# Symmetry in Markov Decision Processes and its Implications for Single Agent and Multiagent Learning

**Martin Zinkevich** MAZ+@CS.CMU.EDU
**Tucker Balch** TRB+@CS.CMU.EDU
Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213 USA

## Abstract

This paper examines the notion of symmetry in Markov decision processes (MDPs). We define symmetry for an MDP and show how it can be exploited for more effective learning in single agent systems as well as multiagent systems and multirobot systems. We prove that if an MDP possesses a symmetry, then the optimal value function and $Q$ function are similarly symmetric and there exists a symmetric optimal policy. If an MDP is known to possess a symmetry, this knowledge can be applied to decrease the number of training examples needed for algorithms like $Q$ learning and value iteration. It can also be used to directly restrict the hypothesis space.

## 1. Introduction and Background

In this paper, we formalize the concept of symmetry in Markov decision processes (MDPs) and derive theoretical results using this formalism. We use these results to improve the performance of existing algorithms and to prove an interesting result regarding homogeneous agents.

A symmetry is a type of equivalence relation. Two states are symmetric if they have symmetric actions. Two actions are symmetric if they lead to symmetric outcomes. We show that if an MDP possesses a symmetry, then it possesses a symmetric optimal value function, a symmetric $Q$ function, symmetric optimal actions, and a symmetric optimal policy. We look at the two types of symmetry: *adherence to an equivalence relation* and *invariance under a group of transformations.* We use homogeneity in multiagent groups as an example to clarify the differences between these two types.

Single agents can exploit symmetry by reusing plans or policies for portions of their problem space (Bowling, 1999). Multiple agents can exploit symmetry in distributed tasks by using homogeneous team policies (e.g. (Balch, 2000)). This paper does not address the problem of recognizing symmetry in an agent task. This problem is being investigated by other researchers, including (McCallum, 1995) and (Bowling, 1999). We focus here on providing a formalism in which this work can be discussed and examined.

MDPs have been studied in operations research and artificial intelligence (Bellman, 1957; Filar & Vrieze, 1997; Mitchell, 1997). Reinforcement Learning is a technique for discovering a solution to an MDP by iteratively updating a value function or $Q$ function with information gained by operating inside the MDP. Kaebling, Littman, & Moore (1996) provide an excellent survey on reinforcement learning. Here we show how algorithms which are guaranteed to converge to an optimal value function or $Q$ function can be accelerated, and we provide a guideline for improving the speed of heuristic techniques.

## 2. Markov Decision Processes

Here we present the traditional definitions relating to MDPs to clarify the notation that will be needed for the definition of symmetry introduced below.

**Definition 1** *A **Markov decision process** is an ordered tuple $(\mathcal{S},\mathcal{A},T,R)$ where $\mathcal{S}$ is a set of states, $\mathcal{A}$ is a set of actions, $T$ is a transition function from $\mathcal{S}\times\mathcal{A}\times\mathcal{S}$ to $[0,1]$ where for all $s\in\mathcal{S}$ and all $a\in\mathcal{A}$, $\sum_{s'\in\mathcal{S}}T(s,a,s')=1$, and $R$ is a reward function from $\mathcal{S}\times\mathcal{A}$ to $\mathbb{R}$.*

An MDP represents a stochastic environment with one agent. At any time, the agent is in a state $s\in\mathcal{S}$. It then performs an action $a\in\mathcal{A}$. The agent receives a reward $R(s,a)$. The probability that the agent is in any state $s'\in\mathcal{S}$ at the next step is $T(s,a,s')$.

There are several types of algorithms that the agent could use to determine its action in a particular state. The agent can use a deterministic function of its current state to determine its next action, or the function could depend on time, or the agent could choose from a probability distribution over states.

**Definition 2** *A **policy** is a function $\sigma : \mathcal{S} \rightarrow \mathcal{A}$.*

If an agent executing policy $\sigma$ is in a state $s \in \mathcal{S}$, it performs action $\sigma(s)$. Thus, the probability that the agent is in state $s' \in \mathcal{S}$ in the next time step is $T(s, \sigma(s), s')$. Also, the reward is $R(s, \sigma(s))$.

