3	a4	a5	a6
States	S1	25					
	S2		30				22
	S3			10		13	
	S4				21		

Greedy $\begin{cases} a_3 & 0.0 \\ a_5 & 1.0 \end{cases}$

Epsilon-Greedy $\begin{cases} a_3 & \epsilon \\ a_5 & 1.0 - \epsilon \end{cases}$

Boltzmann $\begin{cases} a_3 & \frac{\exp(10/\tau)}{\exp(10/\tau)+\exp(13/\tau)} \\ a_5 & \frac{\exp(13/\tau)}{\exp(10/\tau)+\exp(13/\tau)} \end{cases}$

Convergence (1)

- Q-leaning converges to the optimal policy when the environment is stationary and the policy is sufficiently explorative

Convergence (1)

- Q-leaning converges to the optimal policy when the environment is stationary and the policy is sufficiently explorative
- For instance, when the greedy approach is used, there is no guarantee to converge, the policy not being sufficiently explorative

Convergence (1)

- Q-leaning converges to the optimal policy when the environment is stationary and the policy is sufficiently explorative
- For instance, when the greedy approach is used, there is no guarantee to converge, the policy not being sufficiently explorative
- Tradeoff
  - *Exploitation* to maximize the reward
  - *Exploration* to acquire information

Convergence (2)

- Sufficient conditions guaranteeing the convergence are:

Convergence (2)

- Sufficient conditions guaranteeing the convergence are:
  - The learning rate is such that

$$\alpha = \alpha(t)$$

$$\sum_{t=1}^{\infty} \alpha(t) \rightarrow \infty$$

$$\sum_{t=1}^{\infty} \alpha(t)^2 \rightarrow \text{constant}$$

Convergence (2)

- Sufficient conditions guaranteeing the convergence are:
  - The learning rate is such that

$$\alpha = \alpha(t)$$

$$\sum_{t=1}^{\infty} \alpha(t) \rightarrow \infty$$

$$\sum_{t=1}^{\infty} \alpha(t)^2 \rightarrow \text{constant}$$

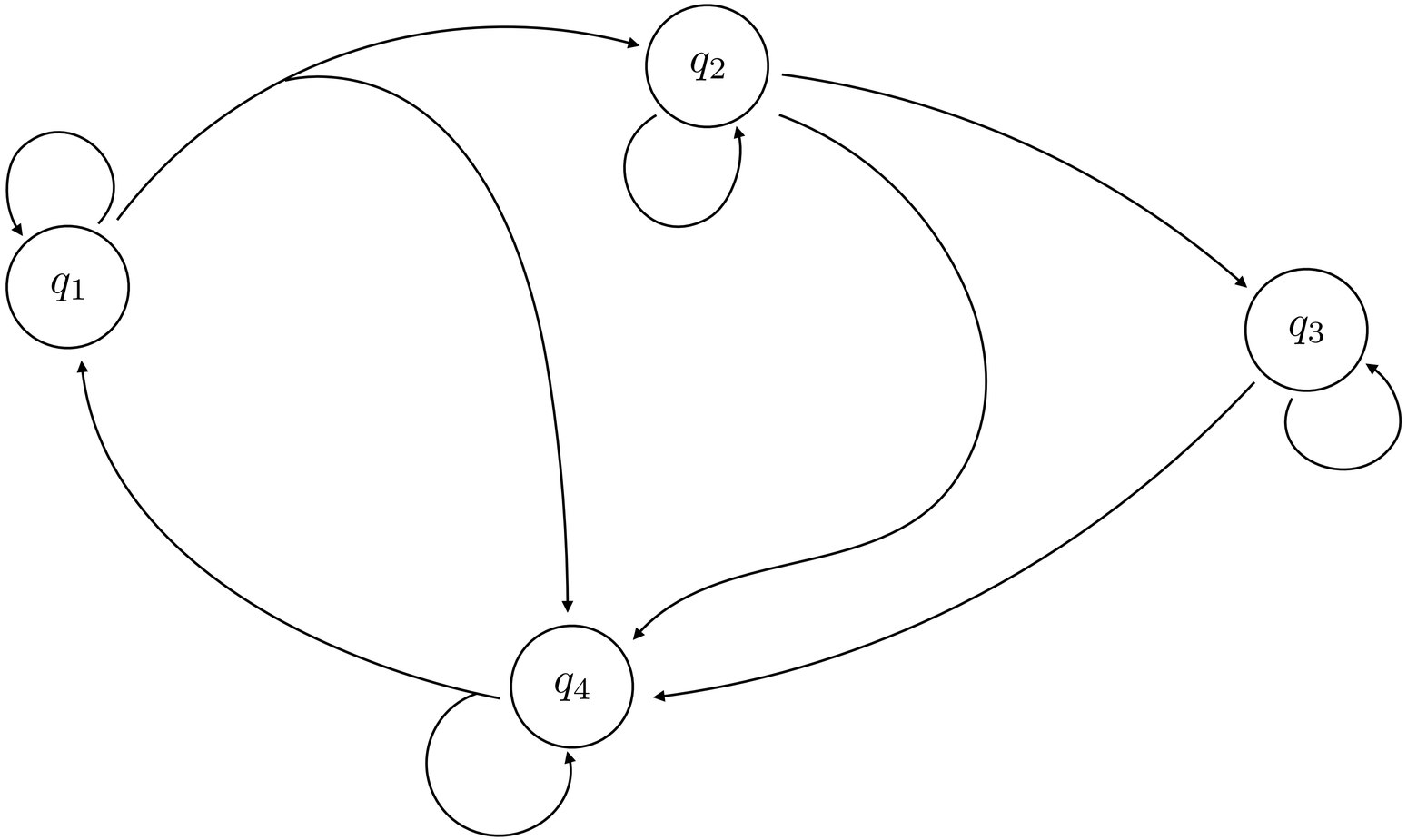
  - The exploration is such that

Epsilon-Greedy	Boltzmann
$\epsilon = \epsilon(t) > 0$	$\tau = \tau(t) > 0$
$\lim_{t \rightarrow \infty} \epsilon(t) = 0$	$\lim_{t \rightarrow \infty} \tau(t) = 0$

Stochastic Games (SG)

From MDP to Stochastic games

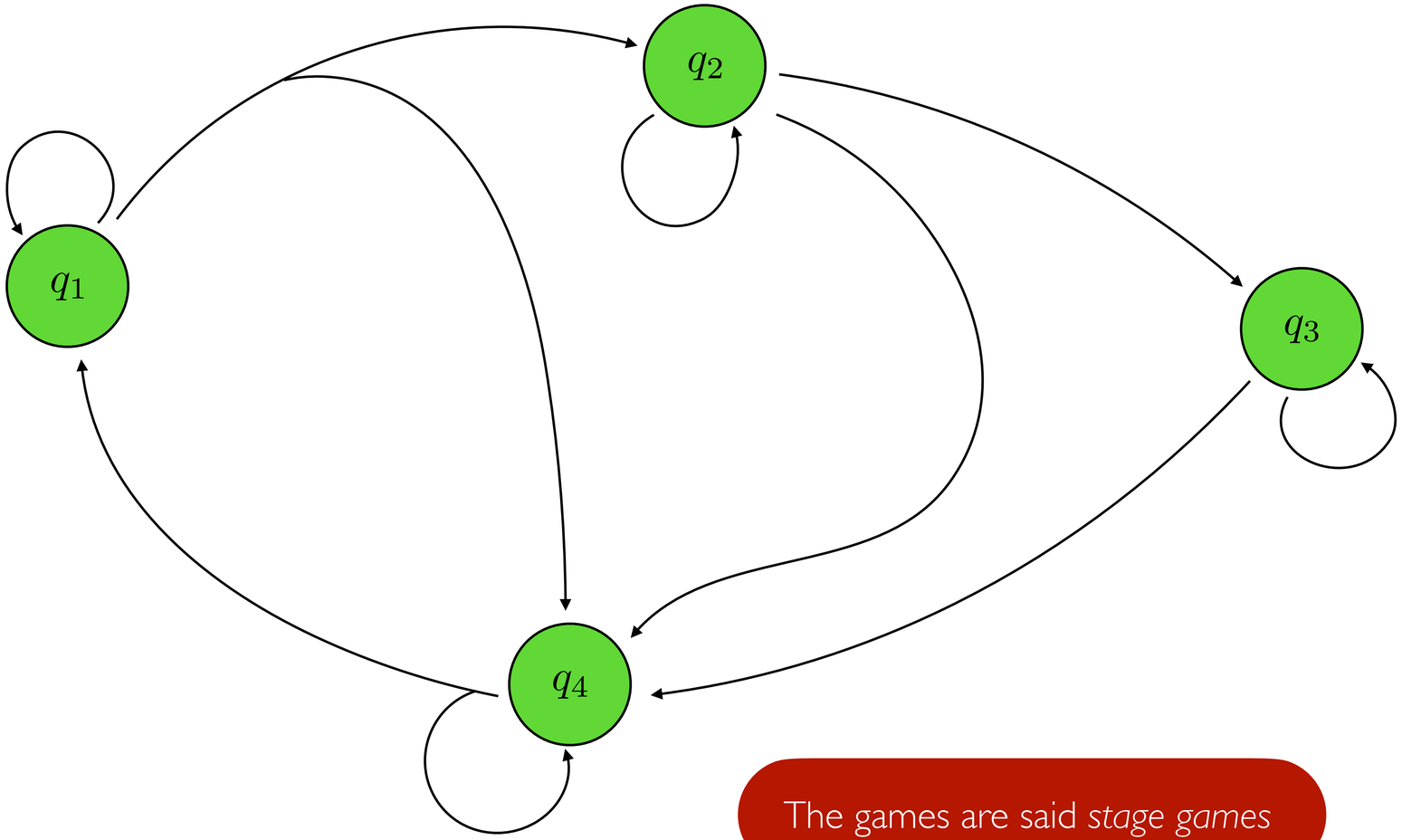
Players 1 and 2



γ

From MDP to Stochastic games

Players 1 and 2



q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

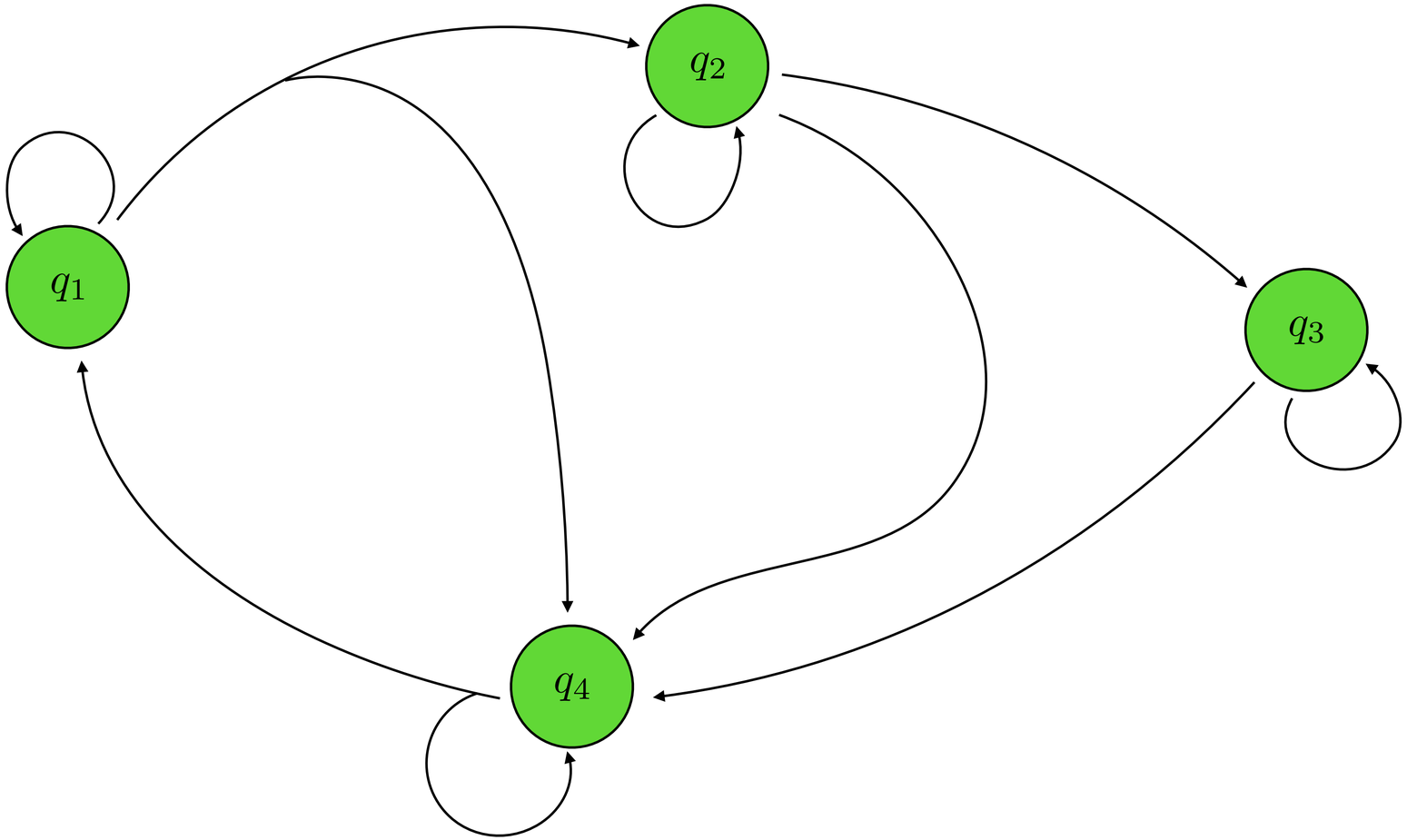
	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

The games are said *stage games*

γ

From MDP to Stochastic games

Players 1 and 2



q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

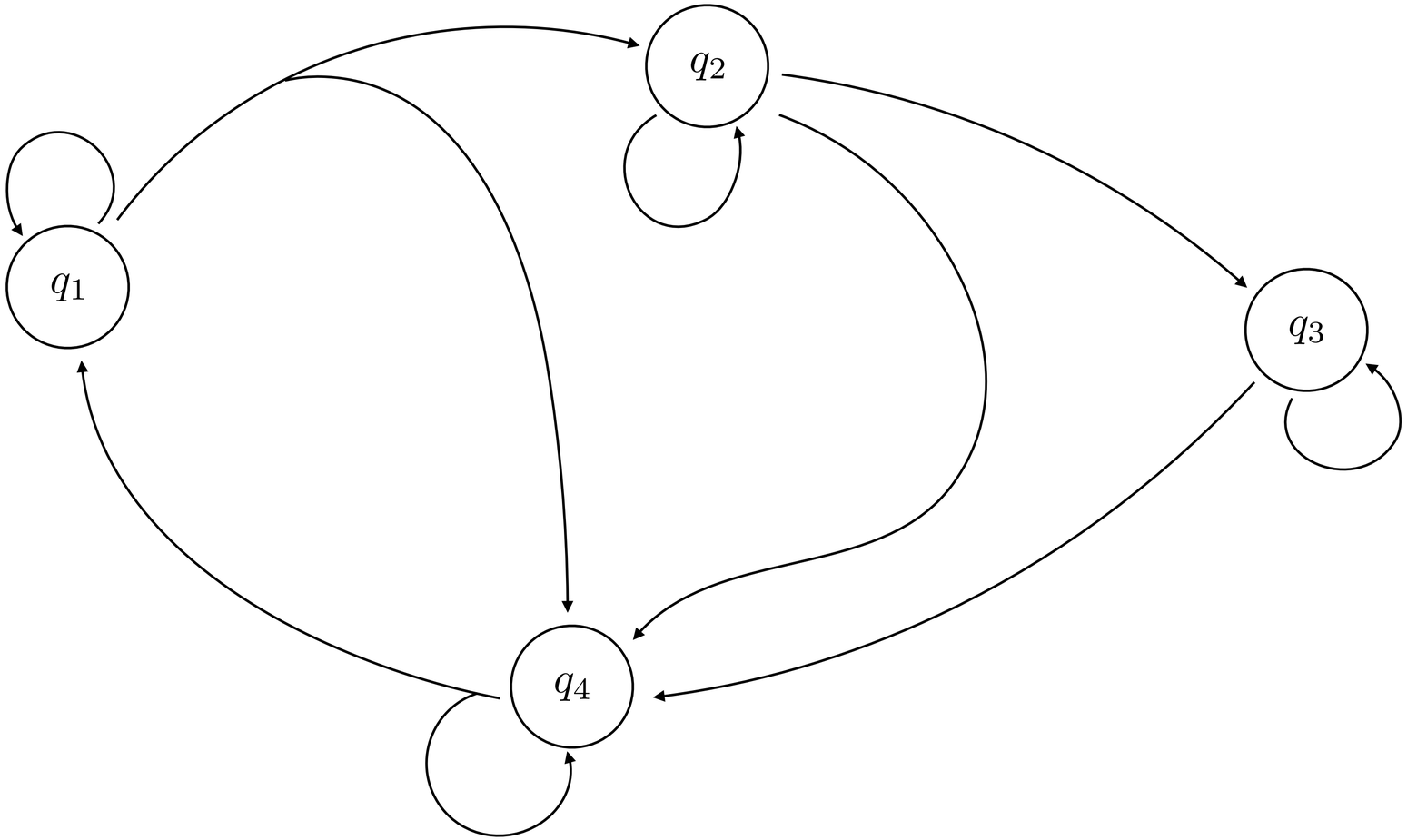
q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

γ

From MDP to Stochastic games

Players 1 and 2



Actions available at game q_1

q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

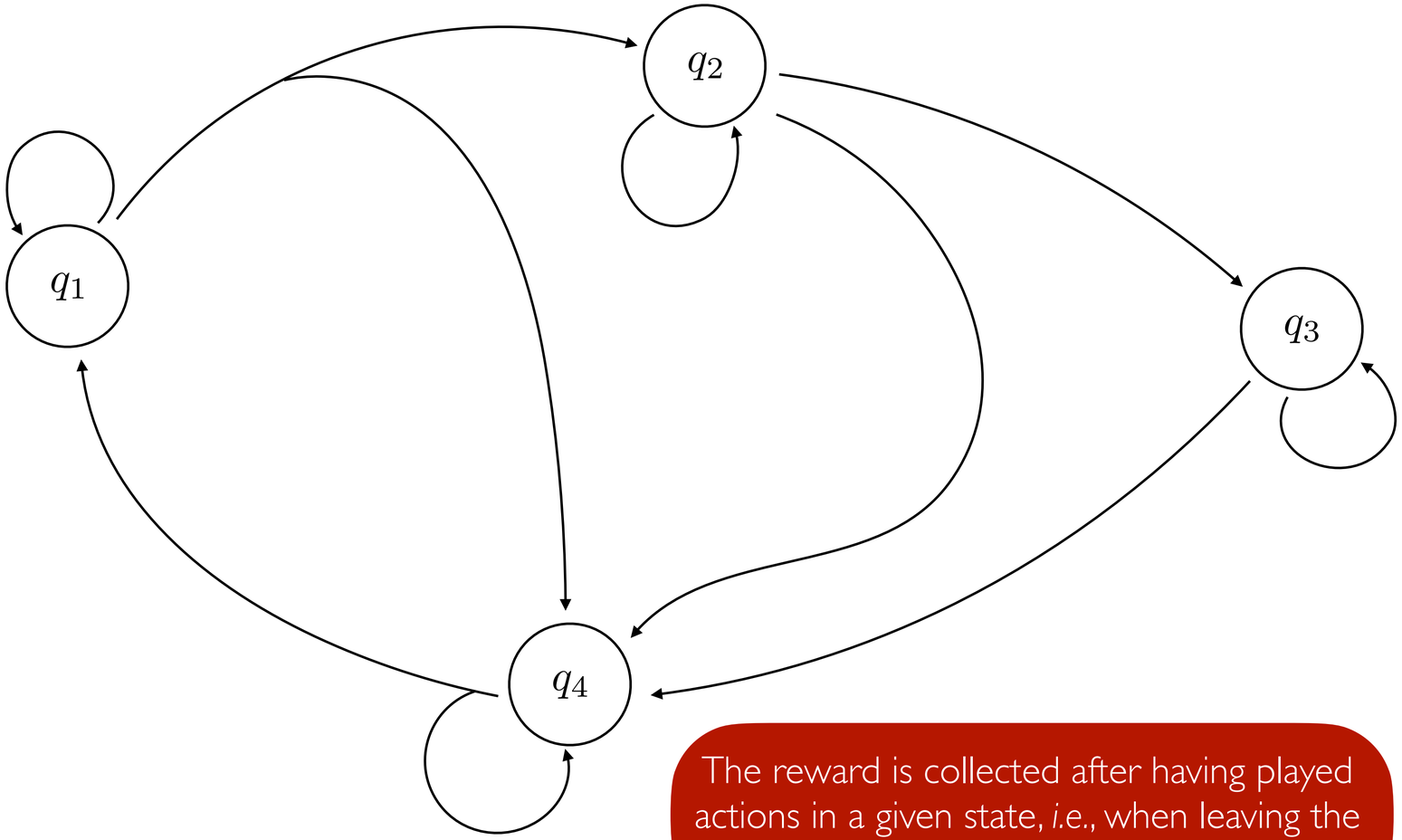
q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

γ

From MDP to Stochastic games

Players 1 and 2



Players' rewards given a pair of actions

q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

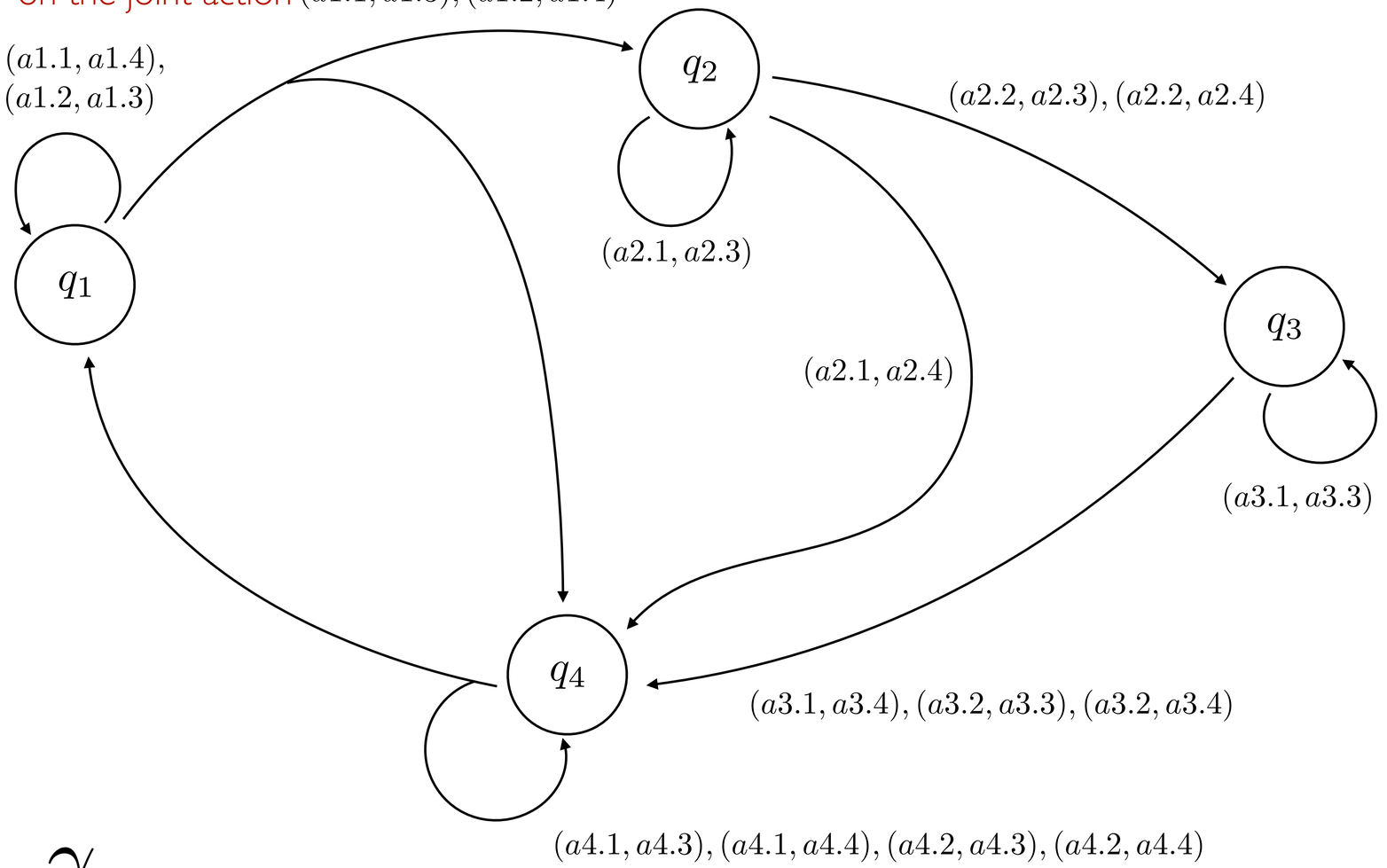
The reward is collected after having played actions in a given state, *i.e.*, when leaving the state, not when entering the state