Thus, the sequence of states is a Markov chain. The transition matrix for the chain is $P_{s's} = T(s, \sigma(s), s')$. If the agent is in state $s \in \mathcal{S}$ at time 0, then the probability that the agent is in state $s' \in \mathcal{S}$ at time $t$ is $(P^t)_{s's}$. Thus the expected reward at time $t$ is $\sum_{s' \in \mathcal{S}} (P^t)_{s's} R(s', \sigma(s'))$.

**Definition 3** *Define the **value function** of an MDP and a policy $\sigma$ to be a function $V_\sigma$ such that for all $s \in \mathcal{S}$:*

$$V_\sigma(s) = \sum_{t=0}^{\infty} \gamma^t \sum_{s' \in \mathcal{S}} (P^t)_{s's} R(s', \sigma(s'))$$

*for some $\gamma \in [0, 1)$.*

$V(s)$ represents a weighted sum of the expected rewards if a policy is played in an MDP. It is important to notice that

$$V_\sigma(s) = R(s, \sigma(s)) + \gamma \sum_{s' \in \mathcal{S}} T(s, \sigma(s), s') V_\sigma(s')$$

Observe that different policies will most likely result in different value functions.

**Definition 4** *An **optimal policy** $\sigma^*$ is a policy such that for all policies $\sigma$, $V_{\sigma^*}(s) \geq V_\sigma(s)$ for all $s \in \mathcal{S}$. The **optimal value function** is a value function of an optimal policy, $V^* = V_{\sigma^*}$.*

It is apparent from the definition that there is no more than one optimal value function. There can be more than one optimal policy.

**Theorem 1** *(Bellman, 1957) A value function $V^*$ is optimal if and only if it obeys **Bellman's equation**:*

$$V^*(s) = \max_{a \in \mathcal{A}} \left( R(s, a) + \gamma \sum_{s' \in \mathcal{S}} T(s, a, s') V^*(s') \right)$$

.

It is well known that there exists an optimal value function (Bellman, 1957). We provide a proof in the Appendix.

**Definition 5** *The $Q$ **function** is a function such that for all $s \in \mathcal{S}$, for all $a \in \mathcal{A}$, $Q(s, a) = R(s, a) + \gamma \sum_{s' \in \mathcal{S}} T(s, a, s') V^*(s)$. An **optimal action** for a state $s \in \mathcal{S}$ is an action $a \in \mathcal{A}$ such that $Q(s, a) = V^*(s)$.*

A strategy $\sigma$ is optimal if and only if for all states $s$ $\sigma(s)$ is an optimal action.

## 3. Symmetry in MDPs

We introduce symmetry in an MDP with an example. Consider a two dimensional pixellated image which is symmetric across the line $y = x$. That is, if one flips the image around the line, it looks the same as it did before. Another way of representing this symmetry is that the pixel $(x, y)$ is the same color as the pixel $(y, x)$. That is, the locations are "equivalent."

**Definition 6** *A **relation** $f$ is a subset of $G \times H$. For all $g \in G$, define $f(g) = \{h | (g, h) \in f\}$. An **equivalence relation** $E$ on a set $K$ is a subset of $K \times K$ obeying the following three properties for all $a, b, c \in K$: $a \in E(a)$; if $a \in E(b)$, then $b \in E(a)$; if $a \in E(b)$, and $b \in E(c)$, then $a \in E(c)$. $E(a)$ is called the **equivalence class** of $a$.*

There are other types of symmetry. Consider an image which is symmetric about the origin. In other words, if you perform a rotation of an arbitrary number of degrees, the image looks the same. Converting this into an equivalence relation, two points are equivalent if they are the same distance from the origin.

A naïve definition of symmetry would be an equivalence relation over the states of the MDP. However, this is insufficient for representing several types of symmetries. For example, consider when an agent is playing soccer, and its actions are of the form "turn ten degrees left" and "move three meters forward". In this game a line of symmetry runs lengthwise down the middle of the field. However, when you flip across this line, "left" and "right" switch, so the agent must modify his actions accordingly. So we define symmetry in the following fashion:

**Definition 7** *An **MDP symmetry** is an ordered pair $(E_\mathcal{S}, E_\mathcal{A})$ where $E_\mathcal{S}$ is an equivalence relation on $\mathcal{S}$ and $E_\mathcal{A}$ is an equivalence relation on the set $\mathcal{S} \times \mathcal{A}$.*

1. *For all $(s, s') \in E_\mathcal{S}$, for all $a \in \mathcal{A}$, there exists an $a' \in \mathcal{A}$ such that $(s', a') \in E_\mathcal{A}(s, a)$.*

2. *For all* $((s,a),(s',a')) \in E_{\mathcal{A}}$, $(s,s') \in E_{\mathcal{S}}$.