γ

From MDP to Stochastic games

Players 1 and 2

Transitions defined
on the joint action $(a1.1, a1.3), (a1.2, a1.4)$



q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

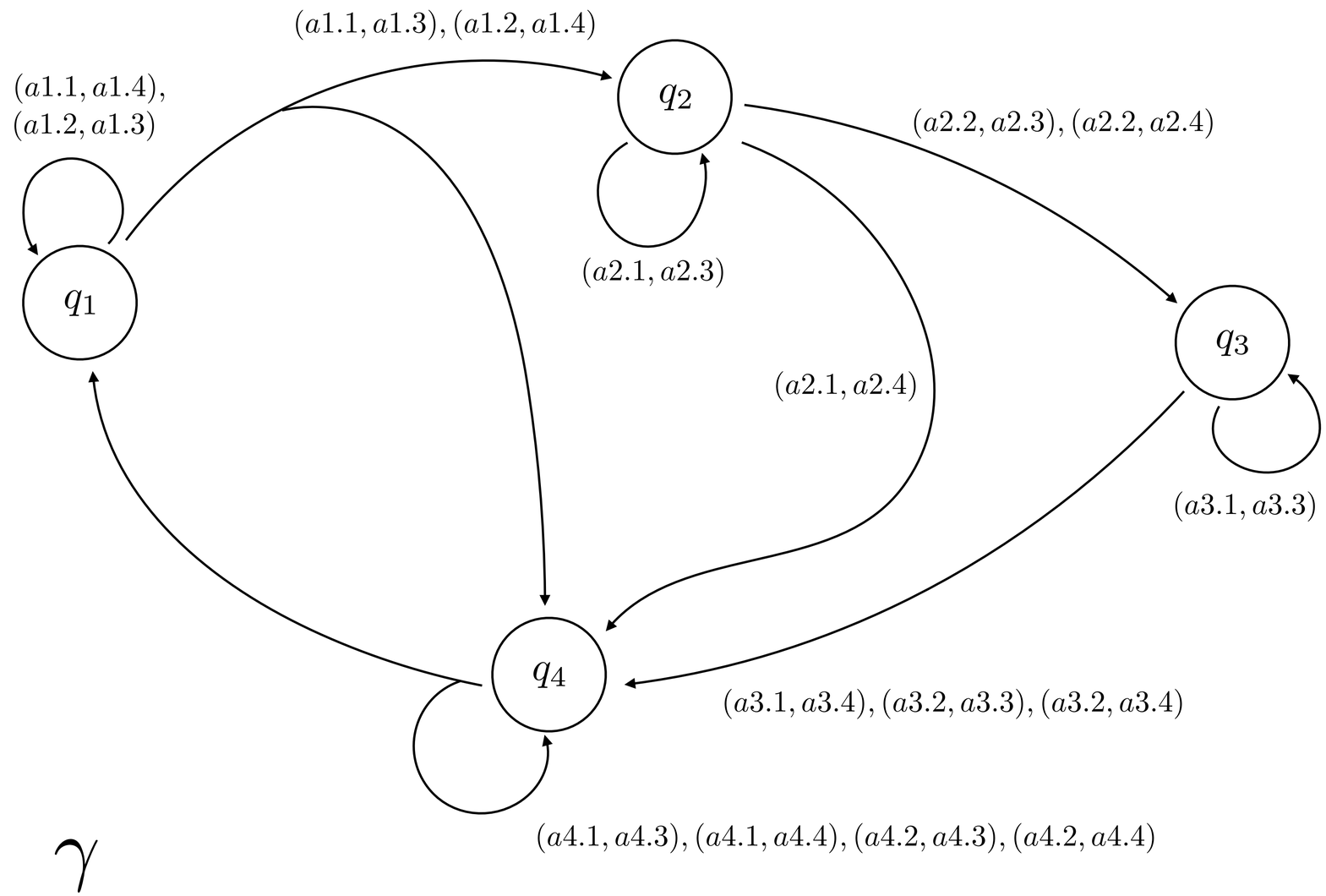
q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

γ

An example

Players 1 and 2



γ

q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

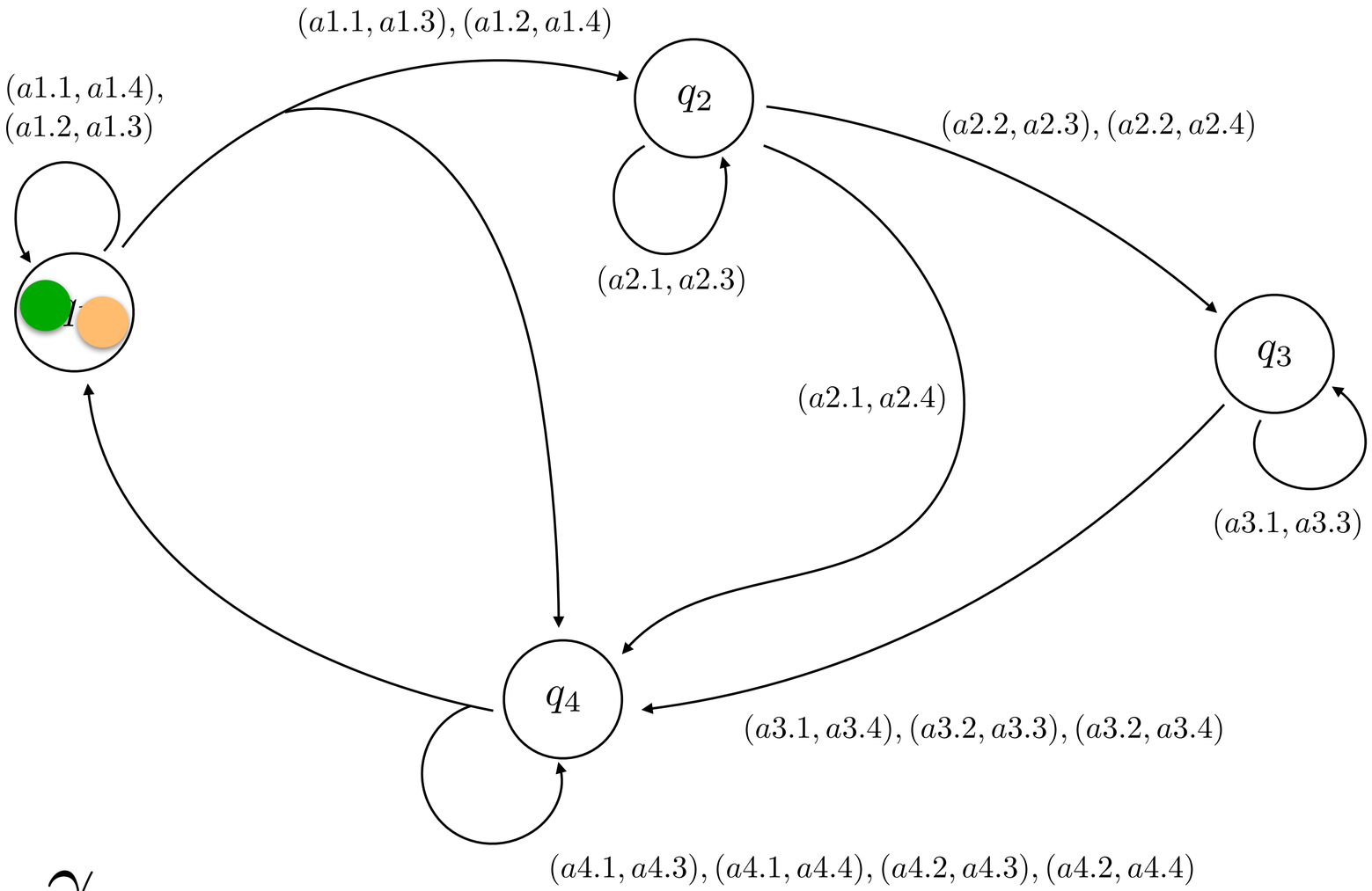
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



γ

q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

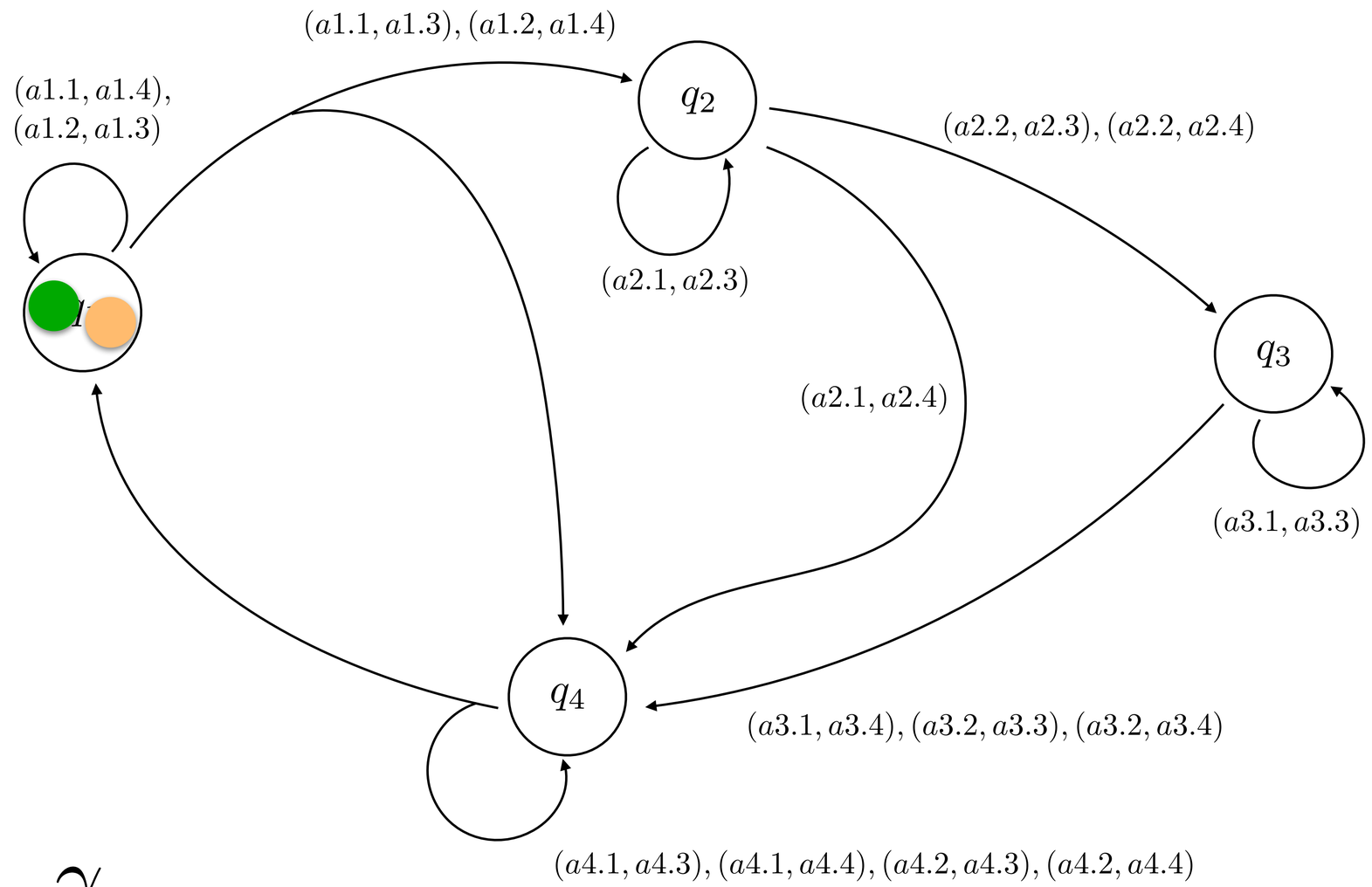
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



γ

q_1

	a1.2	
	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

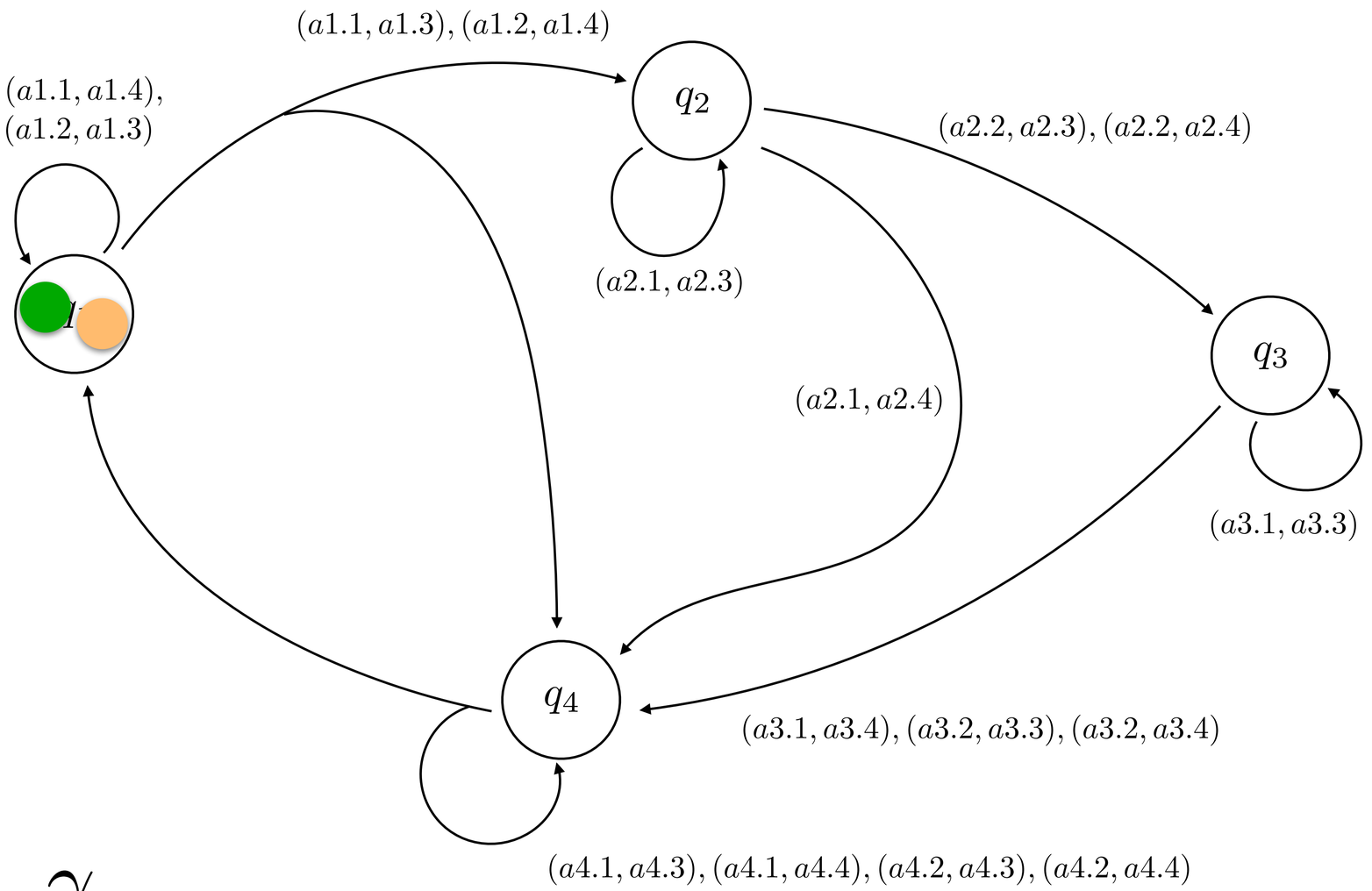
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

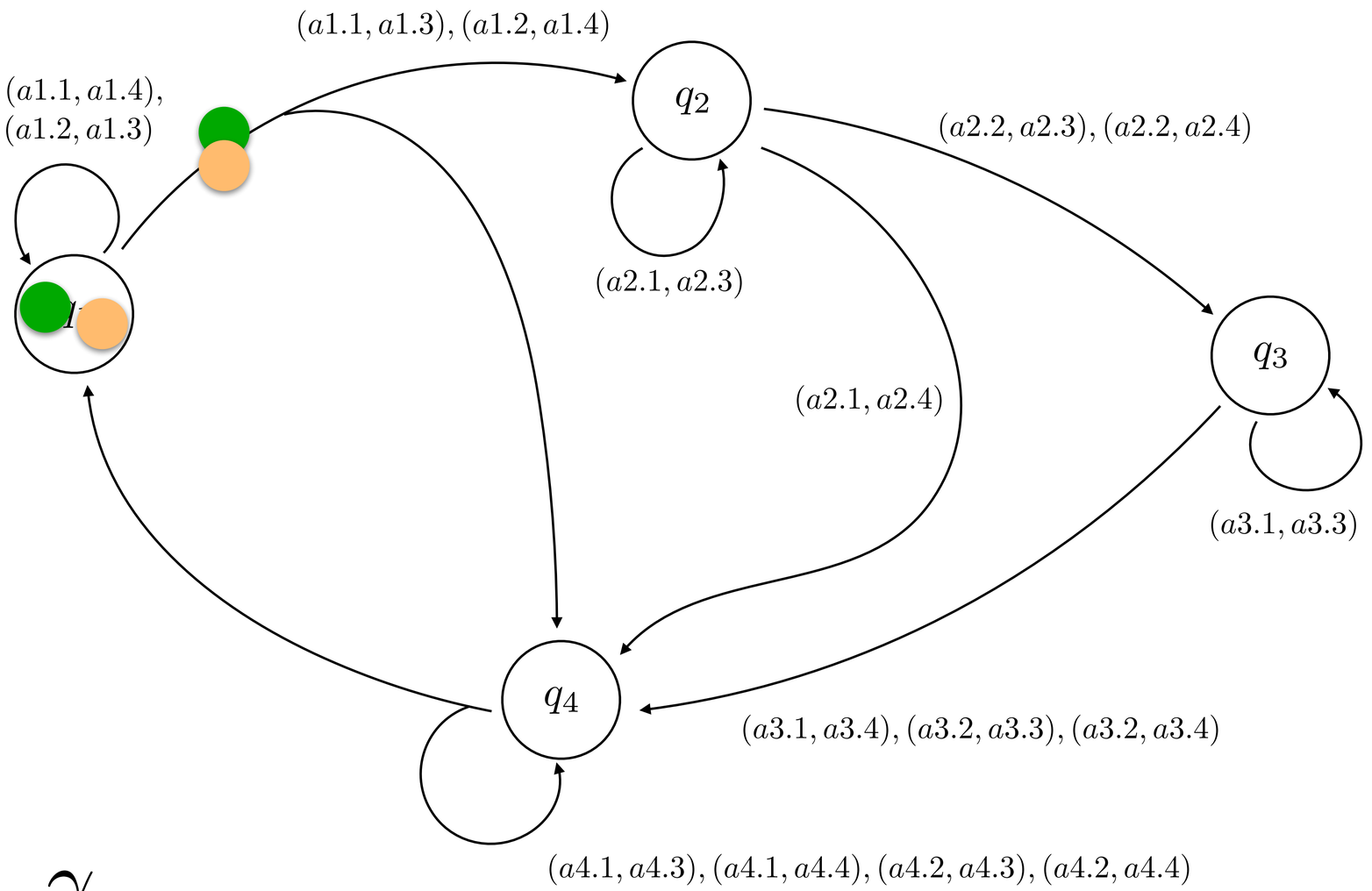
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



γ

q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

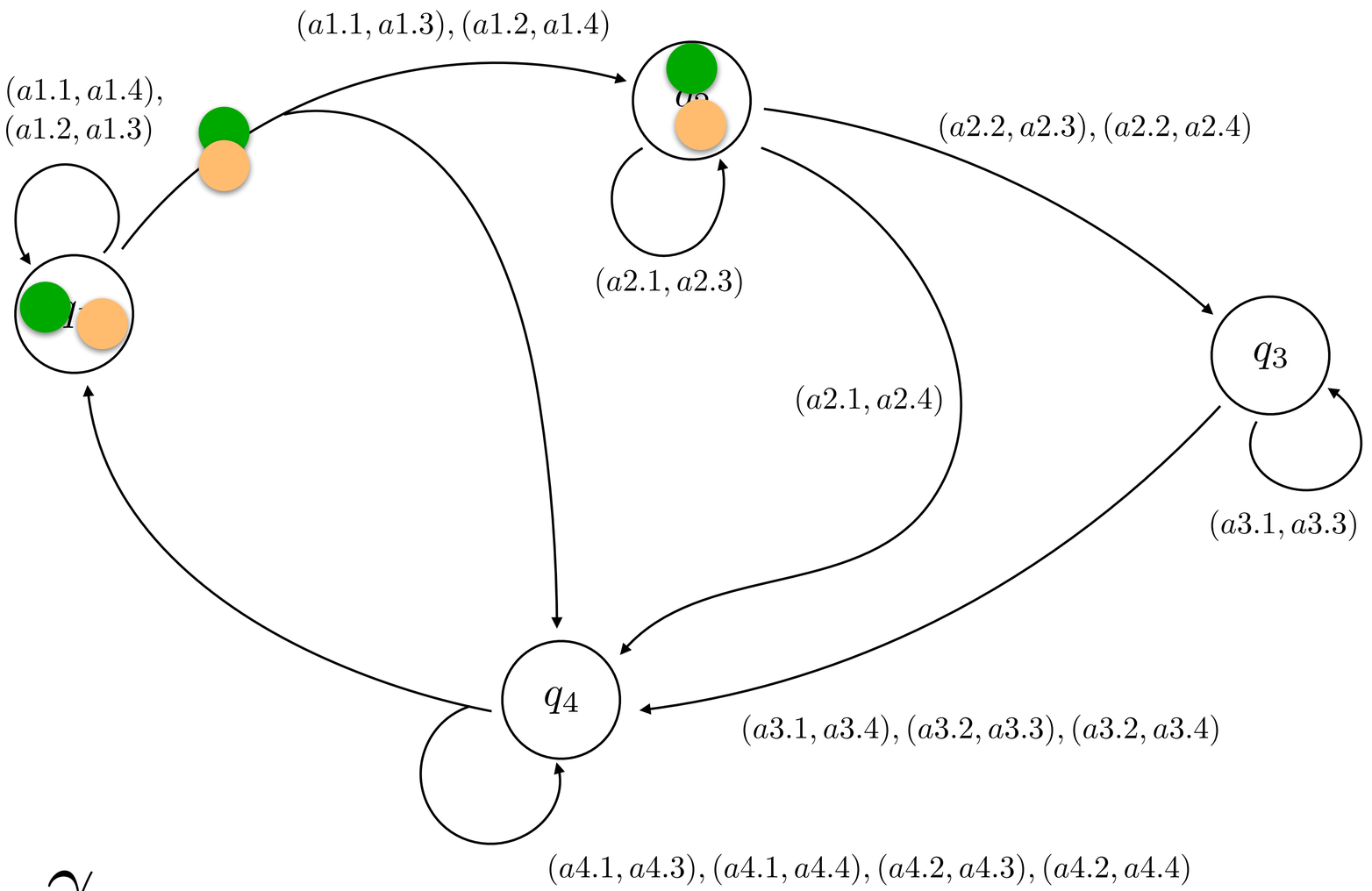
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



γ

q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

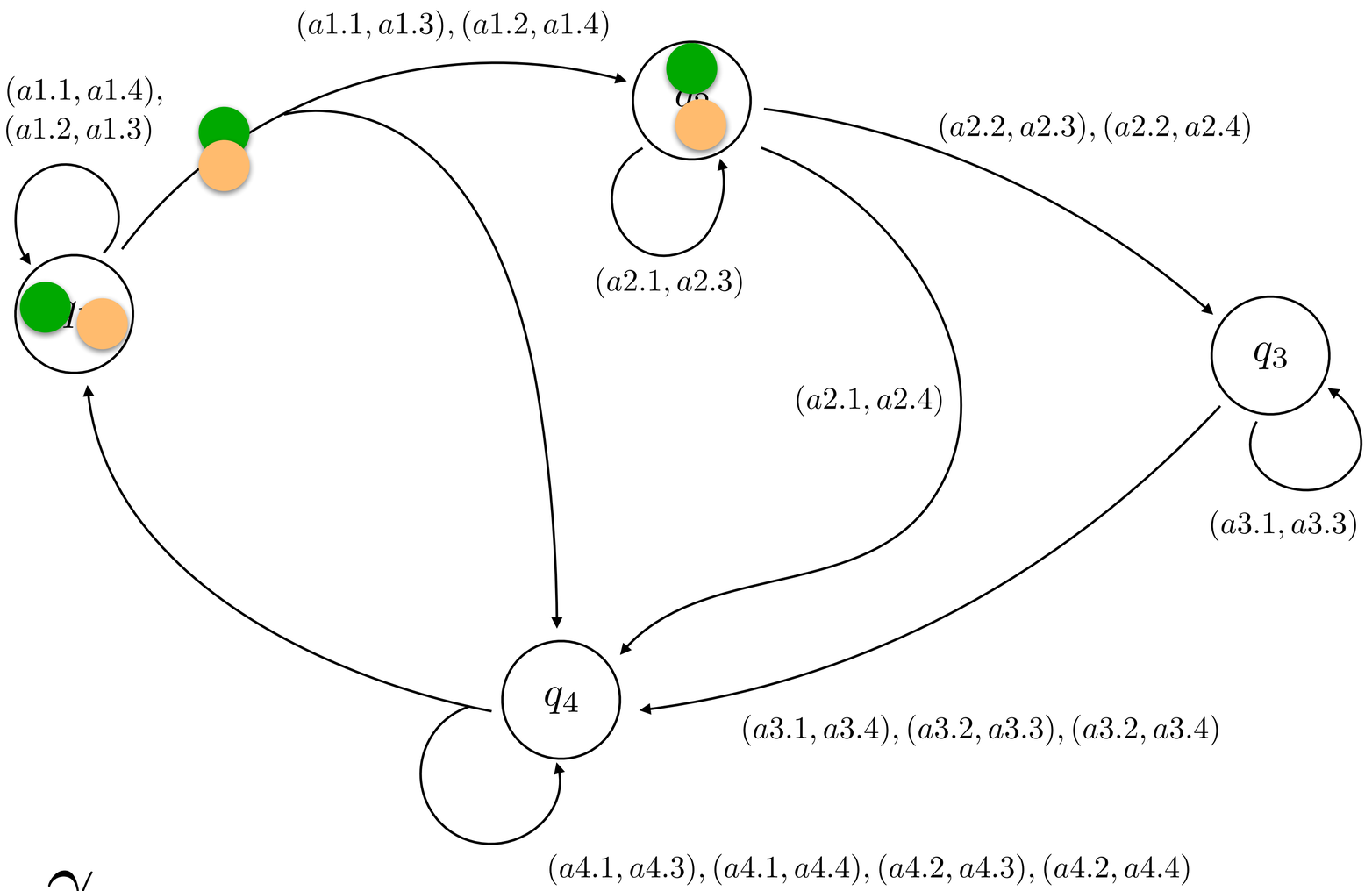
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



γ

q_1

	a1.2	
	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.2	
a2.3	1, 1	3, 0
	0, 3	2, 2

q_3

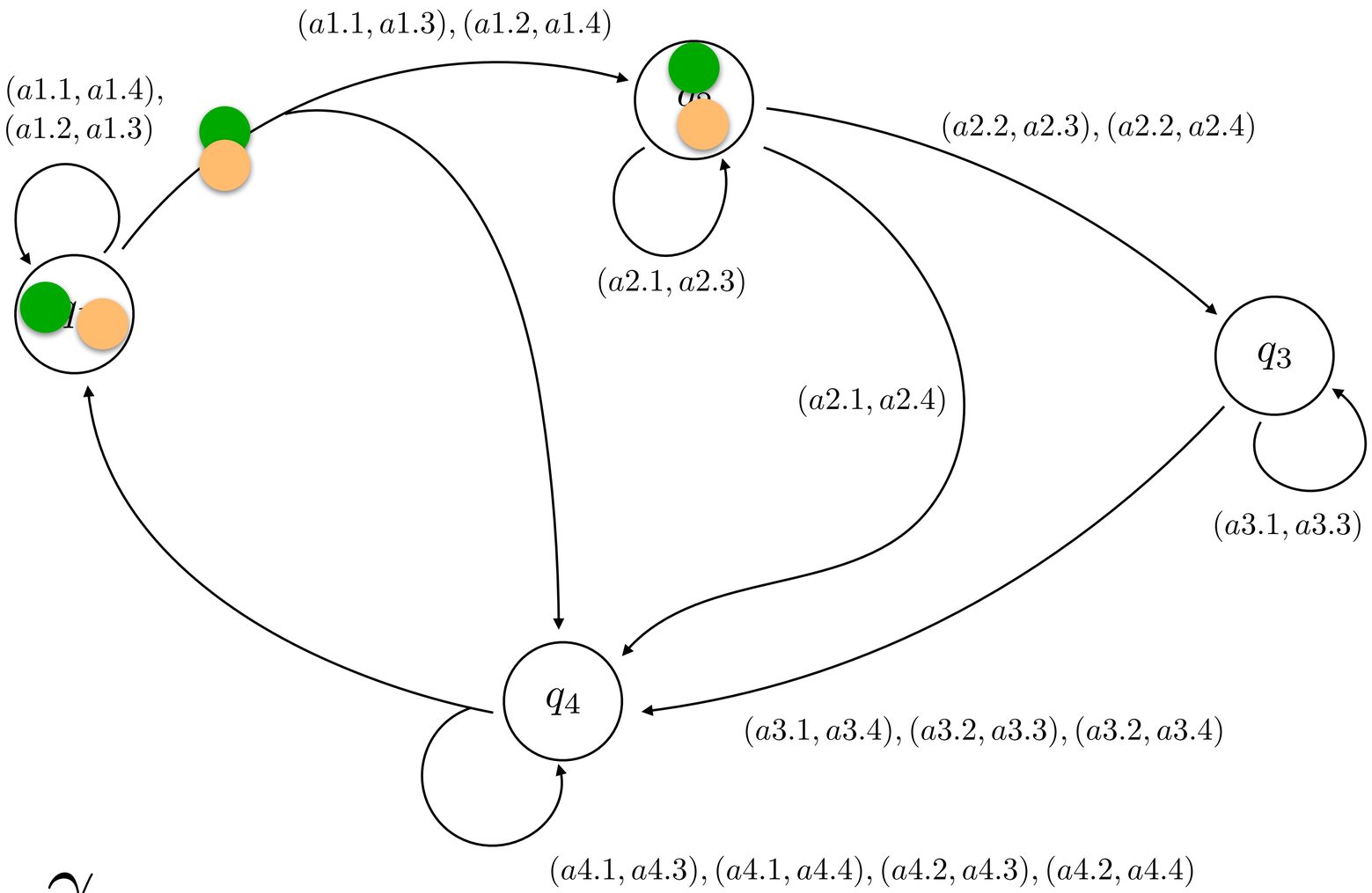
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



γ

q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

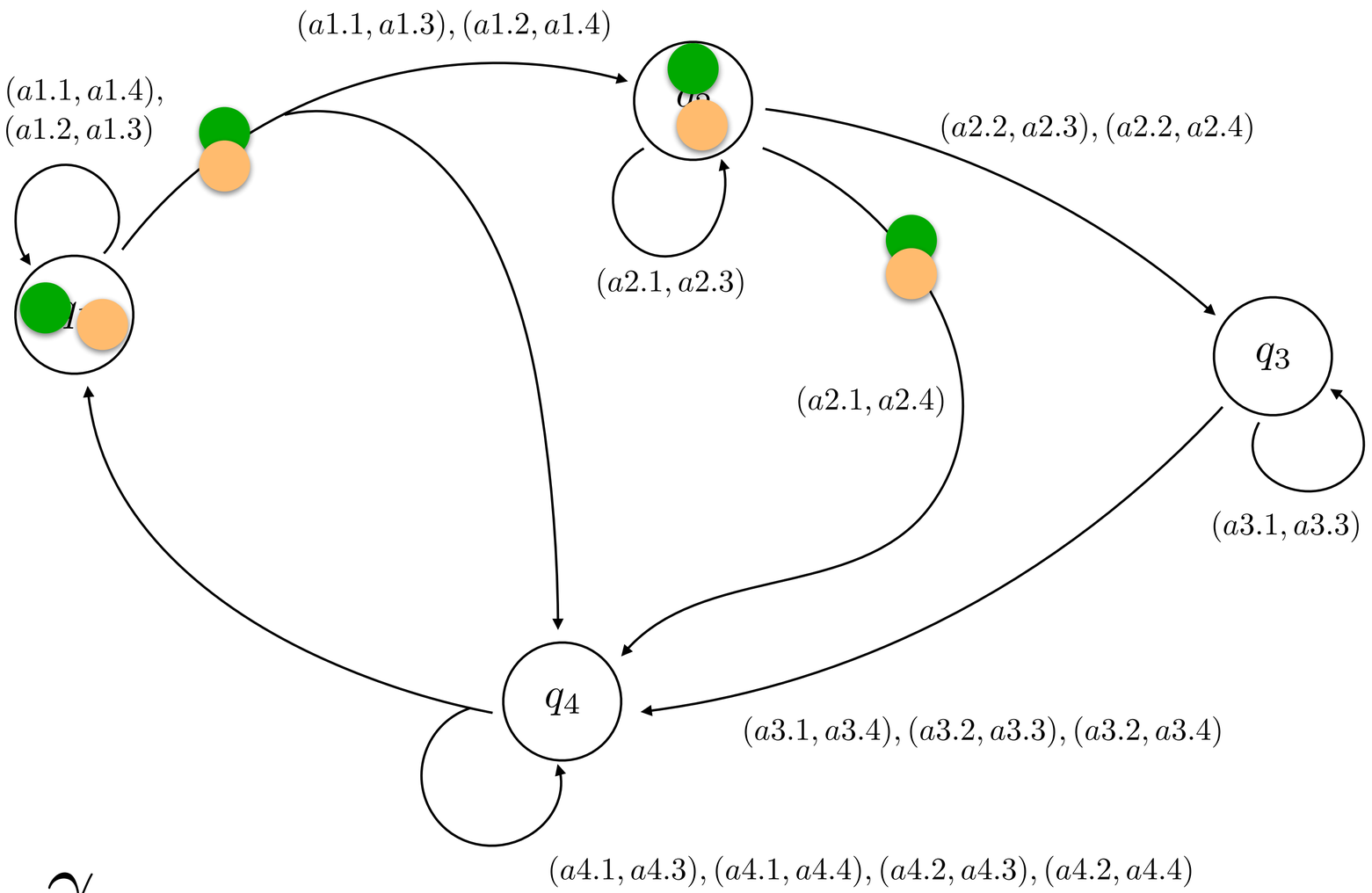
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



γ

q_1

		a1.2
	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

		a2.2
a2.3	1, 1	3, 0
	0, 3	2, 2

q_3

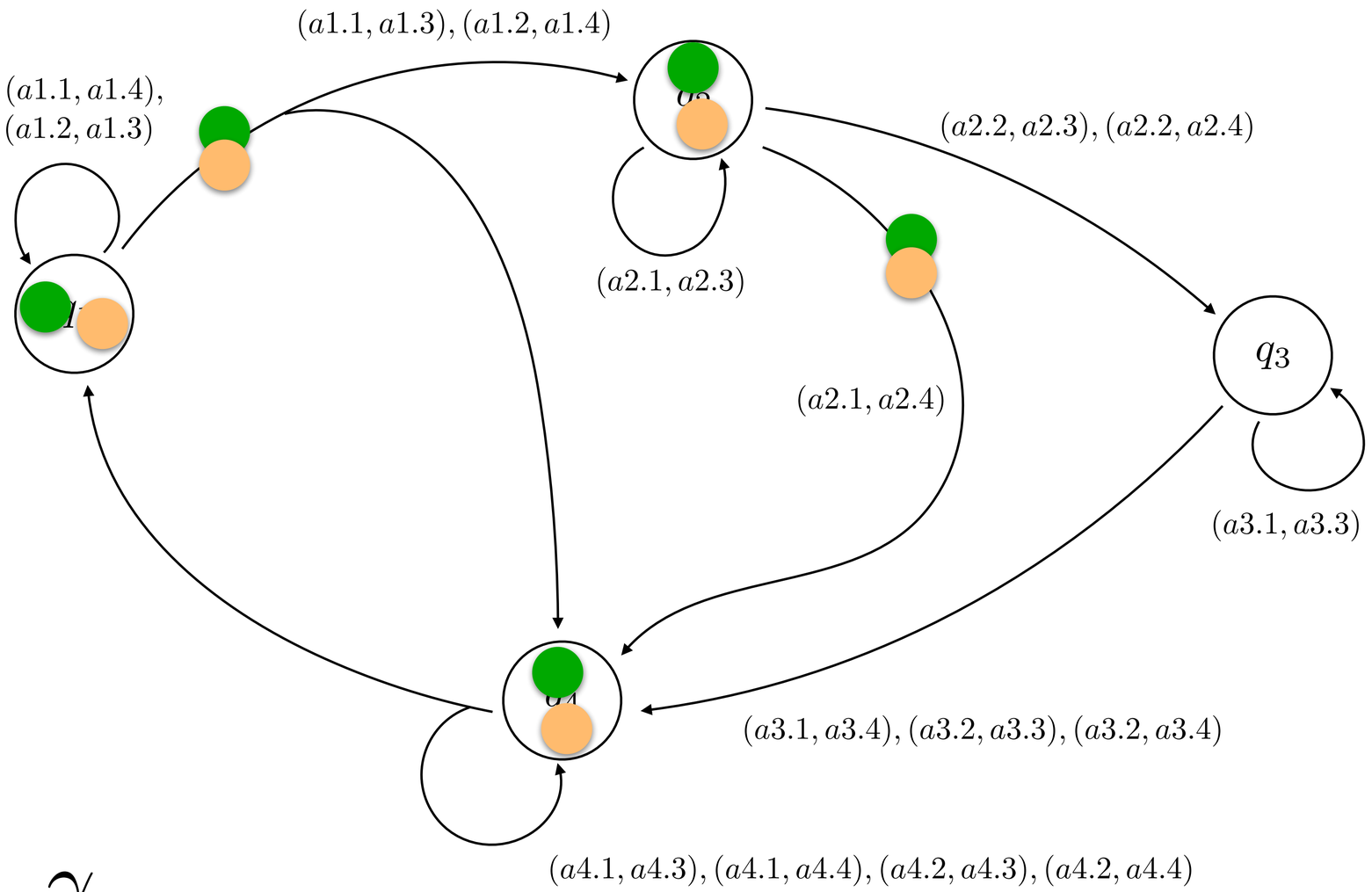
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

An example

Players 1 and 2



q_1

	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

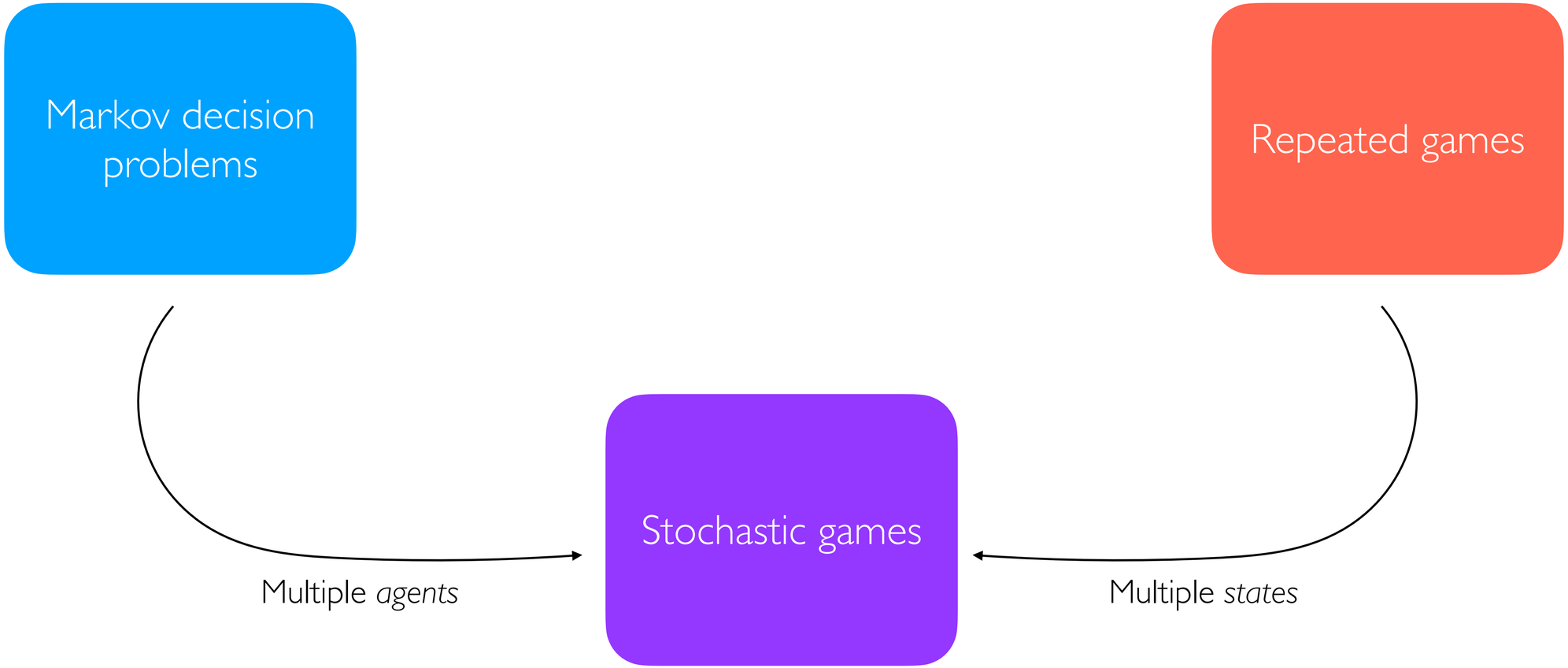
q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

Formal model

- A stochastic game (also known as a Markov game) is a tuple **(Q, N, A, P, r)**, where:
  - **Q** is a finite set of games
  - **N** is a finite set of n players
  - $\mathbf{A} = \mathbf{A}_1 \times \ldots \times \mathbf{A}_n$, where $\mathbf{A}_i$ is a finite set of actions available to player ***i***
  - $\mathbf{P} : \mathbf{Q} \times \mathbf{A} \times \mathbf{Q} \rightarrow [0, 1]$ is the transition probability function; $\mathbf{P}(\mathbf{q}, \mathbf{a}, \mathbf{q}')$ is the probability of transitioning from game **q** to state **q'** after action profile **a**
  - $\mathbf{R} = \mathbf{r}_1, \ldots, \mathbf{r}_n$, where $\mathbf{r}_i : \mathbf{Q} \times \mathbf{A} \rightarrow \mathbf{R}$ is a real-valued payoff function for player **i**

Generalization



Markov Nash equilibrium

- A strategy profile is a Nash equilibrium of the Stochastic game if, for every state, following the strategy profile is a best response to the opponents' strategies for every player

Markov Nash equilibrium

- A strategy profile is a Nash equilibrium of the Stochastic game if, for every state, following the strategy profile is a best response to the opponents' strategies for every player

    - A Nash equilibrium of the Stochastic game, may prescribe, at every state, a strategy different from the Nash equilibrium of the single game

Markov Nash equilibrium

- A strategy profile is a Nash equilibrium of the Stochastic game if, for every state, following the strategy profile is a best response to the opponents' strategies for every player
    - A Nash equilibrium of the Stochastic game, may prescribe, at every state, a strategy different from the Nash equilibrium of the single game
- Every *n*-player discounted stochastic game possesses at least one Nash equilibrium point in stationary strategies

Markov Nash equilibrium

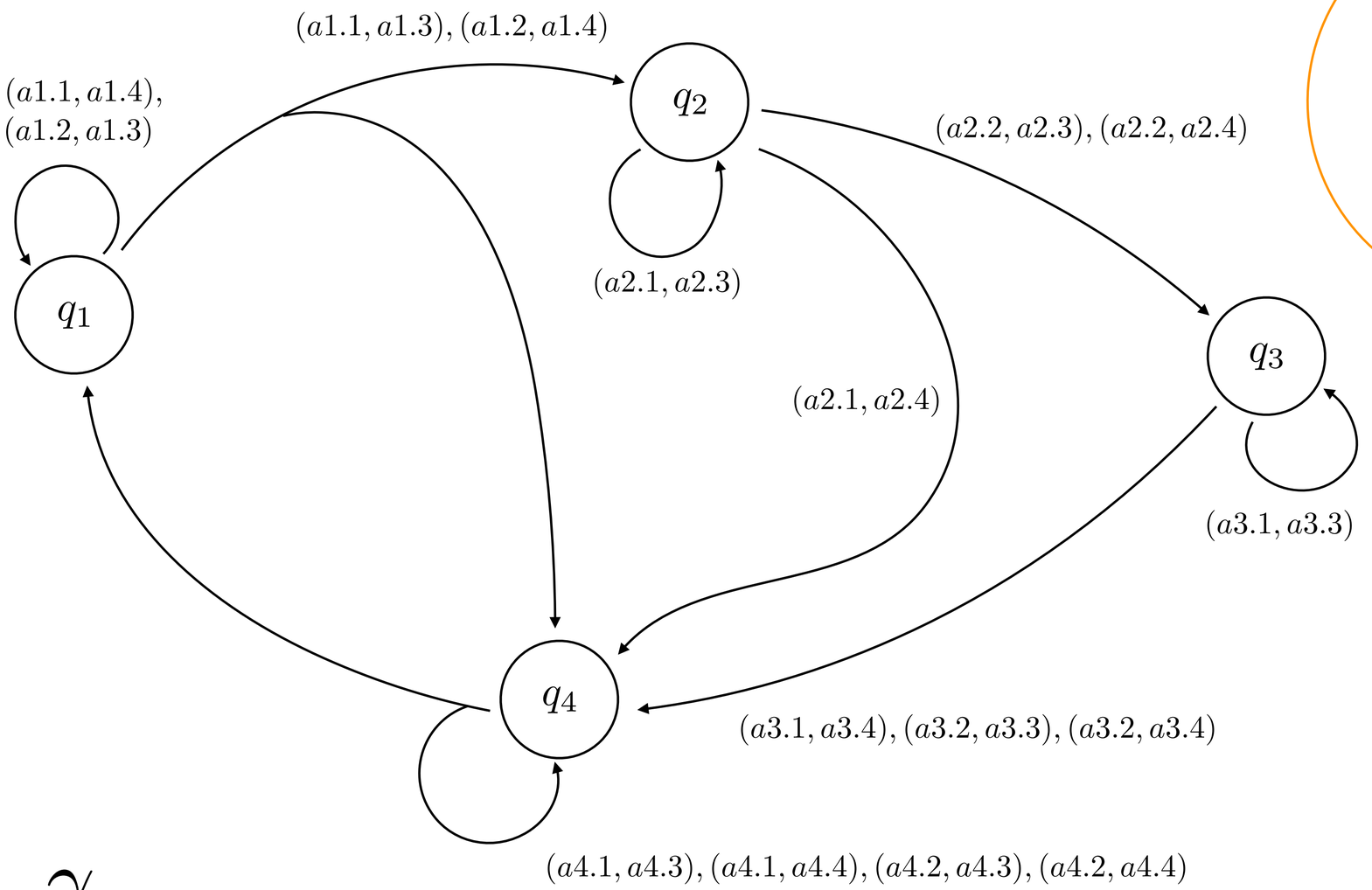
- A strategy profile is a Nash equilibrium of the Stochastic game if, for every state, following the strategy profile is a best response to the opponents' strategies for every player
    - A Nash equilibrium of the Stochastic game, may prescribe, at every state, a strategy different from the Nash equilibrium of the single game
- Every *n*-player discounted stochastic game possesses at least one Nash equilibrium point in stationary strategies
- Stationary (Markov) strategies do not span the whole space of equilibria (e.g., in repeated games, non-stationary equilibria may exist)

An example

Nash equilibrium of the single-stage game

Nash equilibrium of the Stochastic game

Players 1 and 2



q_1

		a1.2
a1.3	2, 1	0, 0
4	0, 0	1, 2

q_2

	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

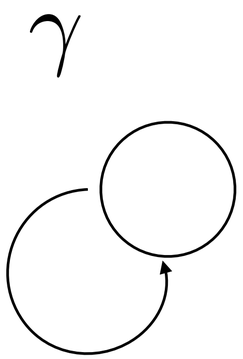
	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

γ

An example (repeated game)



	a2.1	a2.2
a2.3	6, 6	2, 9
a2.4	9, 2	3, 3

Non-stationary strategies are, for instance, to play (a2.2, a2.4) and to switch to play a2.1 or a2.3, respectively, as soon as the opponent deviates from (a2.2, a2.4)

Nash-Q learning

Assumptions

- The players do not know the rewards and the transition probabilities
- In particular,
  - Every player does not know his/her own rewards
  - Every player does not know the rewards of the opponents

Nash-Q learning (1)

- In every state, the Q-table stores the Q-values defined on the joint actions of the players for every player

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3		
a1.4		

q_2

Nash Q values	a2.1	a2.2
a2.3		
a2.4		

q_3

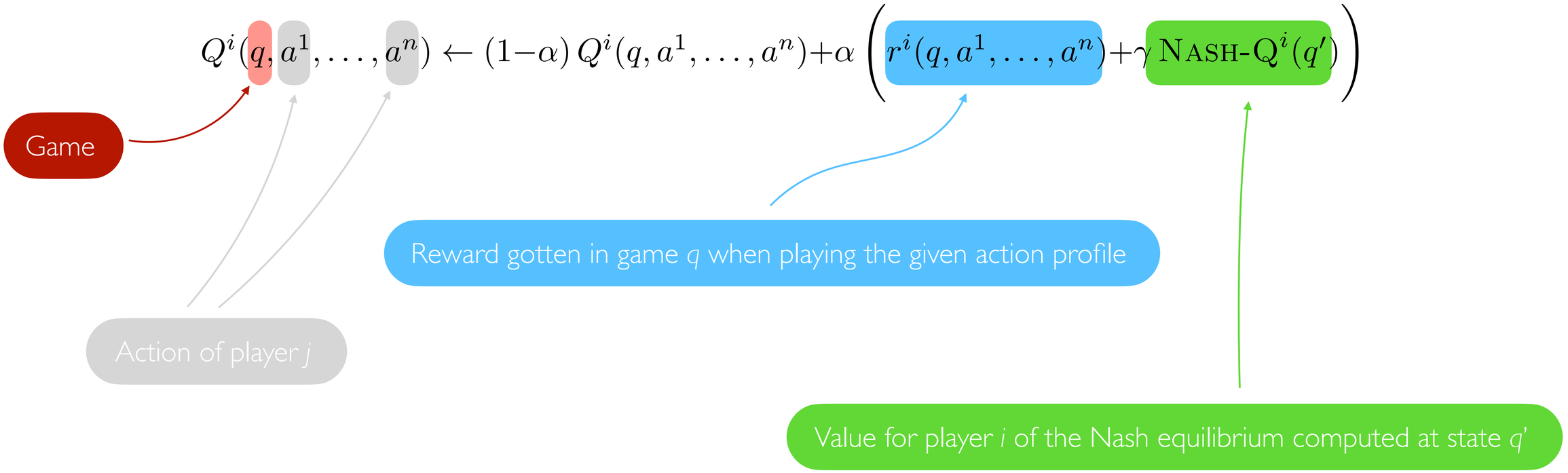
Nash Q values	a3.1	a3.2
a3.3		
a3.4		

q_4

Nash Q values	a4.1	a4.2
a4.3		
a4.4		

Nash-Q learning (2)