*In other words, if two states are equivalent if they have equivalent actions.*

An interpretation of $((s,a),(s',a')) \in E_{\mathcal{A}}$ is that "performing action $a$ in state $s$ is the same as performing action $a'$ in state $s'$." The act of turning eight degrees right on the left side of the soccer field should have the same effect as the act of turning eight degrees left on the right side of the soccer field. This does not mean that it ends in exactly the same position, only that the new positions where it arrives are symmetrical. It should be that a goal from the left side of the field and a goal from the right side are rewarded equally.

**Definition 8** *We define* ***symmetric*** *with respect to* $(E_{\mathcal{S}}, E_{\mathcal{A}})$ *in several contexts as follows:*

a. *A reward function is* ***symmetric*** *if for all* $((s,a),(s',a')) \in E_{\mathcal{A}}$, $R(s,a) = R(s',a')$.

b. *A transition function is* ***symmetric*** *if for all* $((s,a),(s',a')) \in E_{\mathcal{A}}$, *for all* $s'' \in \mathcal{S}$,

$$\sum_{s''' \in E_{\mathcal{S}}(s'')} T(s,a,s''') = \sum_{s''' \in E_{\mathcal{S}}(s'')} T(s',a',s''')$$

*This means that the probability of transitioning to an equivalence class is equal for both state-action pairs.*

c. *An MDP is* ***symmetric*** *if the reward function and transition function are symmetric.*

d. *A value function* $V$ *is* ***symmetric*** *if for all* $(s,s') \in E_{\mathcal{S}}$, $V(s') = V(s)$.

e. *A* $Q$ *function is* ***symmetric*** *if for all* $((s,a),(s',a')) \in E_{\mathcal{A}}$, $Q(s,a) = Q(s',a')$.

f. *The optimal actions are* ***symmetric*** *if for all* $((s,a),(s',a')) \in E_{\mathcal{A}}$, $a$ *is an optimal action in state* $s$ *implies* $a'$ *is an optimal action in state* $s'$.

g. *A policy* $\sigma$ *is* ***symmetric*** *if for all* $(s,s') \in E_{\mathcal{S}}$, $((s,\sigma(s)),(s',\sigma(s'))) \in E_{\mathcal{A}}$.

We illustrate symmetry in an MDP with an example. Consider a single agent sent to forage on a square field. A square has several lines of symmetry, thus for each state there are seven other states which are equivalent to it. However, suppose we introduce a puck in the field to be foraged. If the location of the puck is part of the state, then the associated MDP is still symmetric. However, if we restrict the puck to be in a specific location, the symmetry of the square is broken. Thus symmetry is not only dependent on the environment of the agent but also upon how that environment is represented.

**Theorem 2** *Given an MDP* $(\mathcal{S},\mathcal{A},T,R)$, *which is symmetric with respect to* $(E_{\mathcal{S}},E_{\mathcal{A}})$, *where* $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$ *and* $\mathcal{A}$ *are finite, then*

a. *the optimal value function is symmetric,*

b. *the* $Q$ *function is symmetric,*

c. *the optimal actions are symmetric, and*

d. *there exists an optimal symmetric policy.*

A proof is given in the Appendix. We proceed now to examine the implications of Theorem 2.

## 4. Accelerating Learning

How does one learn to behave optimally in an MDP? Two popular ways of deriving a policy for an MDP are to guess an optimal value function (Samuel, 1959) or $Q$ function (Watkins, 1989) and iteratively improve it. The speed of these algorithms is dependent on the number of states and actions. If one is using a value function approach and one has identified some symmetry possessed by the MDP, one can consider the MDP formed with the equivalence classes as states. For example, in soccer, the set of two points equidistant from the center line and equidistant from the goal could be considered a single state for learning.

This can improve the usefulness of each update immensely. Assume for instance an agent is stuck exploring the right side of a symmetric field. It can translate this information to the left side, so that even if it never escapes the right side, it can learn the dynamics of the whole field.