- The update of the Nash-Q values of every player is as follows:

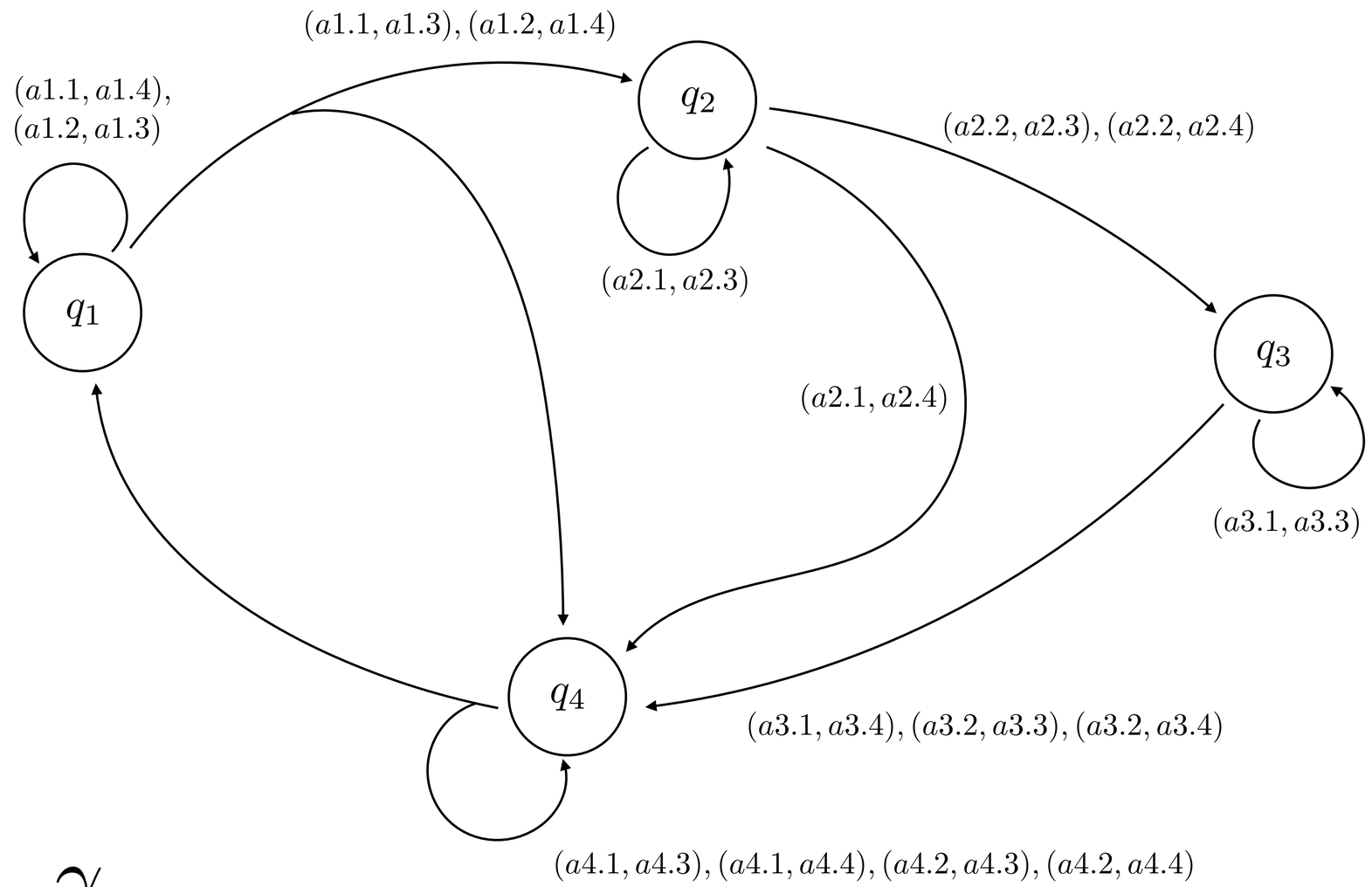


Nash-Q learning (3)

- In every game, the strategies of the players is a Nash equilibrium computed from the Nash-Q values

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	0, 0	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_3

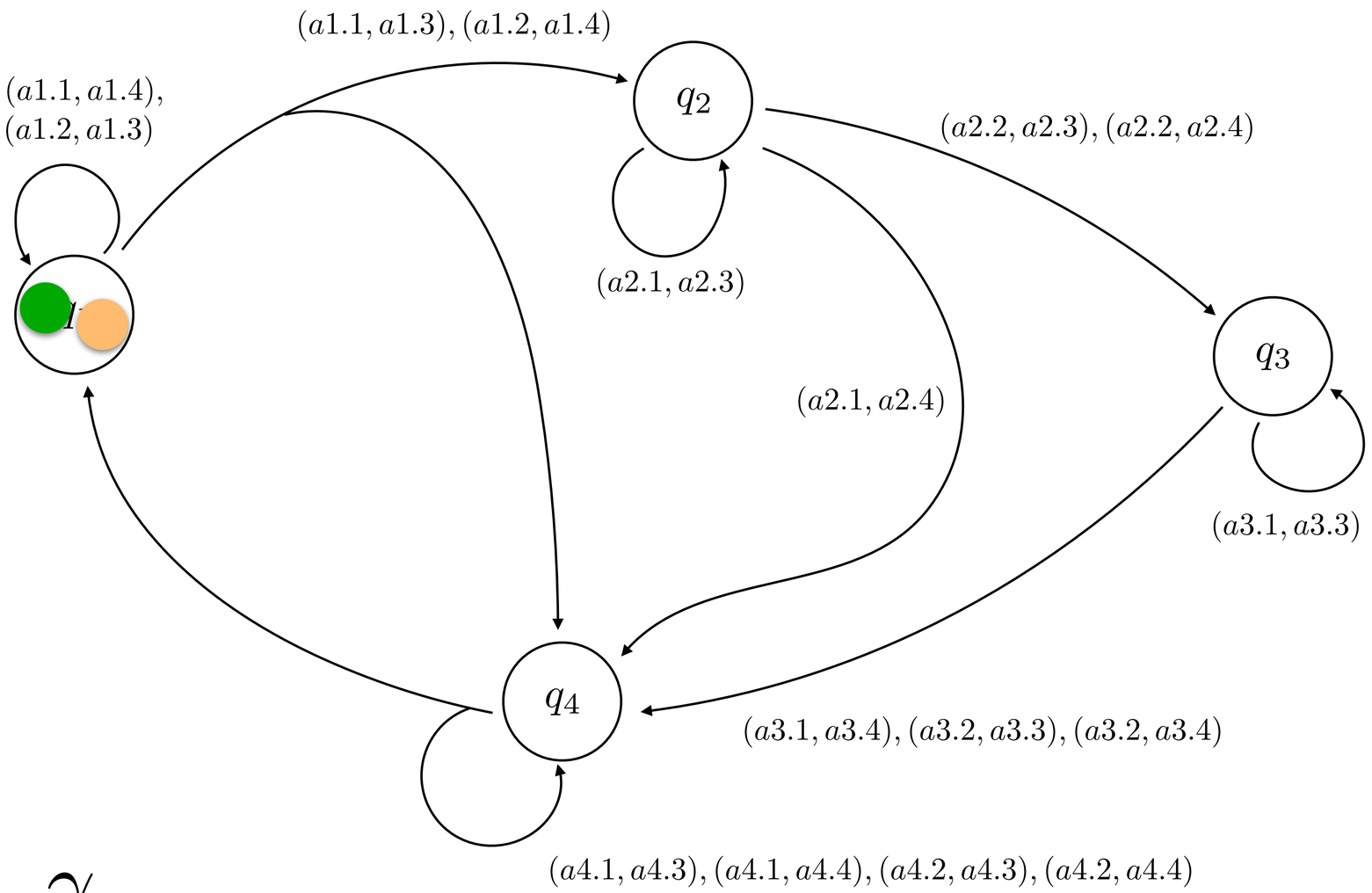
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	0, 0	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_3

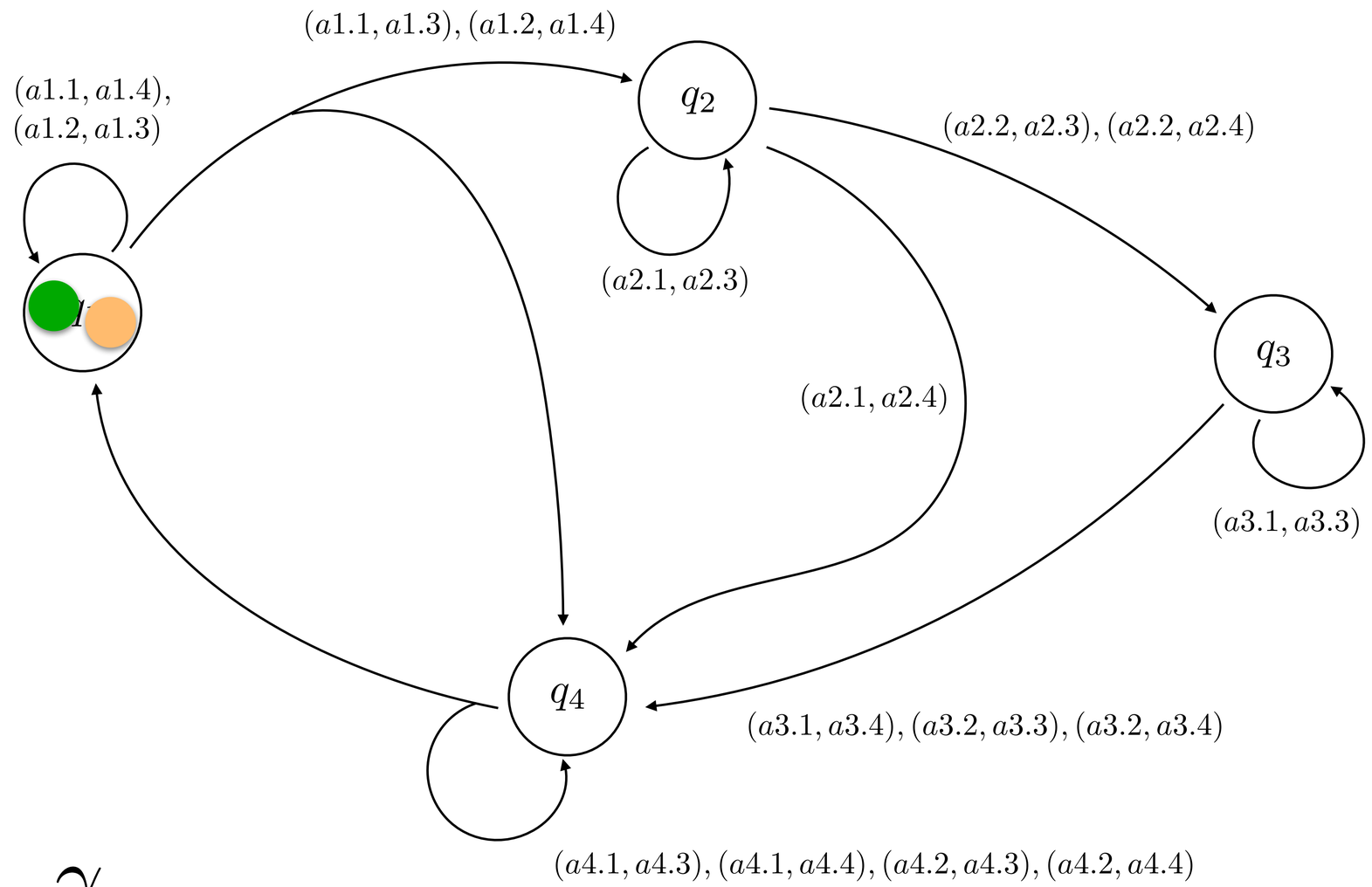
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	0, 0	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_3

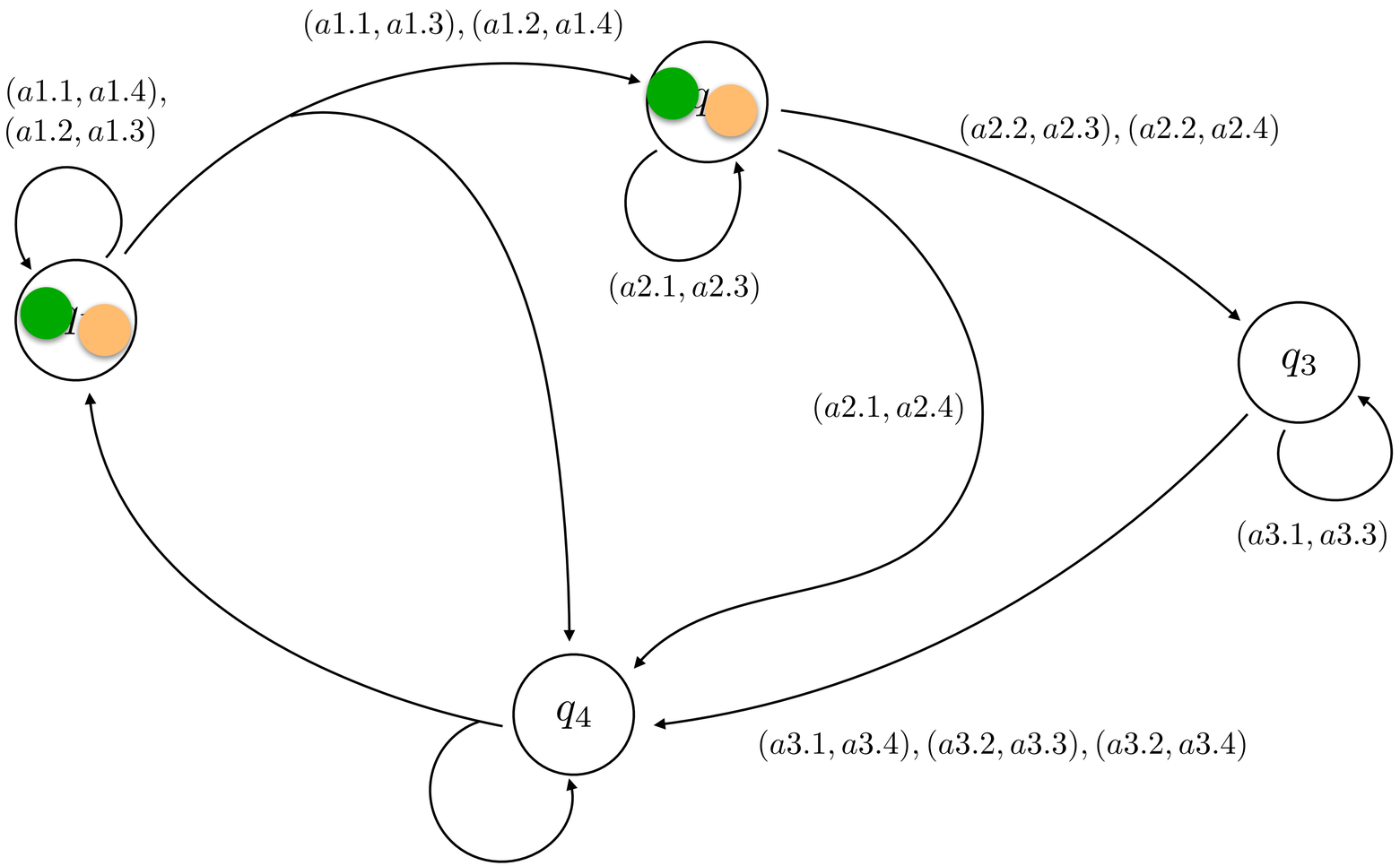
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	0, 0	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_3

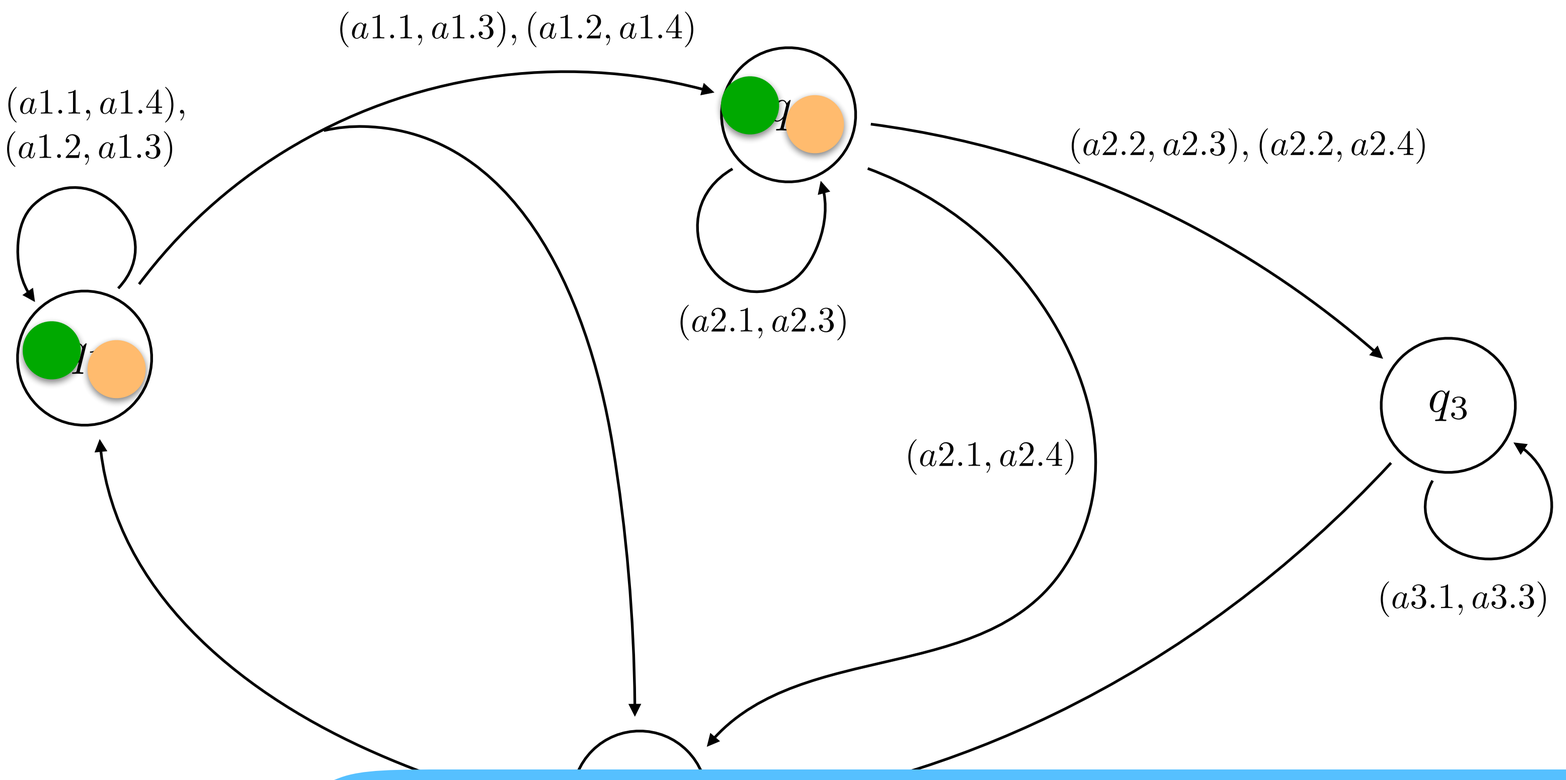
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	0, 0	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_4

	a4.1	a4.2
	0	0, 0
	0, 0	0, 0

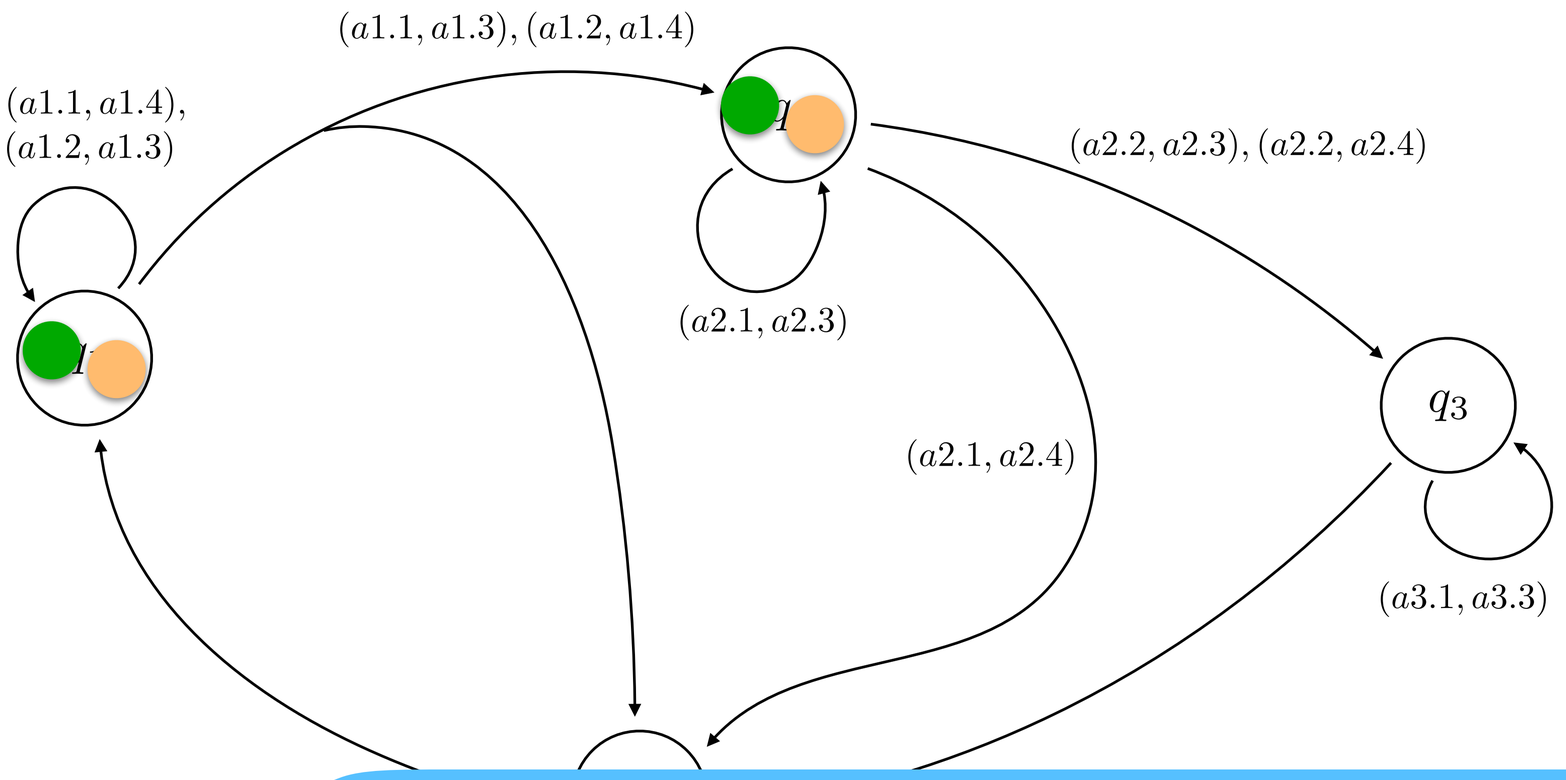
$$Q^i(q, a^1, \ldots, a^n) \leftarrow (1-\alpha)\, Q^i(q, a^1, \ldots, a^n) + \alpha \left(r^i(q, a^1, \ldots, a^n) + \gamma\, \text{NASH-Q}^i(q') \right)$$

(1, .5) 0.5 (0, 0) 0.5 (2, 1) 0.8 (0, 0)

γ

An example

Players 1 and 2



q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	**1, .5**	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_4

	a4.1	a4.2
	0	0, 0
	0, 0	0, 0

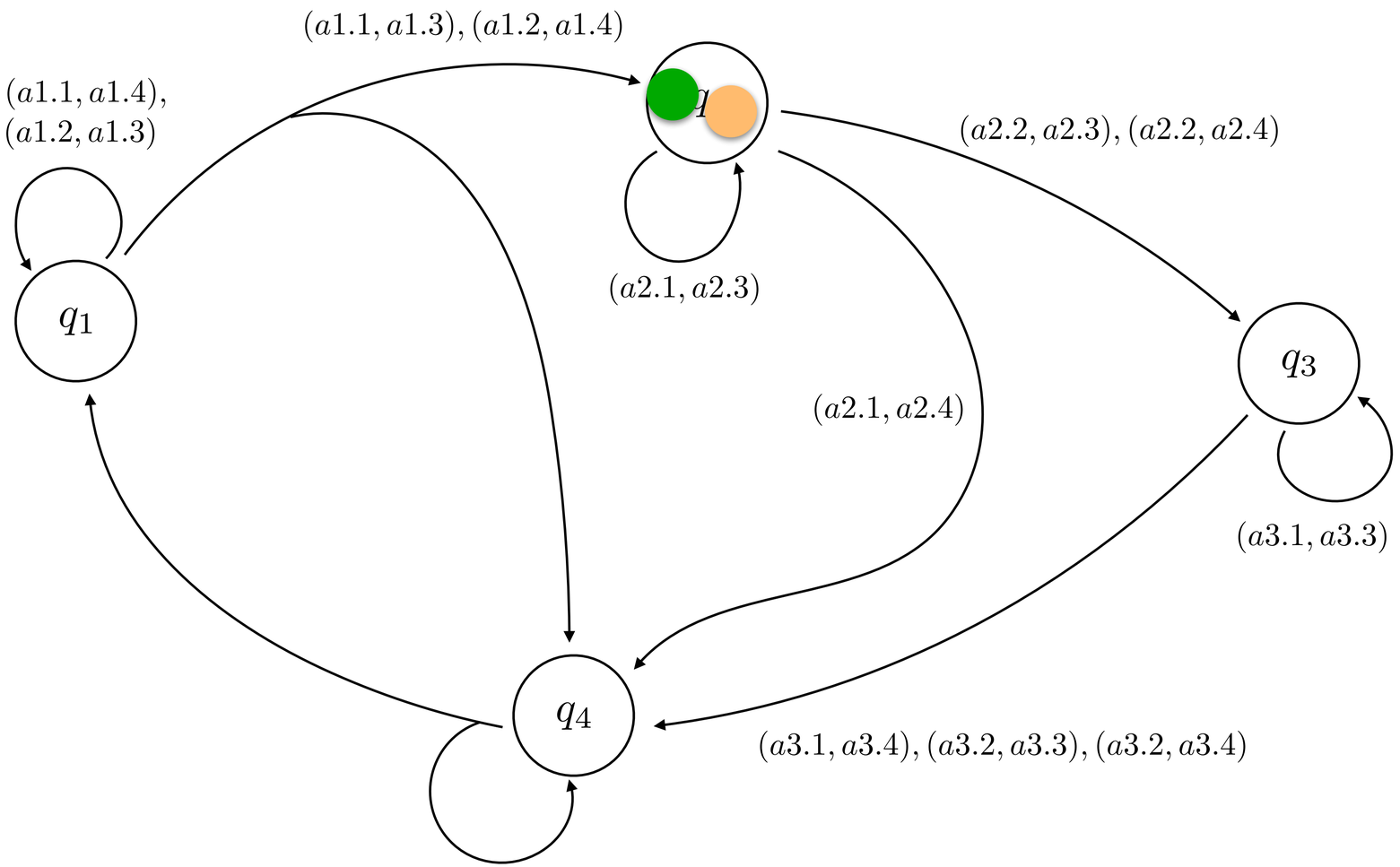
$$Q^i(q, a^1, \ldots, a^n) \leftarrow (1-\alpha)\, Q^i(q, a^1, \ldots, a^n) + \alpha \left(r^i(q, a^1, \ldots, a^n) + \gamma\, \text{NASH-Q}^i(q') \right)$$

(1, .5) 0.5 (0, 0) 0.5 (2, 1) 0.8 (0, 0)

γ

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_3

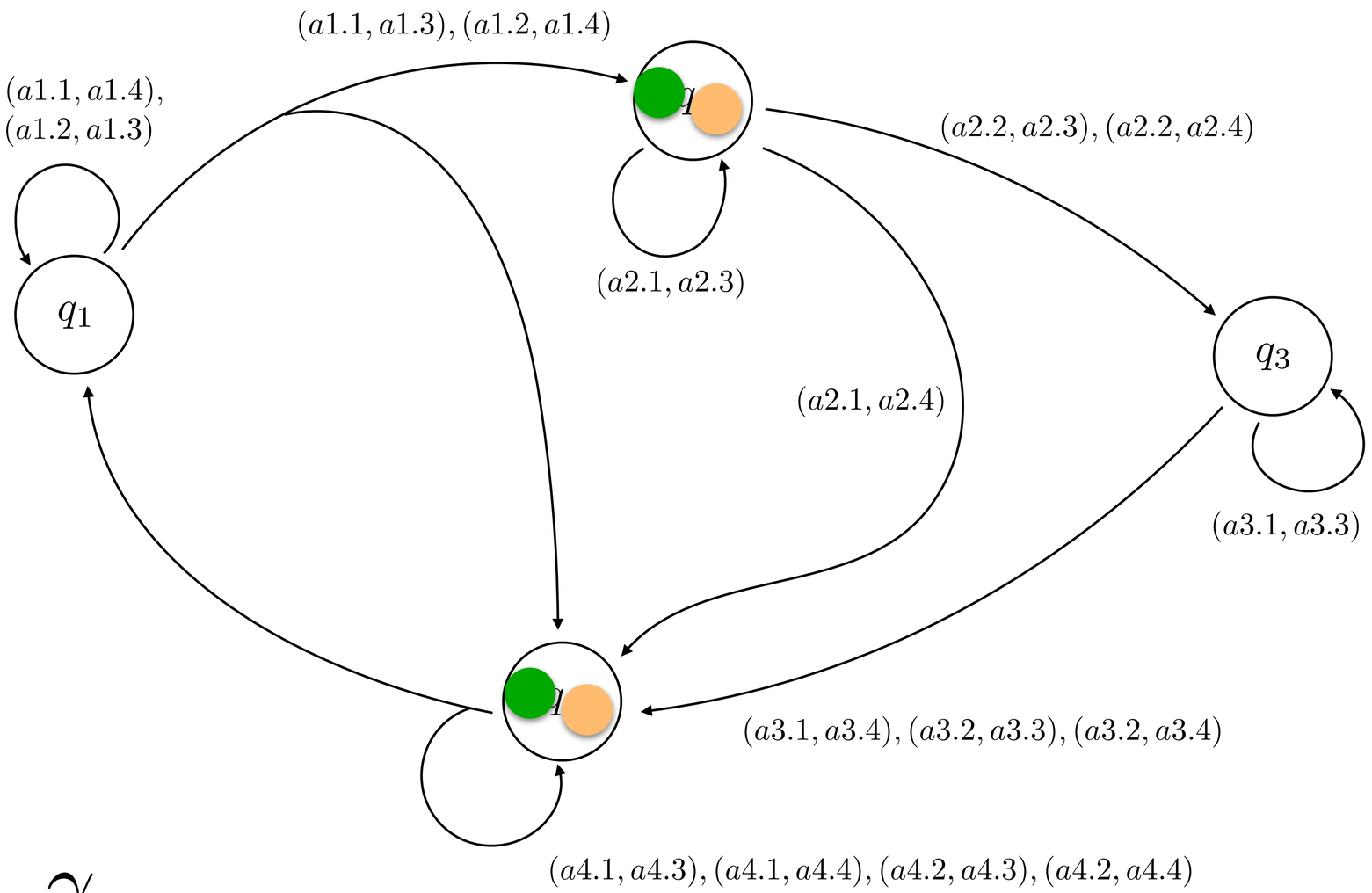
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a	a2.2
a2.3	1, 1	3, 0
a	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_3

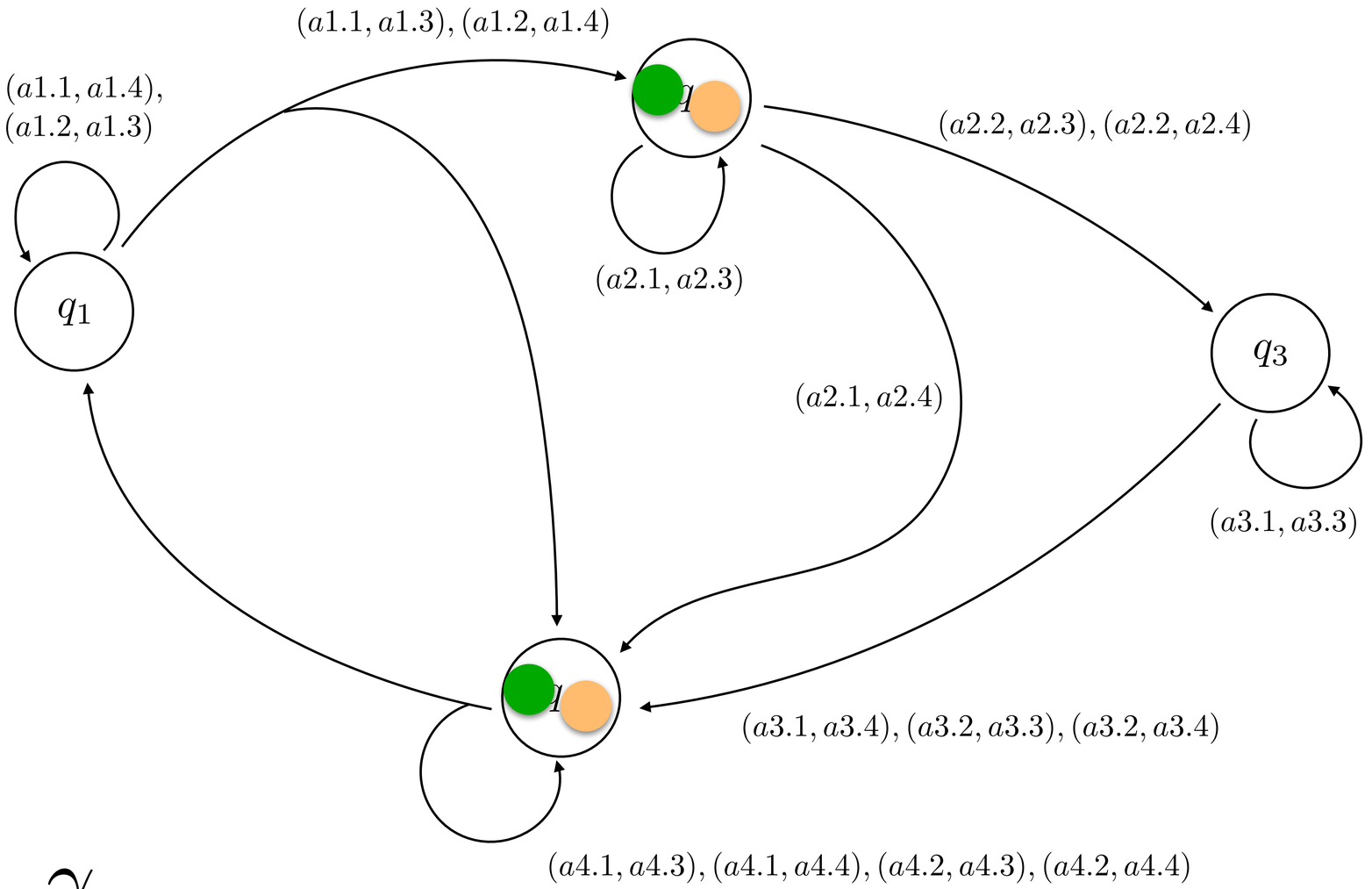
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

$$Q^i(q, a^1, \ldots, a^n) \leftarrow (1-\alpha)\, Q^i(q, a^1, \ldots, a^n) + \alpha \left(r^i(q, a^1, \ldots, a^n) + \gamma\, \text{NASH-Q}^i(q') \right)$$

(0, 1.5) 0.5 (0, 0) 0.5 (0, 3) 0.8 (0, 0)



γ

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 0	0, 0

q_3

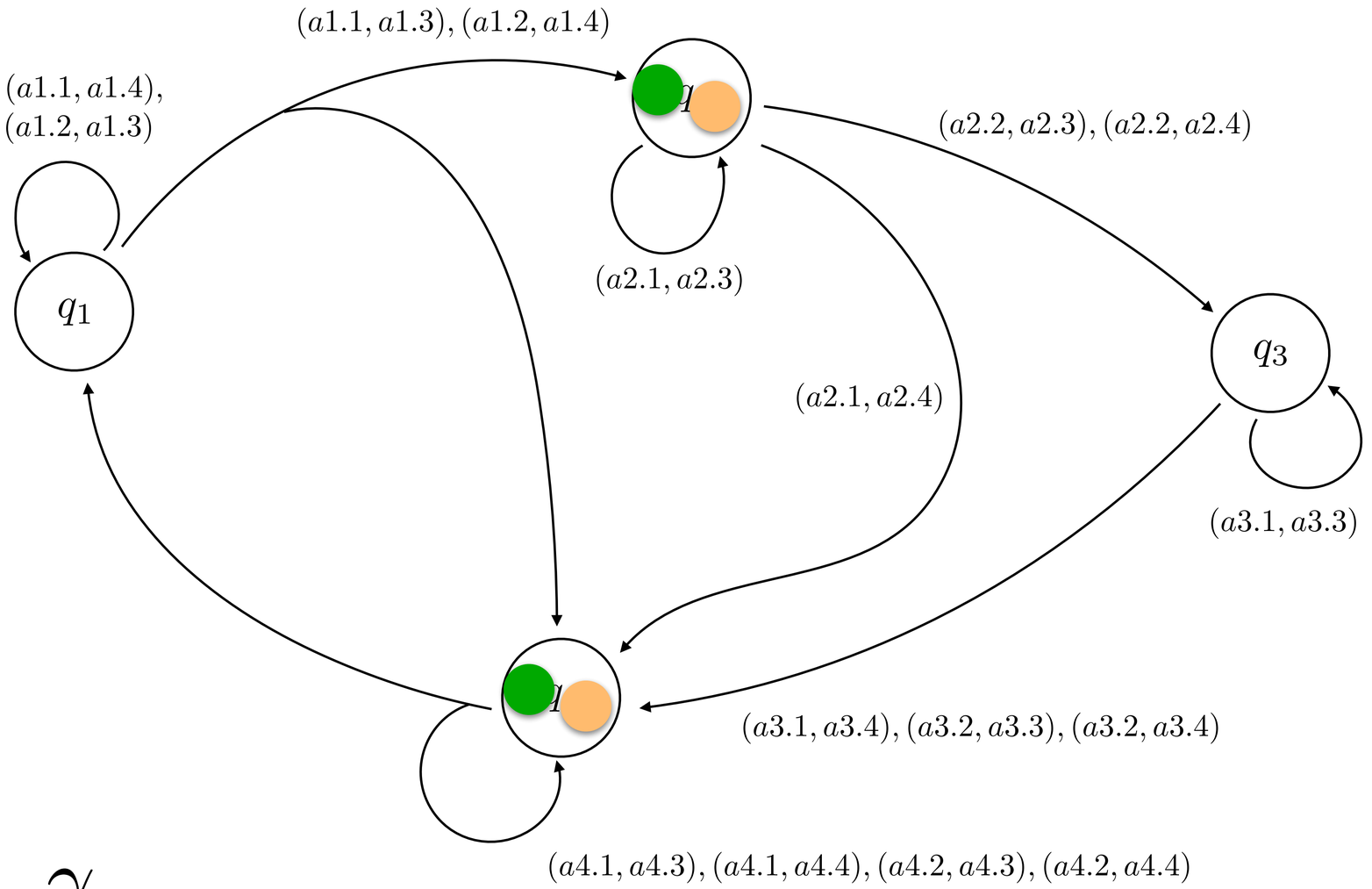
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

$$Q^i(q, a^1, \ldots, a^n) \leftarrow (1-\alpha)\, Q^i(q, a^1, \ldots, a^n) + \alpha \left(r^i(q, a^1, \ldots, a^n) + \gamma\, \text{NASH-Q}^i(q') \right)$$

(0, 1.5) 0.5 (0, 0) 0.5 (0, 3) 0.8 (0, 0)



γ

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	**0, 1.5**	0, 0

q_3

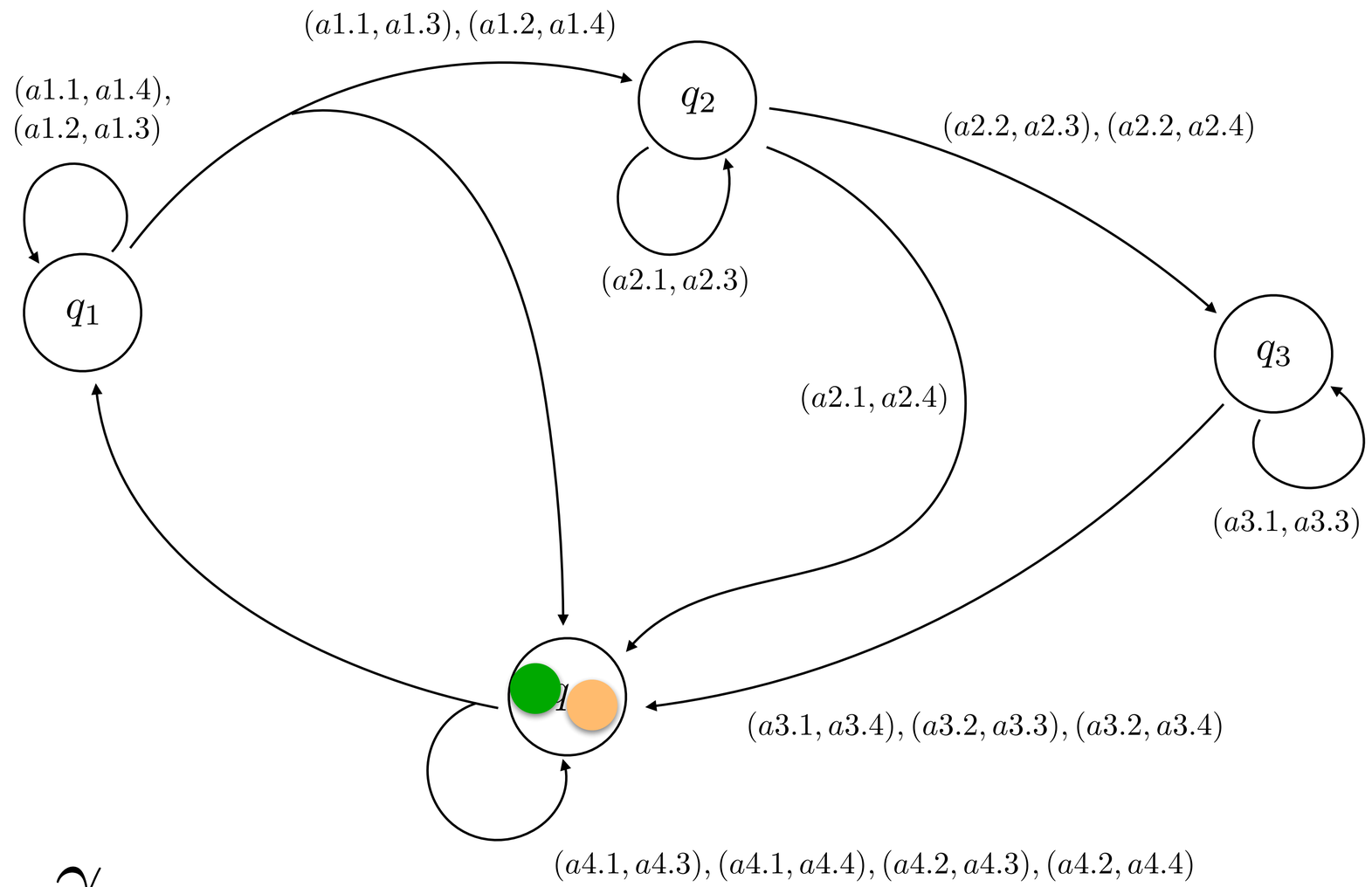
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_3

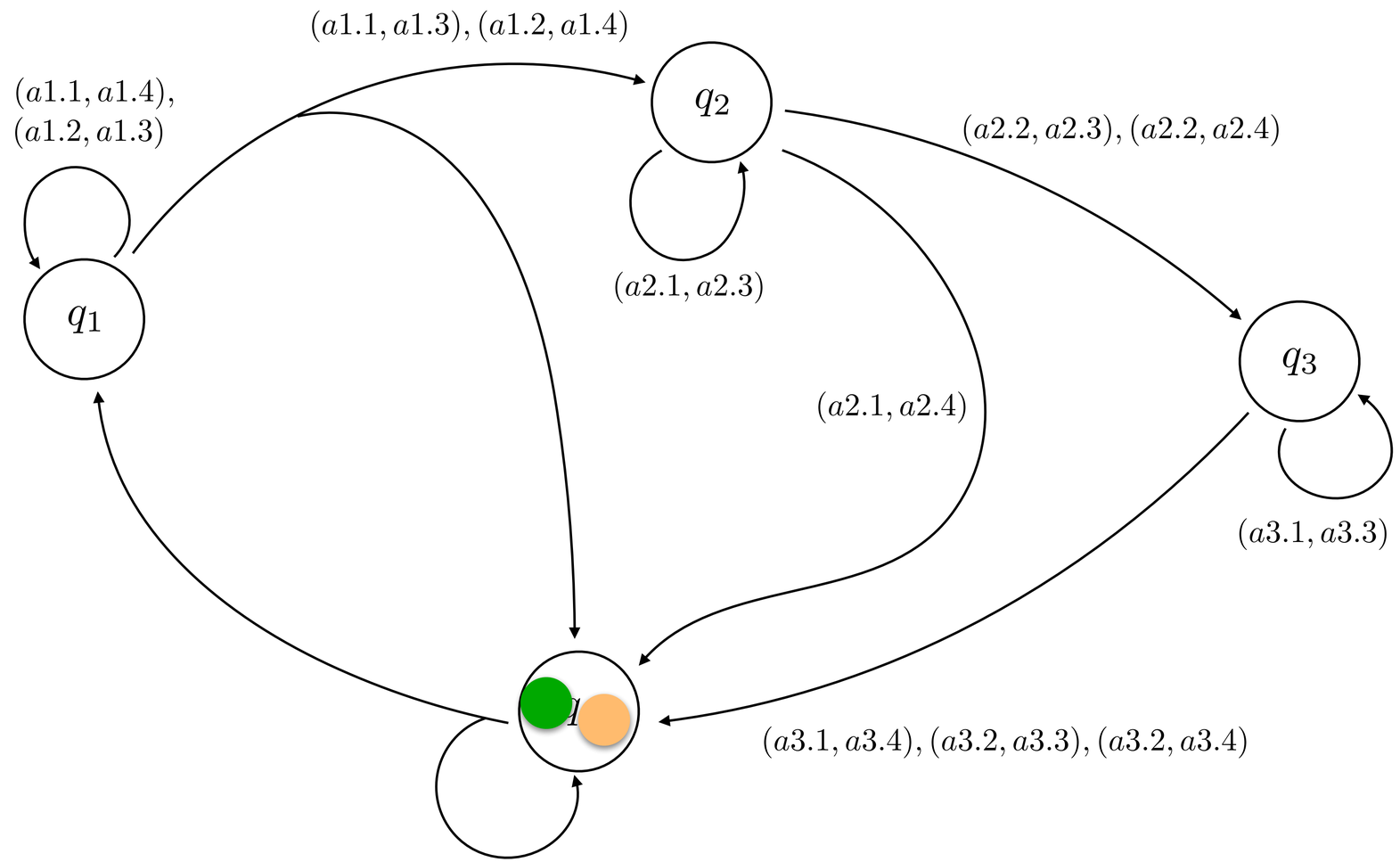
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 1.5	0, 0
a2.4	0, 0	0, 0

q_3

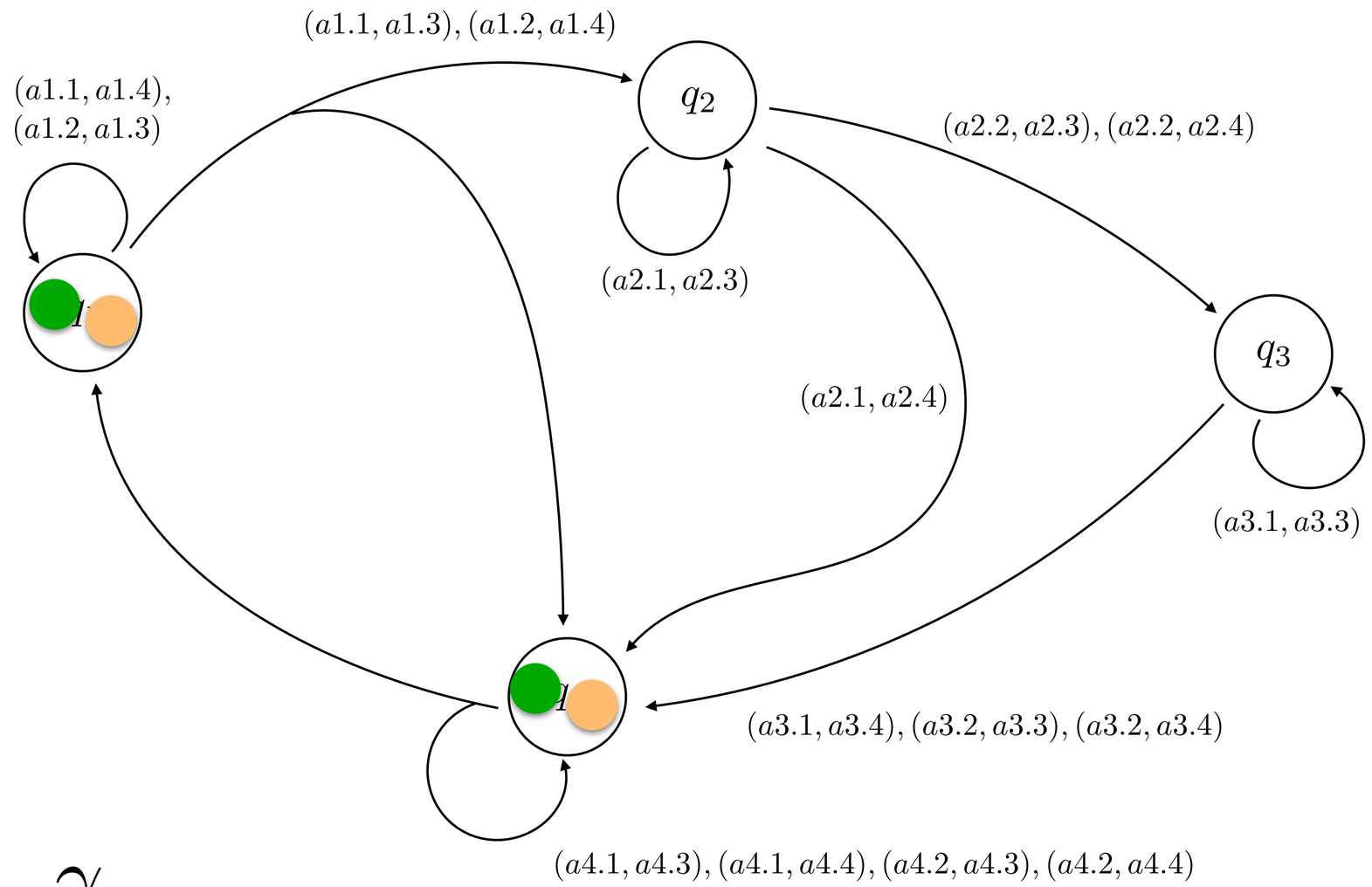
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_3

Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

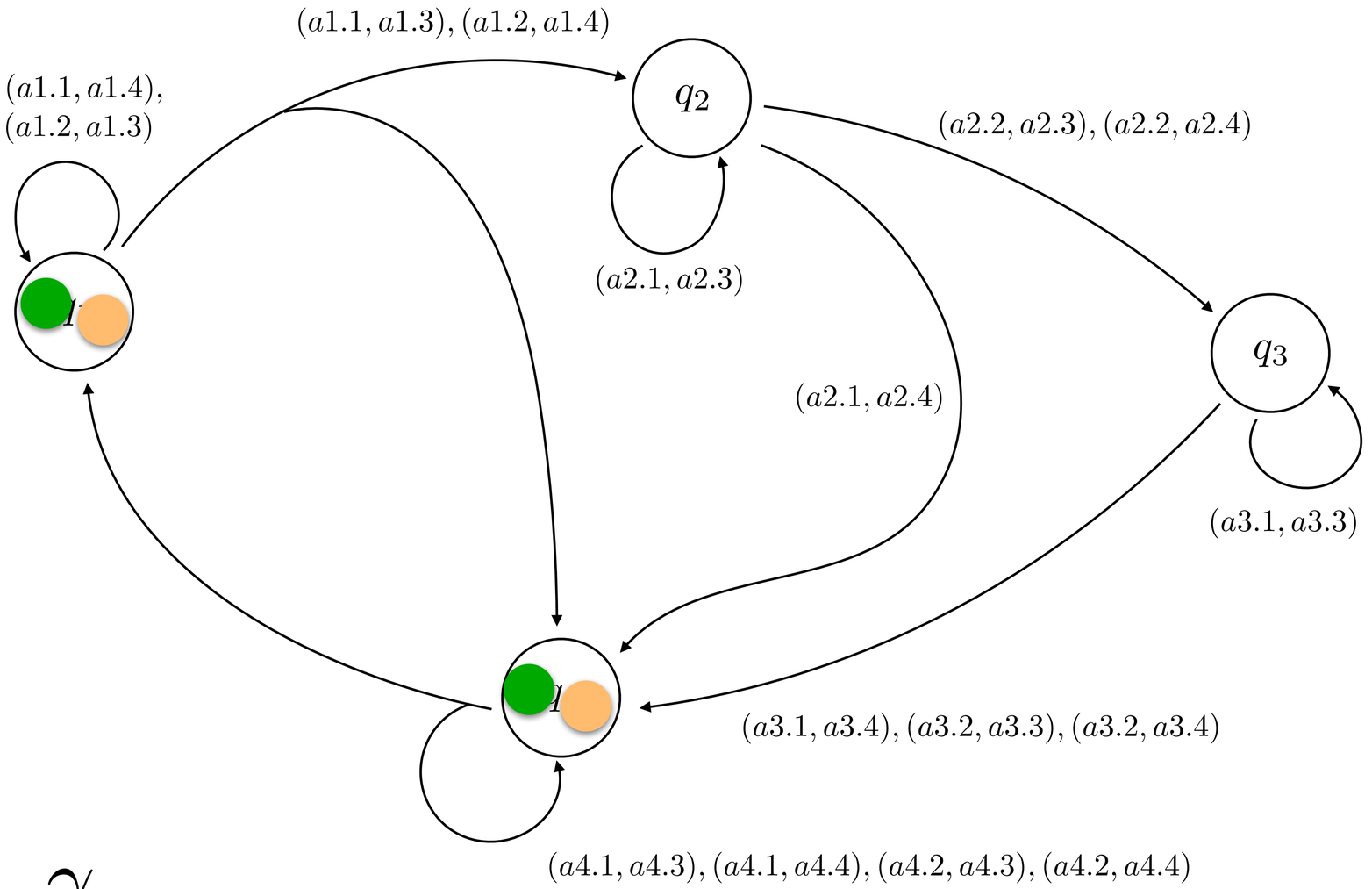
q_4

Nash Q values	a4.1	a4.2
a4.3	0, 0	0, 0
a4.4	0, 0	0, 0

$$Q^i(q, a^1, \ldots, a^n) \leftarrow (1-\alpha)\, Q^i(q, a^1, \ldots, a^n) + \alpha \left(r^i(q, a^1, \ldots, a^n) + \gamma\, \text{NASH-Q}^i(q') \right)$$

(0.4, 0.6) 0.5 (0, 0) 0.5 (0, 0) 0.8 (1, .5)

P



γ

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a	a4.2
a	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_3

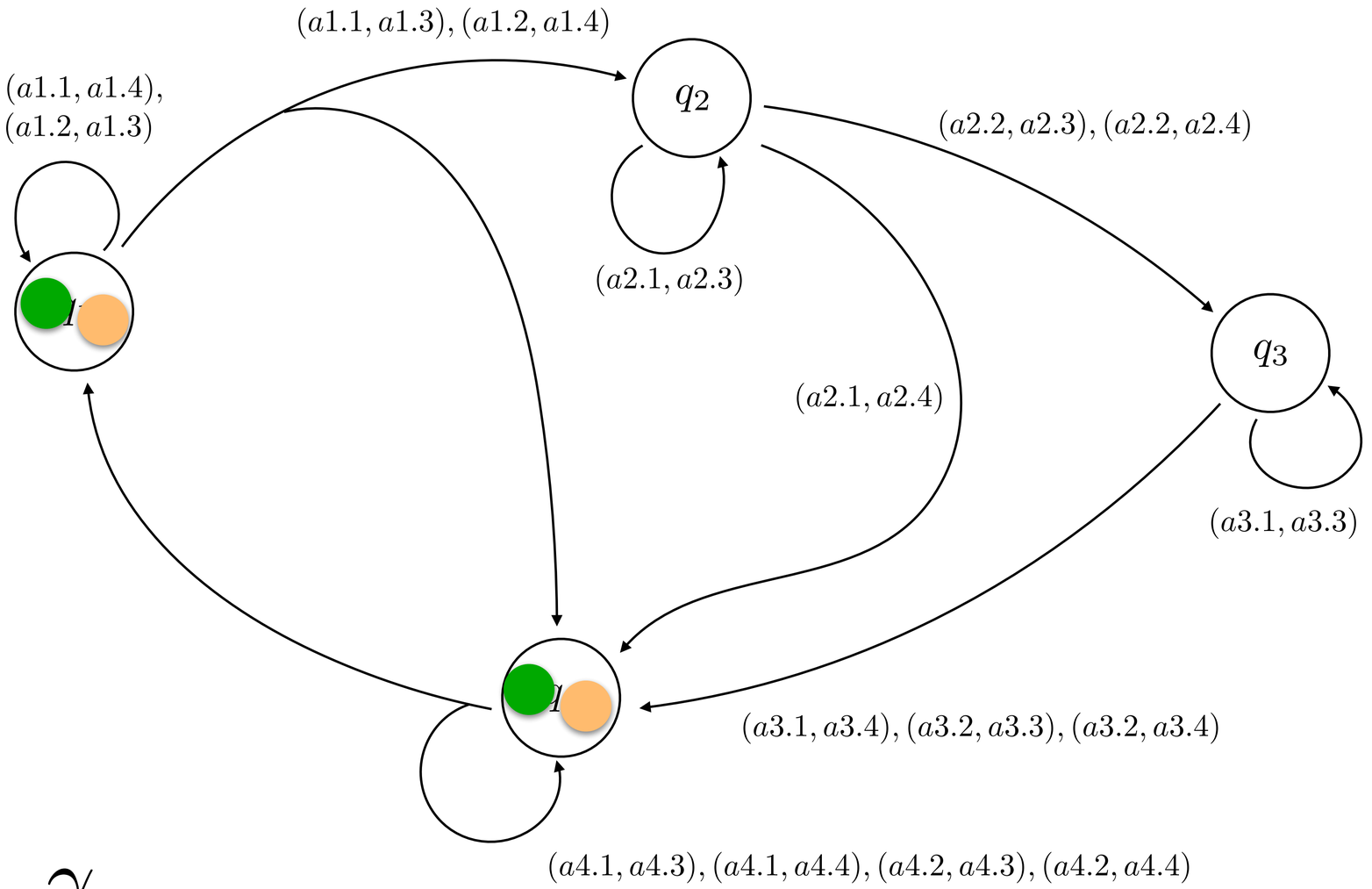
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a	a4.2
a	0, 0	0, 0
a4.4	0, 0	0, 0

$$Q^i(q, a^1, \ldots, a^n) \leftarrow (1-\alpha)\, Q^i(q, a^1, \ldots, a^n) + \alpha \left(r^i(q, a^1, \ldots, a^n) + \gamma\, \text{NASH-Q}^i(q') \right)$$

(0.4, 0.6) 0.5 (0, 0) 0.5 (0, 0) 0.8 (1, .5)



γ

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_3

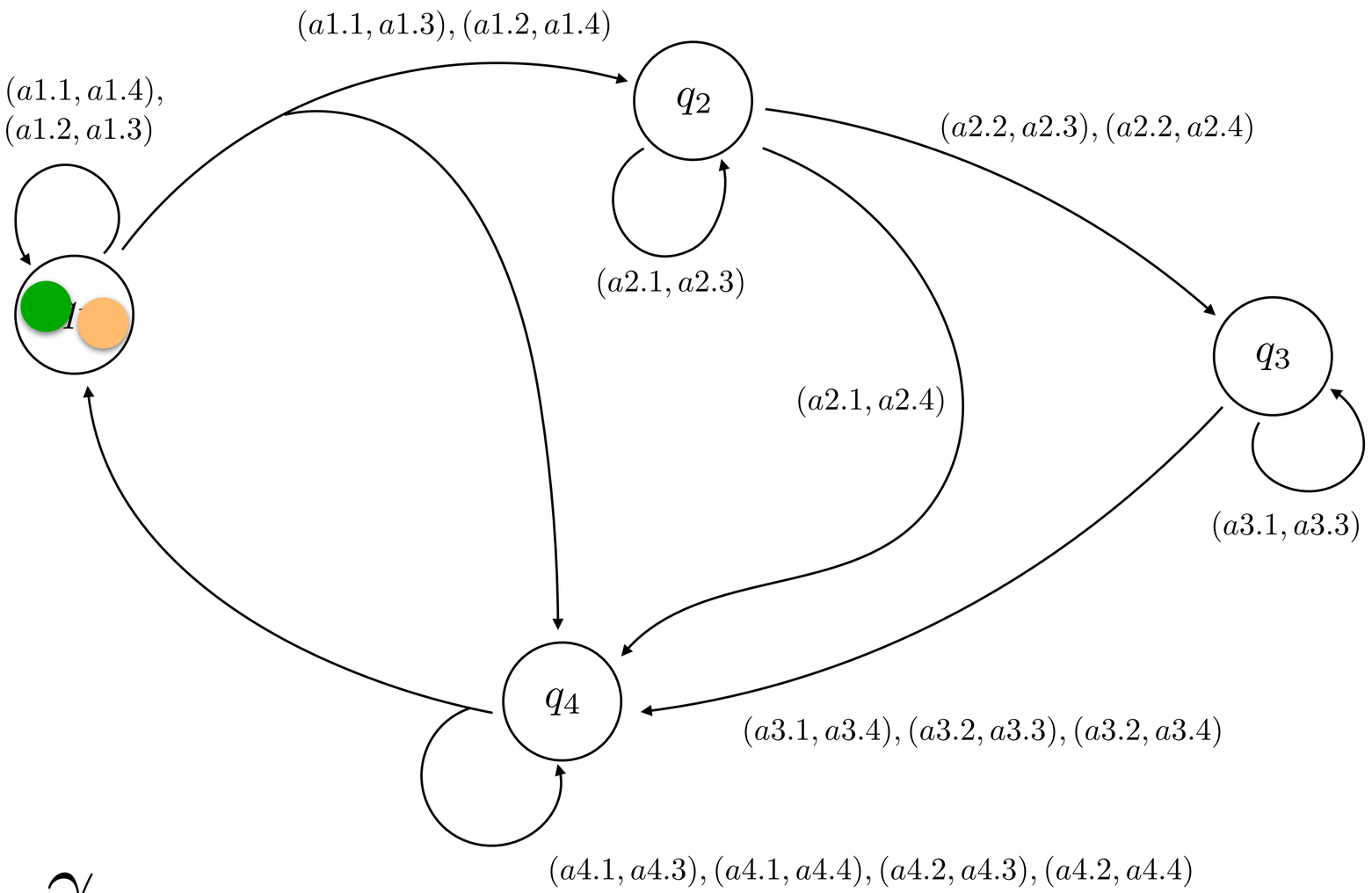
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	**.4, .6**	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_3

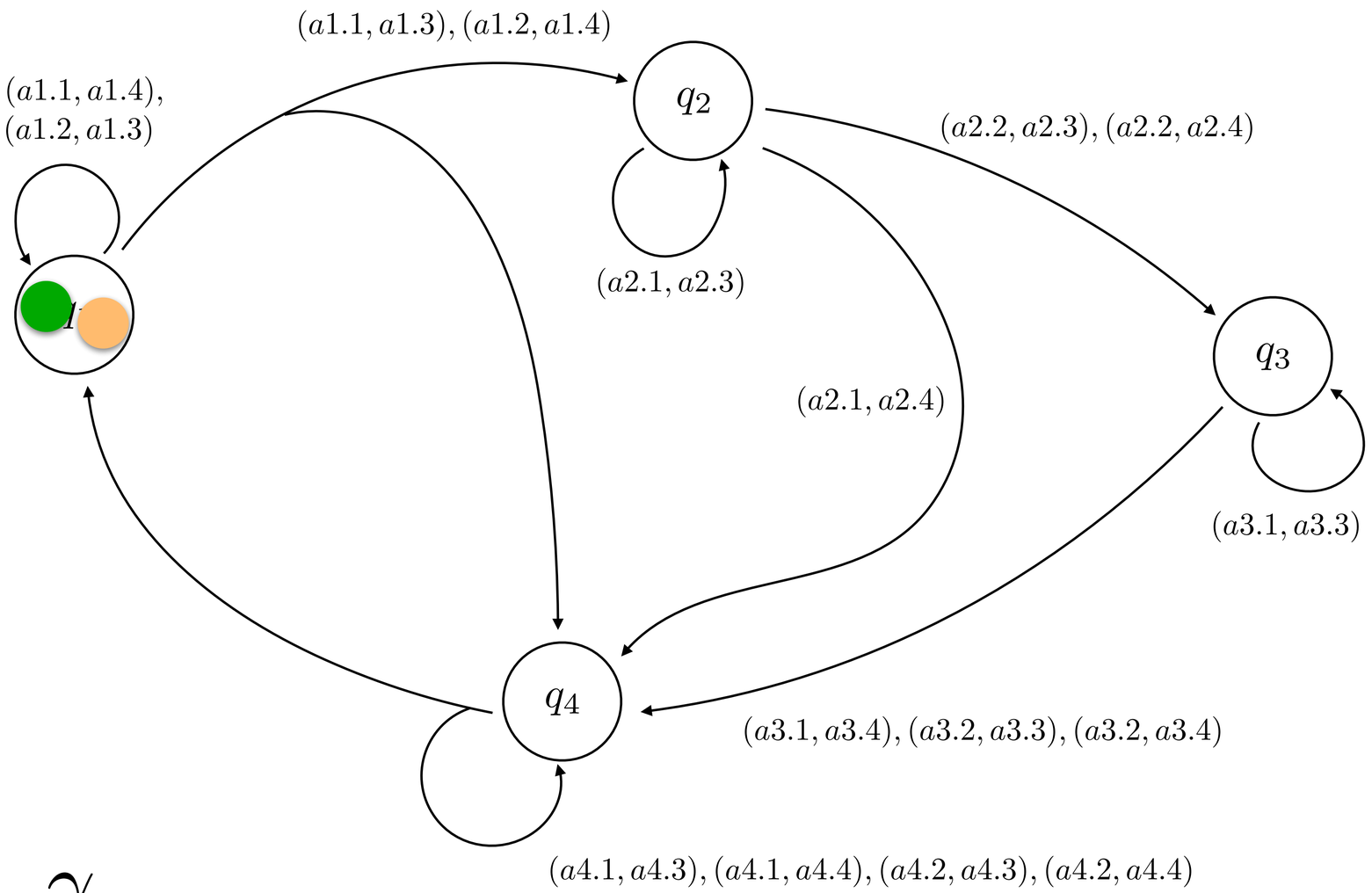
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	.4, .6	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_3

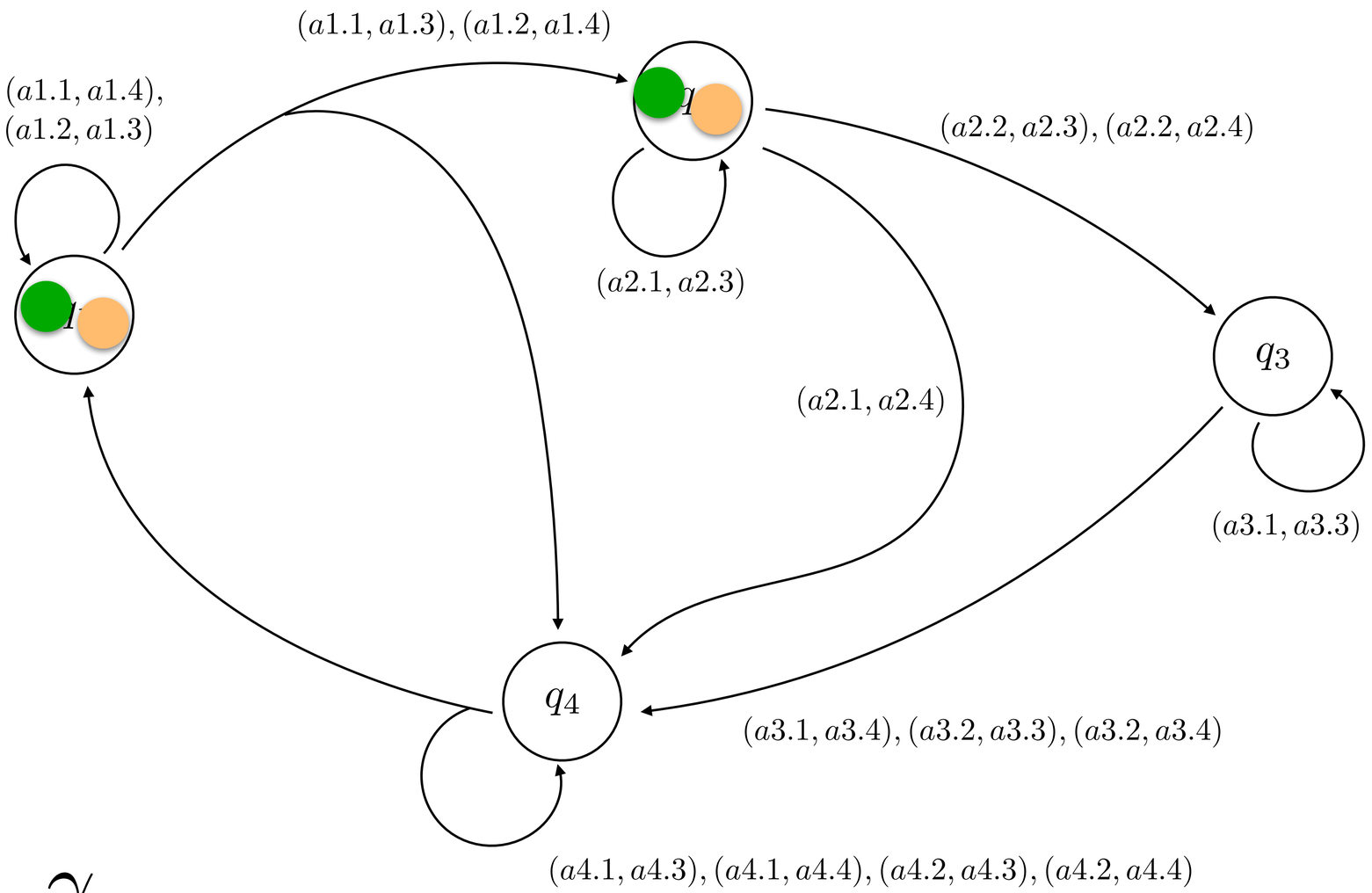
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	.4, .6	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_3

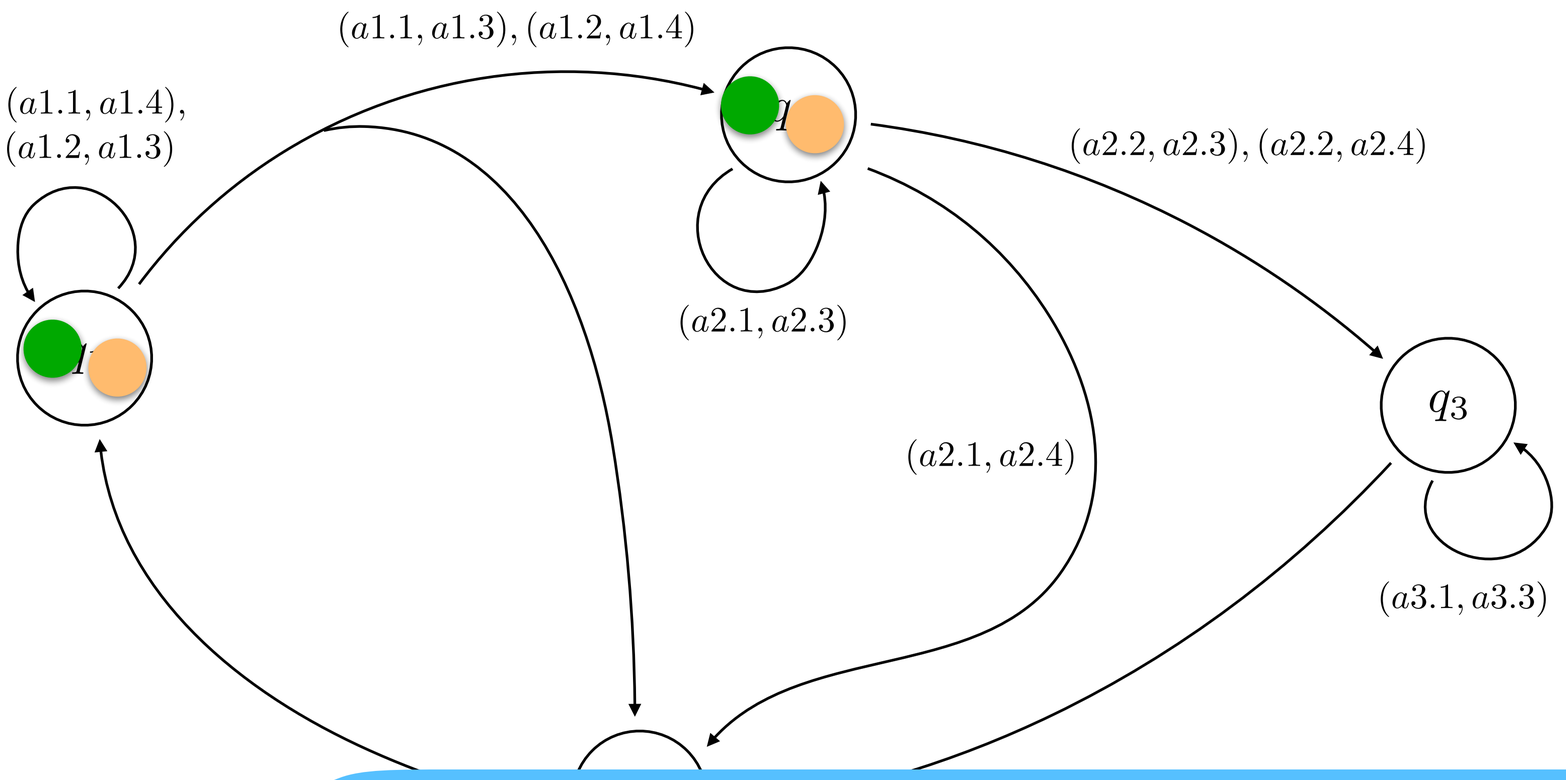
Nash Q values	a3.1	a3.2
a3.3	0, 0	0, 0
a3.4	0, 0	0, 0