If using a $Q$ function to learn a policy, one can learn a value for each equivalence class of state-action pairs. For nondeterministic MDPs, learning the $Q$ function involves recording the number of times a state-action pair is visited, so that later visits can be weighted less. One must now record the total number of executions of all the state-action pairs in an equivalence class.

Both of these approaches enable the agent to learn a strategy for an effectively "smaller" MDP. This should improve the speed of convergence to the optimal value function or $Q$ function.

A table update method is not always the most efficient method for learning how to operate in large MDPs. A function approximator (e.g. an artificial neural network) can be used to approximate $V^*$ or $Q$. Here, the knowledge that symmetrical states and actions have the same value can be used to guide the construction of the network. If an agent is on a circle, then we can enter the distance from the center as opposed to the absolute coordinates of the agent.

# 5. Implications for Multiagent Learning

## 5.1 Multiagent Markov Decision Processes

There are a number of interesting learning problems that require us to find policies for multiple identical agents. These agents might be physical robots or software programs (Mataric, 1997; Noda, 1995; Parker, 1992). It may be convenient and appropriate to use the same program for all the agents: if we know the same policy will be effective for all agents, we only need to learn a single policy rather than one for each agent. However, can such an approach be optimal?

Another reason for addressing this issue is that it helps illustrate the differences between two types of symmetry. Heretofore, we have discussed a symmetry based on *adherence to an equivalence relation*. In the following sections, we present an alternative: *invariance under a group of transformations*. Here, we use group in the strict sense: a set of functions containing the identity and closed under composition and inversion. One reason we introduced the equivalence relation type of symmetry earlier is that an invariance under a group of transformations can be converted to an adherence to an equivalence relation, although the latter may be less restrictive.

In the following discussion we assume the agents (robotic or software) are homogeneous, but that they may utilize heterogeneous policies. By "homogeneous agents" we mean they all share the same sensing and acting capabilities. In the case of robotic agents, this means the robots are mechanically and electrically identical. An MDP is typically used to represent the interactions of a single agent. However, if multiple agents are fully collaborative and have complete knowledge of the environment, they may be treated as a single agent acting in an MDP. If each agent is capable of computing its portion of the joint deterministic optimal policy, then it may simply execute its portion. If the joint policy is nondeterministic, then this may not be the case. Because we are going to be considering homogeneous agents, we will restrict the state sets and action sets of all the agents to be identical. Now we introduce:

**Definition 9** *A* ***Multiagent Markov Decision Process*** *(MMDP) with $n$ agents is an MDP $(\mathcal{S}, \mathcal{A}, T, R)$ where $\mathcal{S} \subseteq (\mathcal{S}_{agent})^n$ for some set $\mathcal{S}_{agent}$ and $\mathcal{A} = (\mathcal{A}_{agent})^n$ for some set $\mathcal{A}_{agent}$, where $\mathcal{S}_{agent}$ is considered the state space of a single agent, and $\mathcal{A}_{agent}$ is considered the set of actions of a single agent.*

The purpose of this representation is for ease of theoretical analysis. In practical applications, augmenting the vector with additional state of the environment and perhaps the state of other agents would be appropriate. The theory extends to such cases.

On the other hand, for some cases, this representation is insufficiently restrictive. If our agents are physical robots, then they will not be in the same physical position, and their exact position is contained in their state, then they will not be in the same state. This gives us an additional restriction on the nature of the state space.

**Definition 10** *An MMDP has* ***agents in distinct states*** *if for all elements $s \in \mathcal{S}$, for all i,j where $i \neq j$, $s_i \neq s_j$.*

Note that for an MMDP, a policy is a function from $\mathcal{S} \subseteq (\mathcal{S}_{agent})^n$ to $(\mathcal{A}_{agent})^n$. This allows each robot's action to depend upon the states of the others.

## 5.2 Homogeneity

How can one say that two robots are physically identical? One way is to say that there are separate, but identical transition functions for each robot depending only on that robot's state. However, this is overly restrictive, because it would not allow for interaction of any kind between robots.

Suppose a group of robots is performing a task. Robot 1 is in position $s_1$, performs action $a_1$, and ends up in state $s'_1$. Simultaneously, robot 2 is in position $s_2$, performs action $a_2$, and ends up in state $s'_2$. If they robots are physically identical, then it should be the case that if you place robot 1 in position $s_2$ and robot 2 in position $s_1$, and robot 1 performs action $a_2$ and robot 2 performs action $a_1$, then robot 1 should end up in state $s'_2$ and robot 2 should end up in state $s'_1$. In general, if we permute the states of the robots and their actions, then the results should be the same. We formalize this concept below:

**Definition 11** *A* ***permutation*** *of a state is a permutation of the elements. If $P$ is a permutation of a set of vectors $V^n$, then there exists a bijection $g : \{1 \ldots n\} \to \{1 \ldots n\}$, such that for all $\vec{v} \in V^n$, for all $i$, $P(\vec{v})_i = v_{g(i)}$. The inverse of a permutation $P$ is the permutation $P^{-1}$ such that for all $\vec{v} \in V^n$, $\vec{v} = P^{-1}(P(\vec{v}))$. The inverse of the inverse of $P$ is $P$.*

**Definition 12** *An* ***equivalence homogeneity*** *for an MMDP is an ordered pair $(H_{\mathcal{S}}, H_{\mathcal{A}})$ where $H_{\mathcal{S}}$ and $H_{\mathcal{A}}$ are defined as follows:*

1. *$(s, s') \in H_{\mathcal{S}}$ if and only if there exists a permuta-*

*tion $P$ such that $s' = P(s)$.*

2. *$((s,a),(s',a')) \in H_{\mathcal{A}}$ if and only if there exists a permutation $P$ such that $s' = P(s)$ and $a' = P(a)$. Thus, if you use any permutation to swap the robots and their actions similarly, the new configuration of robots and actions are equivalent to the old.*

We could choose to define a homogeneous MMDP to be an MMDP possessing this symmetry, but that would allow the agents to "swap" states during a transition. In other words, in soccer, if agent $A$ is on the left side of the field, and agent $B$ is on the right side of the field, a transition which is symmetric with respect to homogeneity could place $B$ on the left and $A$ on the right. Therefore, we provide a more restrictive definition.

**Definition 13** *An MMDP is **functionally homogeneous** if for all permutations $P$, all states $s, s' \in \mathcal{S}$, and all actions $a \in \mathcal{A}$, $T(P(s), P(a), P(s')) = T(s, a, s')$ and $R(P(s), P(a)) = R(s, a)$.*

Observe that a functionally homogeneous MMDP is symmetric with respect to equivalence homogeneity. A functionally homogeneous policy conforms to each agents' determining its action separately using the same algorithm.

**Definition 14** *A policy $\sigma$ is **functionally homogeneous** if for all permutations $P$, for all states $s \in S$, $\sigma(P(s)) = P(\sigma(s))$.*

Observe that this definition is the same as the following: for all $((s,a),(s',a')) \in H_{\mathcal{A}}$, if $\sigma(s) = a$, then $\sigma(s') = a'$. For an equivalence homogeneity it is required that if two states are equivalent, the actions performed in these states are equivalent. A functionally homogeneous policy requires that if two state-action pairs are equivalent and one is part of the optimal policy, then the other must be. But what if two actions in the same state are equivalent? The following examples clarify this issue.

### 5.3 A Counterexample

It might seem that all functionally homogeneous MMDPs have at least one optimal functionally homogeneous policy. In fact, this is not the case.

Consider the following situation: Wacko Foods has two stores in Fruitytown. Each day, each store can sell either apples or oranges, but not both. If one store sells apples and the other sells oranges, they sell them all and Wacko Foods makes ten dollars. If both stores sell apples or both stores sell oranges, they sell half and Wacko Foods makes five dollars. If we defines this as an MMDP with one state, it is homogeneous. However, there exists no optimal policy which involves the managers doing the same thing! This is an example of a functionally homogeneous MMDP without agents in distinct states. Other examples of MMDPs without agents in distinct states might be software agents managing stock portfolios, or agents managing purchasing decisions for a company, or any other cerebral agent.

When does one know that the agents are in distinct states? If the agents are **physical robots**, where their state contains their exact position, and these positions cannot overlap, then they would be guaranteed to always be in distinct states. Examples include soccer players, soldiers in the battlefield, etc.