q_4

Nash Q values	a4.1	a4.2
a4.3	.4, .6	0, 0
a4.4	0, 0	0, 0

An example

Players 1 and 2



q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1, .5	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_4

	a4.1	a4.2
	.6	0, 0
	0, 0	0, 0

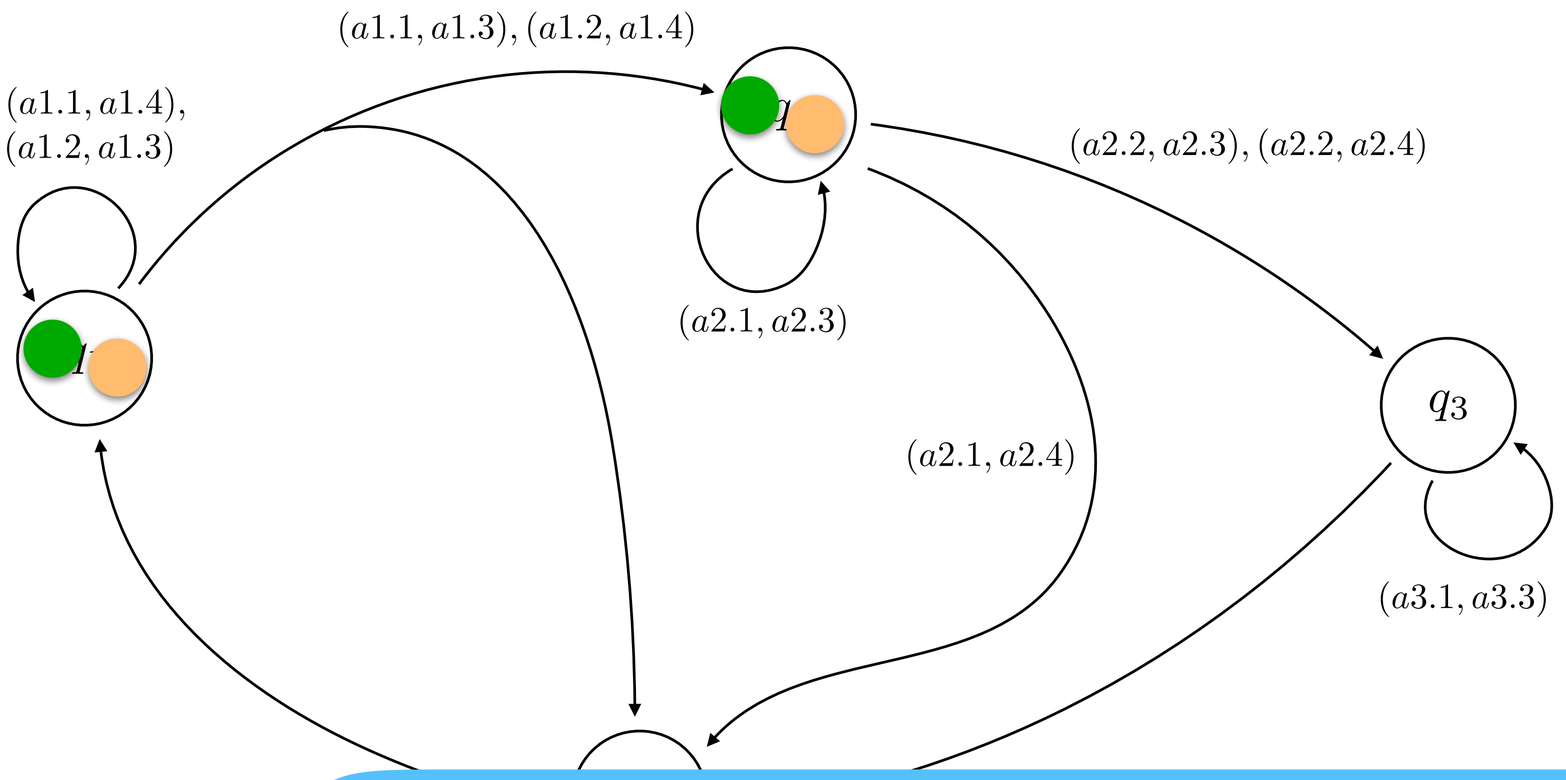
$$Q^i(q, a^1, \ldots, a^n) \leftarrow (1-\alpha)\, Q^i(q, a^1, \ldots, a^n) + \alpha \left(r^i(q, a^1, \ldots, a^n) + \gamma\, \text{NASH-Q}^i(q') \right)$$

(1.5, 1.35) 0.5 (1, 0.5) 0.5 (2, 1) 0.8 (0, 1.5)

γ

An example

Players 1 and 2



q_1

Reward	a1.1	a1.2
a1.3	2, 1	0, 0
a1.4	0, 0	1, 2

q_2

Reward	a2.1	a2.2
a2.3	1, 1	3, 0
a2.4	0, 3	2, 2

q_3

Reward	a3.1	a3.2
a3.3	2, 0	0, 2
a3.4	0, 1	1, 0

q_4

Reward	a4.1	a4.2
a4.3	1, 1	0, 0
a4.4	0, 0	2, 2

q_1

Nash Q values	a1.1	a1.2
a1.3	1.5, 1.35	0, 0
a1.4	0, 0	0, 0

q_2

Nash Q values	a2.1	a2.2
a2.3	0, 0	0, 0
a2.4	0, 1.5	0, 0

q_4

	a4.1	a4.2
	.6	0, 0
	0, 0	0, 0

$$Q^i(q, a^1, \ldots, a^n) \leftarrow (1-\alpha)\, Q^i(q, a^1, \ldots, a^n) + \alpha \left(r^i(q, a^1, \ldots, a^n) + \gamma\, \text{NASH-Q}^i(q') \right)$$

(1.5, 1.35) 0.5 (1, 0.5) 0.5 (2, 1) 0.8 (0, 1.5)

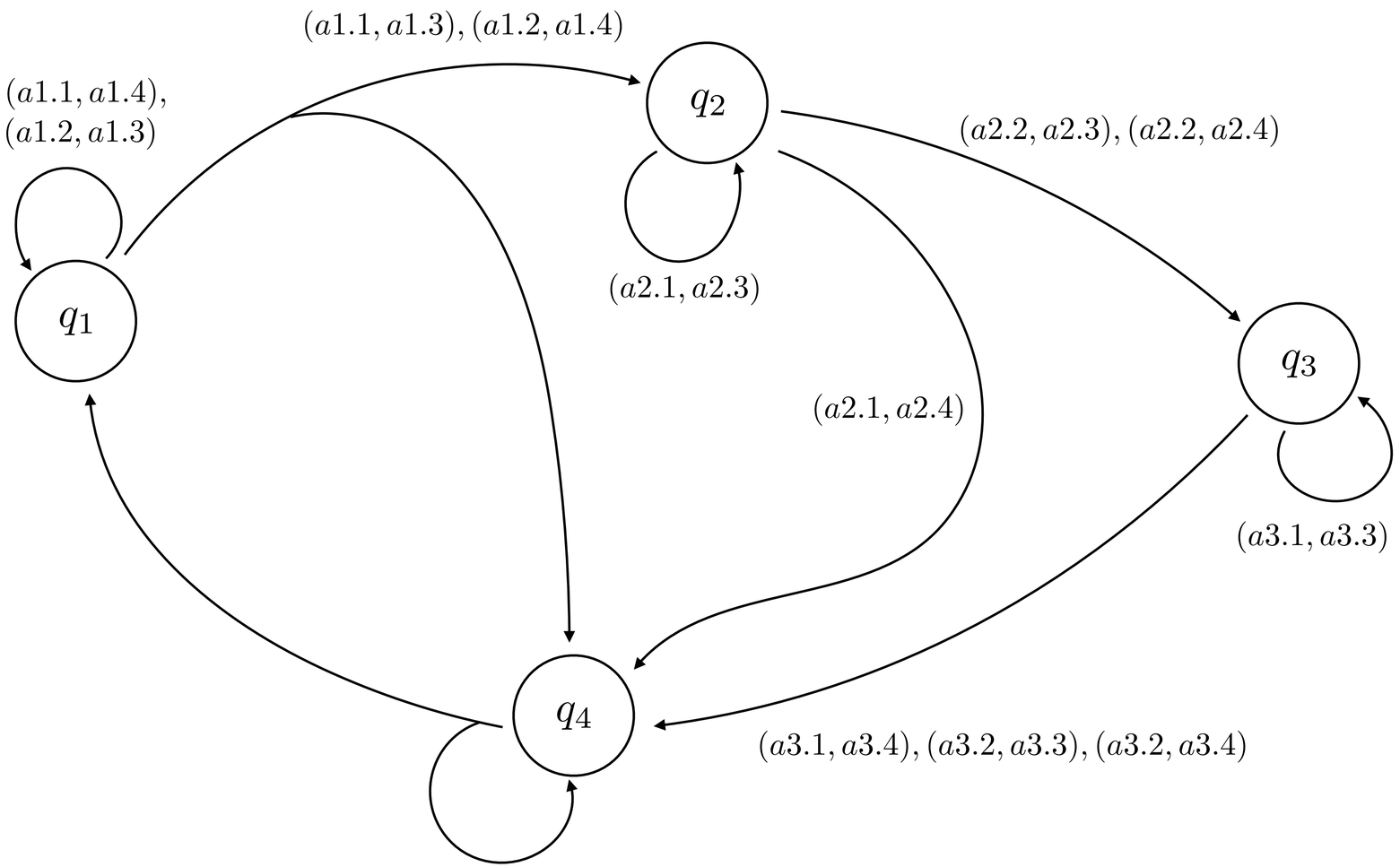
γ

Strategies Played

- **At every stage**
  - The players calculate a Nash equilibrium
  - If multiple Nash equilibria exist, they must agree on a specific Nash equilibrium
  - The actions of the players are drawn from the strategies
- **Exploration**
  - Epsilon-Nash strategy
    - With a probability of 1- epsilon, the Nash equilibrium is played
    - With a probability of epsilon, any joint strategy is played
  - Boltzmann exploration
    - Quantal equilibrium is computed in place of Nash equilibrium

Nash equilibrium

Players 1 and 2



γ

$$\sigma_1 = \begin{cases} a1.1 & 1.0 \\ a1.2 & 0.0 \end{cases} \quad \sigma_2 = \begin{cases} a1.3 & 1.0 \\ a1.4 & 0.0 \end{cases}$$

q_1

Nash Q values	a1.1	a1.2
a1.3	5, 4	2, 3
a1.4	3, 2	4, 5

$$\sigma_1 = \begin{cases} a2.1 & 0.5 \\ a2.2 & 0.5 \end{cases} \quad \sigma_2 = \begin{cases} a2.3 & 0.5 \\ a2.4 & 0.5 \end{cases}$$

q_2

Nash Q values	a2.1	a2.2
a2.3	3, 4	6, 3
a2.4	4, 5	5, 6

$$\sigma_1 = \begin{cases} a3.1 & 0.5 \\ a3.2 & 0.5 \end{cases} \quad \sigma_2 = \begin{cases} a3.3 & 0.5 \\ a3.4 & 0.5 \end{cases}$$

q_3

Nash Q values	a3.1	a3.2
a3.3	3, 4	6, 3
a3.4	4, 5	5, 6

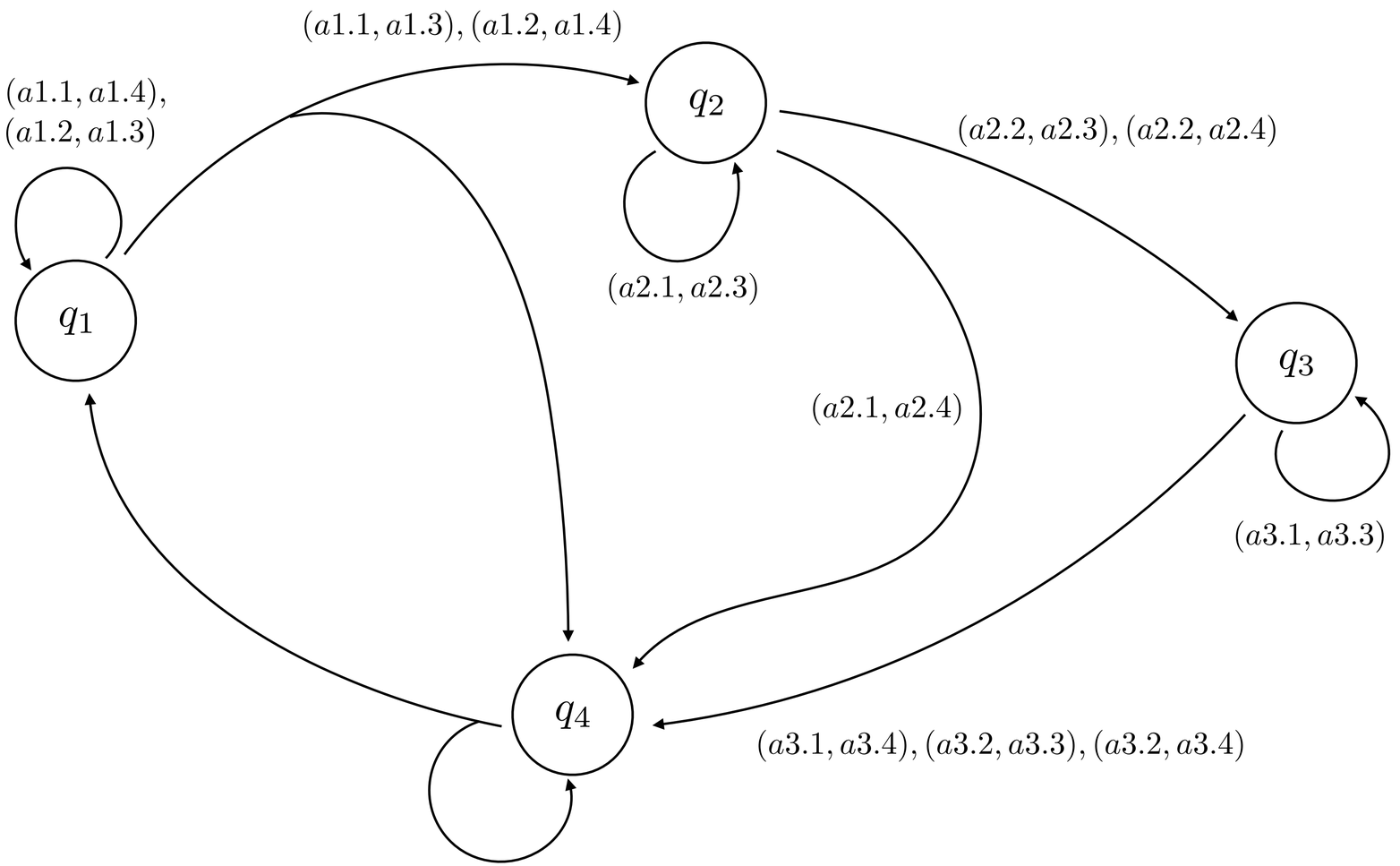
$$\sigma = \begin{cases} a4.1 & 1.0 \\ a4.2 & 0.0 \end{cases} \quad \sigma_2 = \begin{cases} a4.3 & 1.0 \\ a4.4 & 0.0 \end{cases}$$

q_4

Nash Q values	a4.1	a4.2
a4.3	5, 4	2, 3
a4.4	3, 2	5, 4

Epsilon-Nash strategy (exploration)

Players 1 and 2



γ

$$\sigma = \begin{cases} & a1.1 & a1.2 \\ a1.3 & 1-\epsilon\,3/4 & \epsilon/4 \\ a1.4 & \epsilon/4 & \epsilon/4 \end{cases}$$

$$\sigma = \begin{cases} & a2.1 & a2.2 \\ a2.3 & 0.5 & 0.5 \\ a2.4 & 0.5 & 0.5 \end{cases}$$

q_1

Nash Q values	a1.1	a1.2
a1.3	5, 4	2, 3
a1.4	3, 2	4, 5

q_2

Nash Q values	a2.1	a2.2
a2.3	3, 4	6, 3
a2.4	4, 5	5, 6

q_3

Nash Q values	a3.1	a3.2
a3.3	3, 4	6, 3
a3.4	4, 5	5, 6

q_4

Nash Q values	a4.1	a4.2
a4.3	5, 4	2, 3
a4.4	3, 2	5, 4

$$\sigma = \begin{cases} & a3.1 & a3.2 \\ a3.3 & 0.5 & 0.5 \\ a3.4 & 0.5 & 0.5 \end{cases}$$

$$\sigma = \begin{cases} & a4.1 & a4.2 \\ a4.3 & 1-\epsilon\,3/4 & \epsilon/4 \\ a4.4 & \epsilon/4 & \epsilon/4 \end{cases}$$

Convergence (1)

- Nash-Q values may not converge to the values of the Markov Nash equilibrium in general Stochastic games, even when the conditions assuring the convergence of the Q-learning are satisfied (learning rate and explorative strategies)

- A critical issue concerns the existence of multiple Nash equilibria in the stages

Convergence (2)

- In addition to the sufficient conditions for the convergence of the Q-learning algorithm, we require, for the convergence of the Nash-Q learning algorithm, one of the following
  - **Case 1**
    - For every game, there is a global optimum: a joint strategy providing the maximum possible reward (of that game) to all the players (this global optimum is a Nash equilibrium)
    - The global optima must be chosen as Nash equilibria during the whole learning dynamics
  - **Case 2**
    - For every game, there is a saddle point: a joint strategy that is a Nash equilibrium and is such that the reward of every player increases if at least an opponent deviates
    - The saddle points must be chosen as Nash equilibria during the whole learning dynamics

Examples

Reward	a1.1	a1.2
a1.3	3, 2	0, 1
a1.4	0, 0	1, 0

Global optimum

Reward	a1.1	a1.2
a1.3	5, 5	0, 6
a1.4	6, 0	2, 2

Saddle point

Reward	a1.1	a1.2
a1.3	6, 6	0, 3
a1.4	3, 0	2, 2

Global optimum

Saddle point

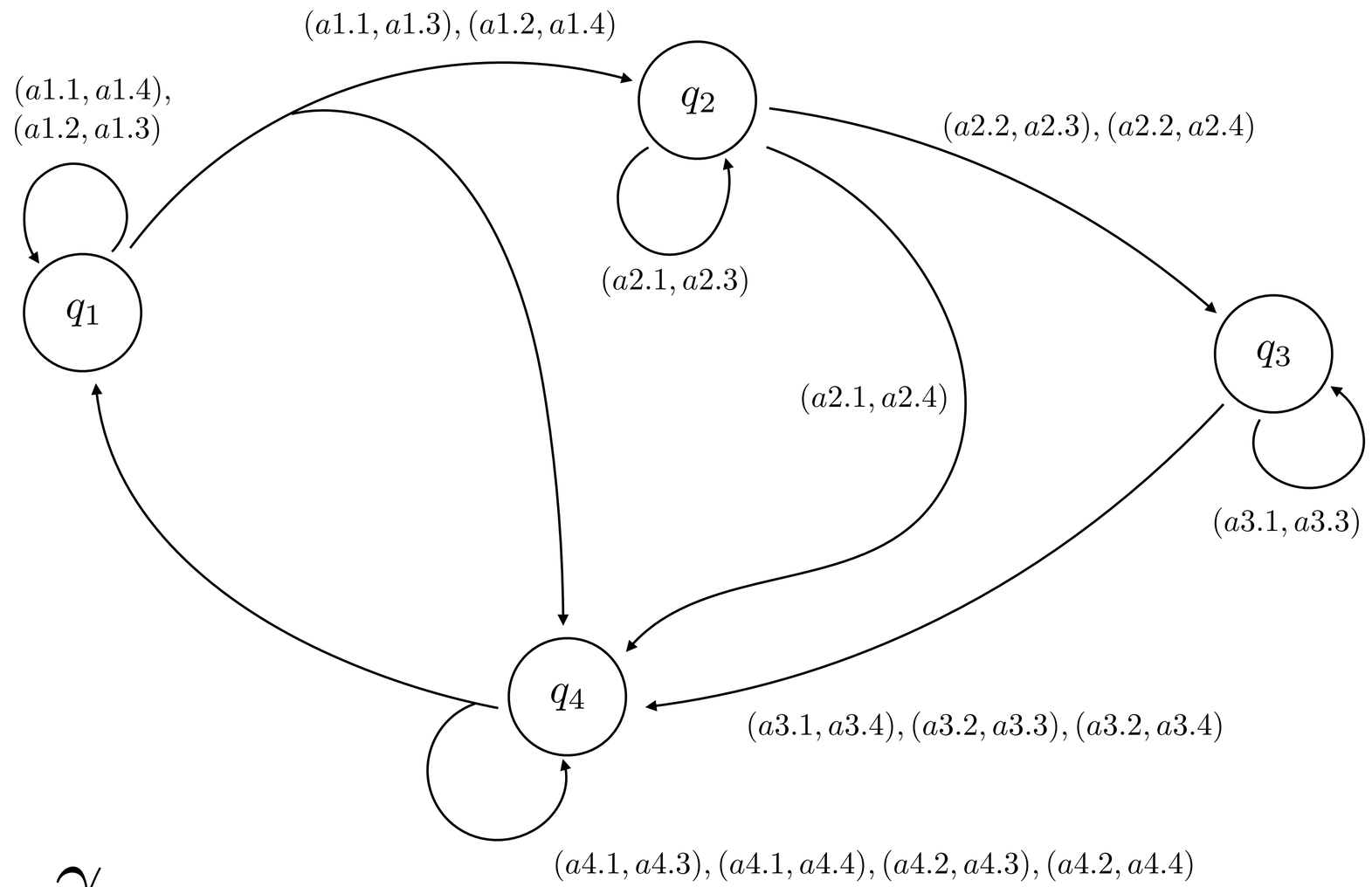
Convergence (3)

- In practice, it is sufficient that, in every stage game the same Nash equilibrium is played by all the players from a given time point on

Minimax-Q learning

2-player zero-sum stochastic game

Players 1 and 2



q_1

Reward	a1.1	a1.2
a1.3	1	-1
a1.4	-1	1

q_2

Reward	a2.1	a2.2
a2.3	2	-1
a2.4	3	2

q_3

Reward	a3.1	a3.2
a3.3	2	-1
a3.4	-3	2

q_4

Reward	a4.1	a4.2
a4.3	1	-1
a4.4	-1	1

γ

Minimax-Q learning (1)

- In every state, the Q-table stores the Q-values defined on the joint actions of player max

q_1

Reward	a1.1	a1.2
a1.3	1	-1
a1.4	-1	1

q_2

Reward	a2.1	a2.2
a2.3	2	-1
a2.4	3	2

q_3

Reward	a3.1	a3.2
a3.3	2	-1
a3.4	-3	2

q_4

Reward	a4.1	a4.2
a4.3	1	-1
a4.4	-1	1

q_1

Nash Q values	a1.1	a1.2
a1.3		
a1.4		

q_2

Nash Q values	a2.1	a2.2
a2.3		
a2.4		

q_3

Nash Q values	a3.1	a3.2
a3.3		
a3.4		

q_4

Nash Q values	a4.1	a4.2
a4.3		
a4.4		

Minimax-Q learning (2)

- The update of the Nash-Q values of every player is as follows:

$$Q(q, a^1, a^2) \leftarrow (1-\alpha)\, Q(q, a^1, a^2) + \alpha \left(r(q, a^1, a^2) + \gamma\, \textsc{Maxmin-Q}^i(q') \right)$$

Game

Action of player *j*

Reward gotten in game *q* when playing the given action profile

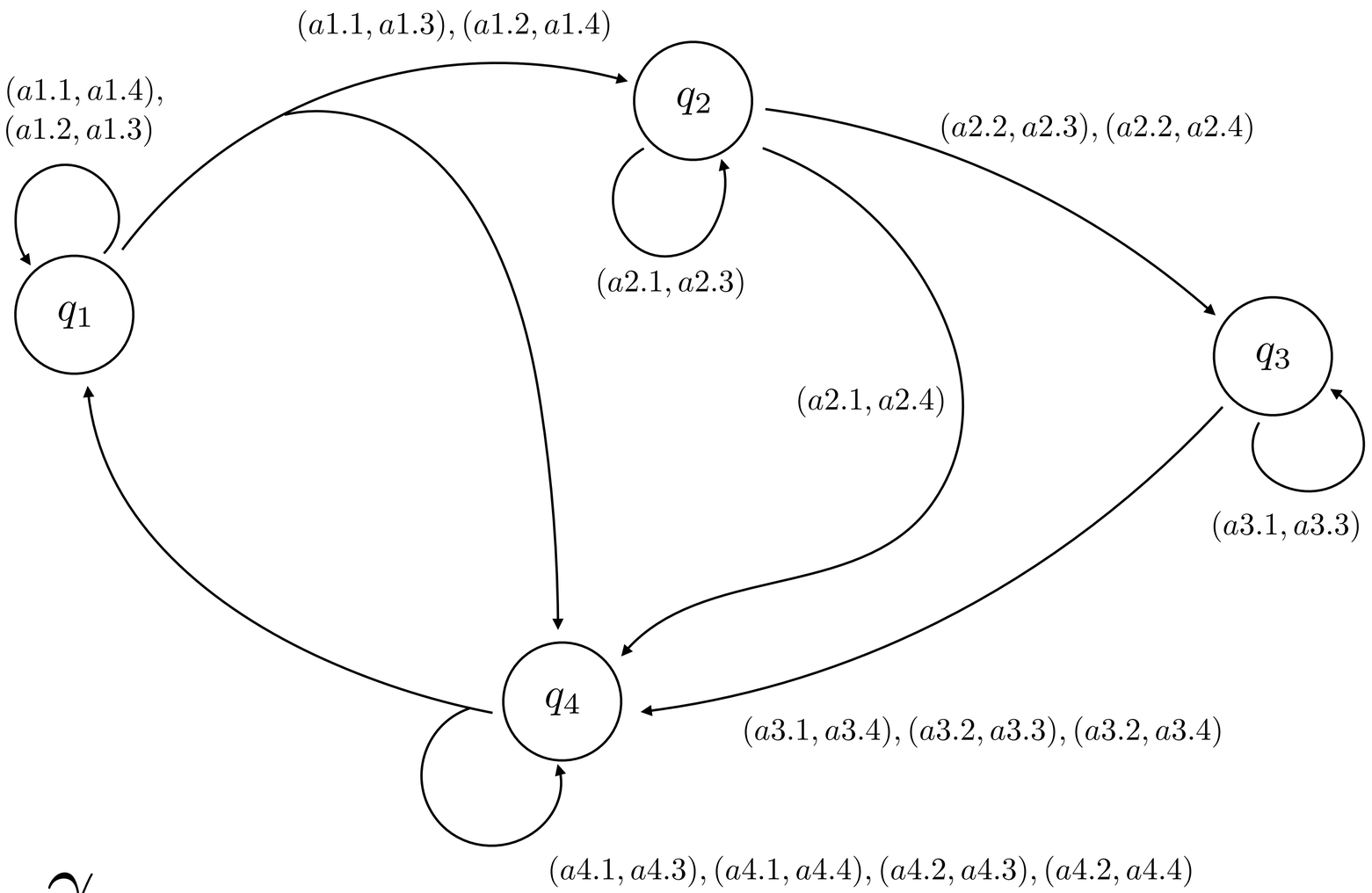
Value for player *i* of the MaxMin equilibrium computed at state *q'*

Convergence

- Minimax-Q converges the Markov Nash equilibrium in every zero-sum Stochastic game under the same assumptions for the convergence of Q-learning in stationary MDP games

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	1	-1
a1.4	-1	1

q_2

Reward	a2.1	a2.2
a2.3	2	-1
a2.4	3	2

q_3

Reward	a3.1	a3.2
a3.3	2	-1
a3.4	-3	2

q_4

Reward	a4.1	a4.2
a4.3	1	-1
a4.4	-1	1

q_1

Nash Q values	a1.1	a1.2
a1.3	0	0
a1.4	0	0

q_2

Nash Q values	a2.1	a2.2
a2.3	0	0
a2.4	0	0

q_3

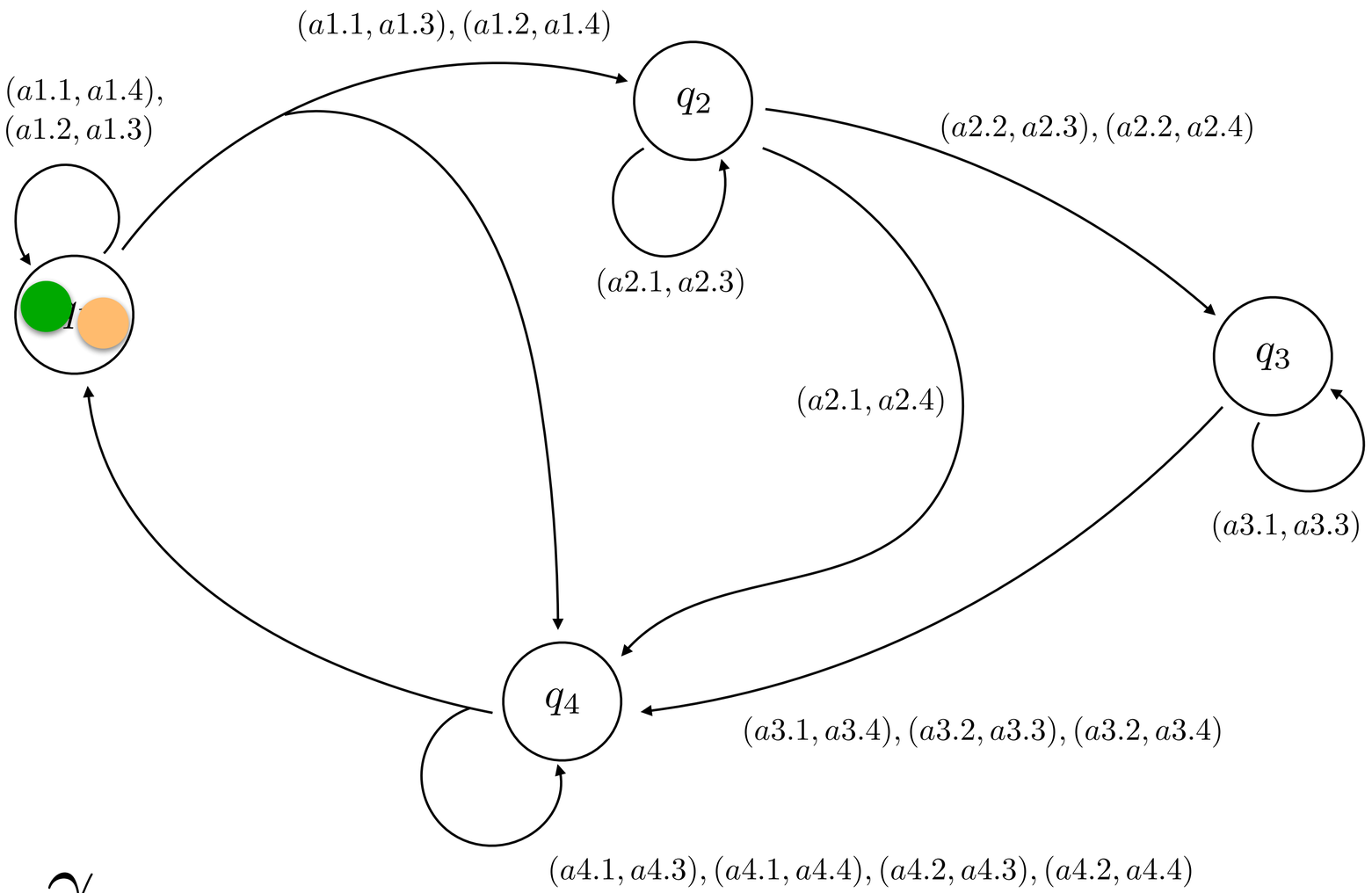
Nash Q values	a3.1	a3.2
a3.3	0	0
a3.4	0	0

q_4

Nash Q values	a4.1	a4.2
a4.3	0	0
a4.4	0	0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	1	-1
a1.4	-1	1

q_2

Reward	a2.1	a2.2
a2.3	2	-1
a2.4	3	2

q_3

Reward	a3.1	a3.2
a3.3	2	-1
a3.4	-3	2

q_4

Reward	a4.1	a4.2
a4.3	1	-1
a4.4	-1	1

q_1

Nash Q values	a1.1	a1.2
a1.3	0	0
a1.4	0	0

q_2

Nash Q values	a2.1	a2.2
a2.3	0	0
a2.4	0	0

q_3

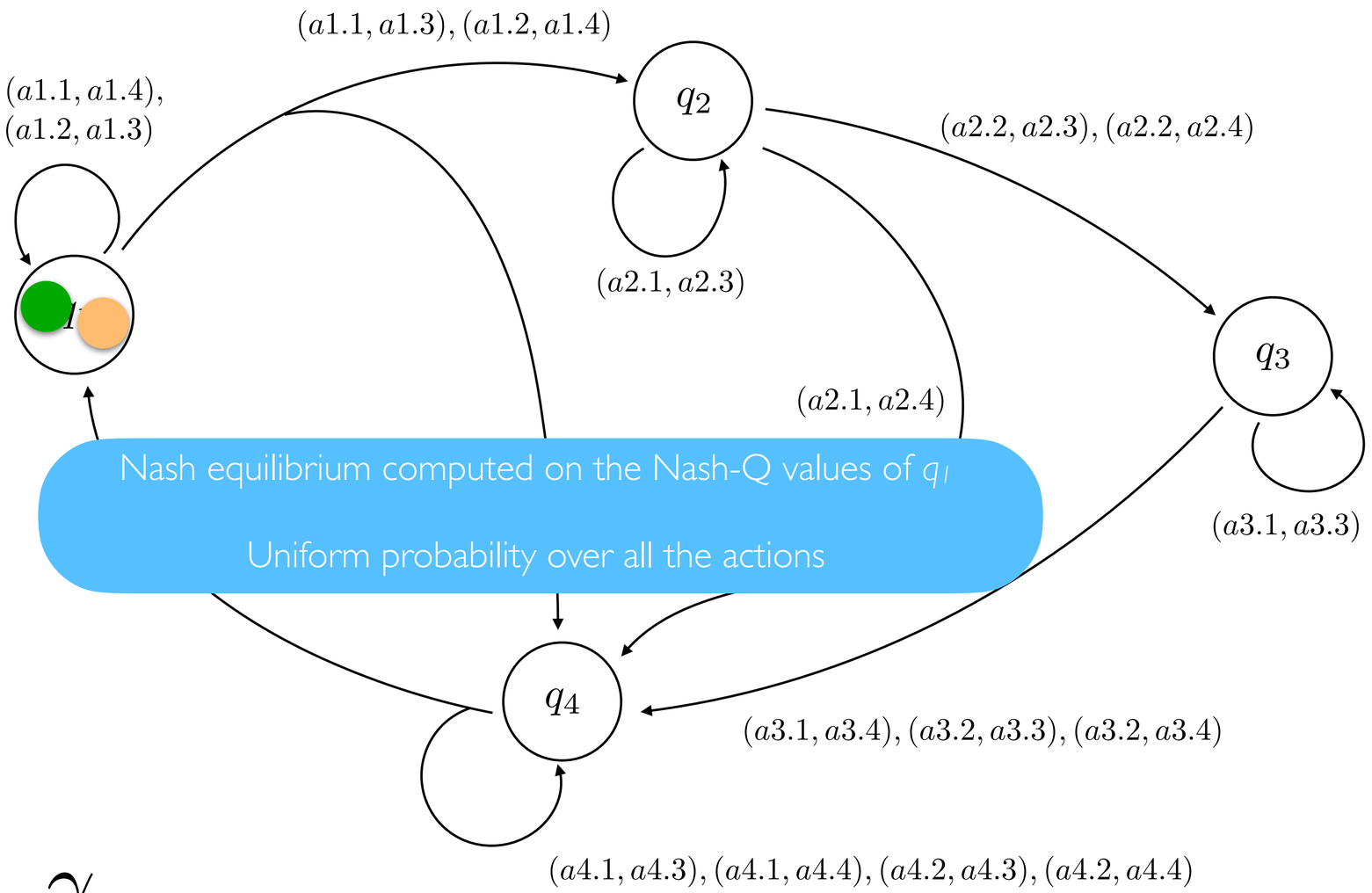
Nash Q values	a3.1	a3.2
a3.3	0	0
a3.4	0	0

q_4

Nash Q values	a4.1	a4.2
a4.3	0	0
a4.4	0	0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	1	-1
a1.4	-1	1

q_2

Reward	a2.1	a2.2
a2.3	2	-1
a2.4	3	2

q_3

Reward	a3.1	a3.2
a3.3	2	-1
a3.4	-3	2

q_4

Reward	a4.1	a4.2
a4.3	1	-1
a4.4	-1	1

q_1

Nash Q values	a1.1	a1.2
a1.3	0	0
a1.4	0	0

q_2

Nash Q values	a2.1	a2.2
a2.3	0	0
a2.4	0	0

q_3

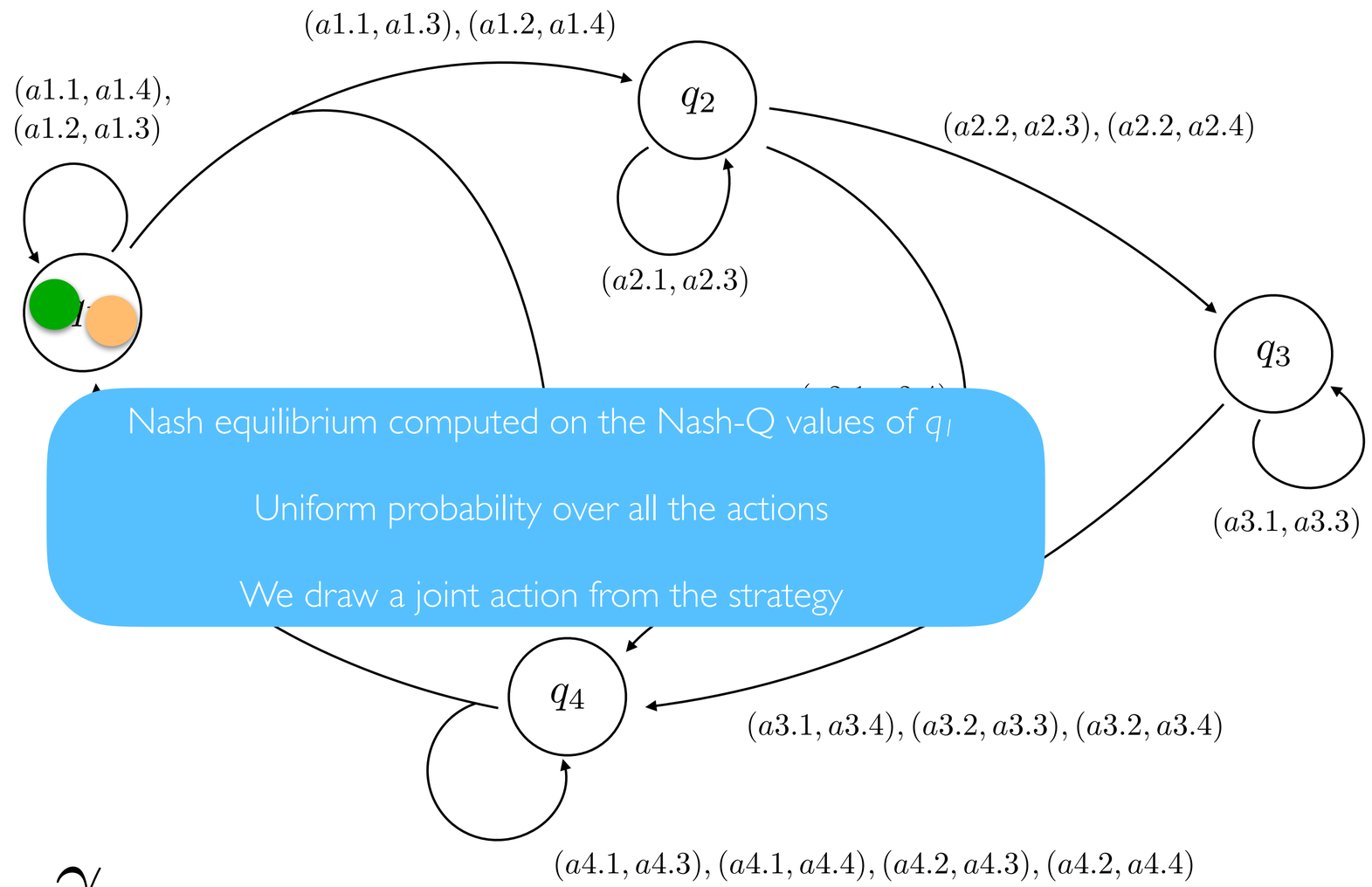
Nash Q values	a3.1	a3.2
a3.3	0	0
a3.4	0	0

q_4

Nash Q values	a4.1	a4.2
a4.3	0	0
a4.4	0	0

An example

Players 1 and 2



γ

q_1

Reward	a1.1	a1.2
a1.3	1	-1
a1.4	-1	1

q_2

Reward	a2.1	a2.2
a2.3	2	-1
a2.4	3	2

q_3

Reward	a3.1	a3.2
a3.3	2	-1
a3.4	-3	2

q_4

Reward	a4.1	a4.2
a4.3	1	-1
a4.4	-1	1

q_1

Nash Q values	a1.1	a1.2
a1.3	0	0
a1.4	0	0

q_2

Nash Q values	a2.1	a2.2
a2.3	0	0
a2.4	0	0

q_3

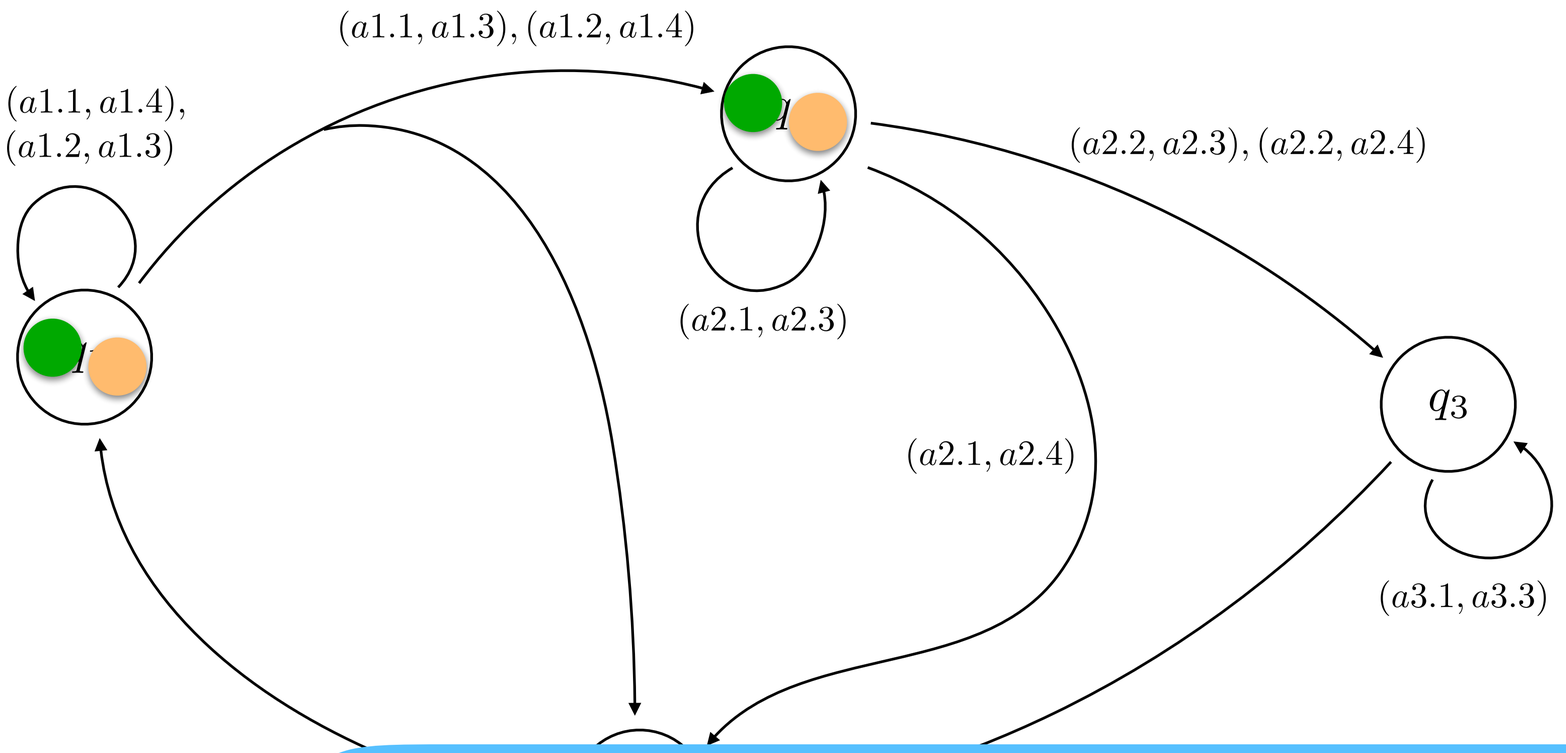
Nash Q values	a3.1	a3.2
a3.3	0	0
a3.4	0	0

q_4

Nash Q values	a4.1	a4.2
a4.3	0	0
a4.4	0	0

An example

Players 1 and 2



q_1

Reward	a1.1	a1.2
a1.3	1	-1
a1.4	-1	1

q_2

Reward	a2.1	a2.2
a2.3	2	-1
a2.4	3	2

q_3

Reward	a3.1	a3.2
a3.3	2	-1
a3.4	-3	2

q_4

Reward	a4.1	a4.2
a4.3	1	-1
a4.4	-1	1

q_1

Nash Q values	a1.1	a1.2
a1.3	0	0
a1.4	0	0

q_2

Nash Q values	a2.1	a2.2
a2.3	0	0
a2.4	0	0

q_4

	a4.1	a4.2
	0	0
	0	0

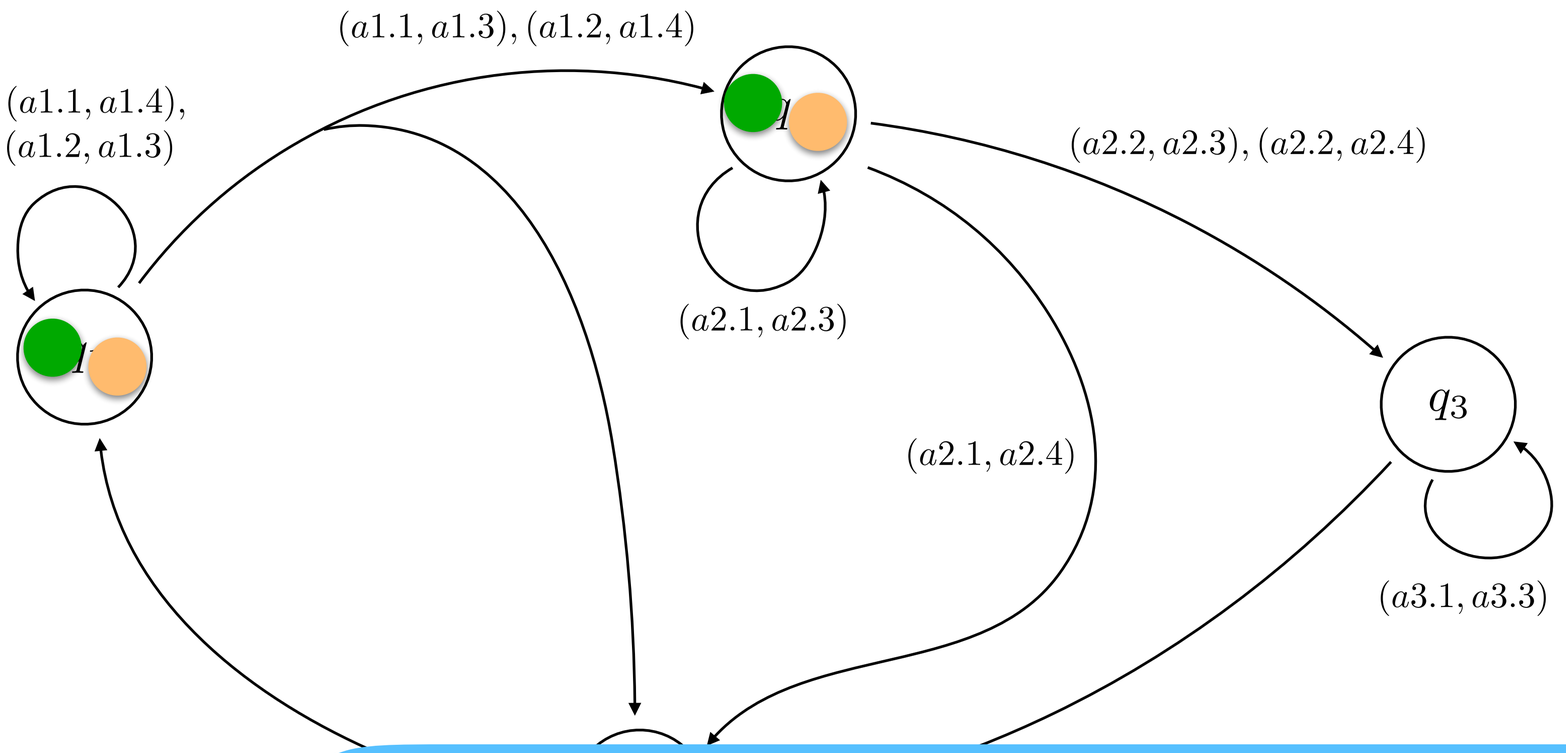
$$Q(q, a^1, a^2) \leftarrow (1-\alpha)\, Q(q, a^1, a^2) + \alpha \left(r(q, a^1, a^2) + \gamma\, \textsc{Maxmin-Q}^i(q') \right)$$

0.5 0.5 0 0.5 1 0.8 0

γ

An example

Players 1 and 2



q_1

Reward	a1.1	a1.2
a1.3	1	-1
a1.4	-1	1

q_2

Reward	a2.1	a2.2
a2.3	2	-1
a2.4	3	2

q_3

Reward	a3.1	a3.2
a3.3	2	-1
a3.4	-3	2

q_4

Reward	a4.1	a4.2
a4.3	1	-1
a4.4	-1	1

q_1

Nash Q values	a1.1	a1.2
a1.3	**0.5**	0
a1.4	0	0

q_2

Nash Q values	a2.1	a2.2
a2.3	0	0
a2.4	0	0

q_4

	a4.1	a4.2
	0	0
	0	0

$$Q(q, a^1, a^2) \leftarrow (1-\alpha)\, Q(q, a^1, a^2) + \alpha \left(r(q, a^1, a^2) + \gamma\, \textsc{Maxmin-Q}^i(q') \right)$$

0.5 0.5 0 0.5 1 0.8 0

γ