However, one could formulate the MMDP of the stores in Fruitytown in a different way. The exact location of each manager could be included in the state. Suppose that one store is on Cherry Boulevard and the other is on Kiwi Drive. In this situation, the manager in the store on Cherry Boulevard could sell apples, and the manager in the store on Kiwi Drive could sell oranges. But it makes no sense to encode this information into the state, because it has no real bearing on the problem.

### 5.4 Optimal Homogeneous Policies

**Lemma 1** *In an MMDP with agents in distinct states, for all $s \in \mathcal{S}$, for all $a, a' \in \mathcal{A}$, $((s,a),(s,a')) \in H_{\mathcal{A}}$ implies $a = a'$.*

Proof: If $((s,a),(s,a')) \in H_{\mathcal{A}}$, then there exists a permutation $P$ such that $P(s) = s$ and $P(a) = a'$. If agents are in distinct states, then for all $s \in \mathcal{S}$, the elements are unique. Therefore the only permutation $P$ for which $s = P(s)$ is the identity. Therefore $a = a'$. ∎

**Lemma 2** *If for all $s \in \mathcal{S}$, for all $a, a' \in \mathcal{A}$, $((s,a),(s,a')) \in H_{\mathcal{A}}$ implies $a = a'$, then all policies which are symmetric with respect to homogeneity are functionally homogeneous.*

Proof: Assume $\sigma$ is a policy which is symmetric with respect to homogeneity. For all $((s,\sigma(s)),(s',a')) \in H_{\mathcal{A}}$, if $s = s'$, then $a' = a$. Thus $\sigma(s') = a'$.If $s \neq s'$, then first observe that $(s,s') \in H_{\mathcal{S}}$. Thus $((s,\sigma(s)),(s',\sigma(s'))) \in H_{\mathcal{A}}$. Therefore, $((s',\sigma(s')),(s',a')) \in H_{\mathcal{A}}$. Thus $\sigma(s') = a'$, implying that the policy is homogeneous. ∎

**Theorem 3** *A functionally homogeneous MMDP* *with agents in distinct states possesses a homogeneous optimal policy.*

Proof: As stated before in Section 5.2, a functionally homogeneous MMDP is symmetric with respect to homogeneity. Therefore, there exists an optimal policy which is symmetric with respect to equivalence homogeneity. From Lemma 1 and Lemma 2, this strategy is functionally homogeneous. ∎

## 6. Conclusions and Future Work

In this paper, we introduce a definition of symmetry. We show that a symmetric MDP has a symmetric optimal value function, a symmetric $Q$ function, symmetric optimal actions, and a symmetric optimal policy. Symmetry in an MDP can often be easily recognized. Symmetries in a playing field, identical objects, and identical agents are often explicit or apparent in the statement of a problem. This paper presents a theoretical justification for exploiting these types of symmetry to accelerate learning for single and multiagent systems.

We introduce two types of symmetry in this paper: equivalence symmetry and functional symmetry. An equivalence symmetry is in some ways more fundamental in that an MDP which has an equivalence symmetry has a symmetric optimal policy. However, some types of symmetry cannot be represented by an equivalence symmetry, so the more restrictive functional symmetry is quite useful for proving properties like homogeneity.

The concept of symmetry can be taken further than has been done in this paper. For symmetric MDPs which are invariant under a group of functions and have an infinite number of states or actions, there exists a symmetric optimal policy. At present, we also have a proof regarding symmetric stochastic games in preparation for publication.

The proof technique in this paper is straightforward and can probably be extended to several other types of systems. MDPs with an infinite number of states, hidden Markov models, partially observable MDPs, partially observable stochastic games, and other games in economics would have similar properties.

## Acknowledgements

We would like to thank Avrim Blum and Pat Riley for helpful discussions on this paper. This material is based upon work supported under a National Science Foundation Graduate Research Fellowship. Any opinion, findings, conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation.

## References

Balch, T. (2000), Heirarchic Social Entropy: An Information Theoretic Measure of Robot Group Diversity. *Autonomous Robots, 8(3).*

Bellman, R. E. (1957). *Dynamic Programming.* Princeton, NJ: Princeton University Press.

Border, K. C. (1985) *Fixed-point theorems with applications to economics and game theory.* New York: Cambridge University Press.

Bowling, M. H. & Veloso, M. M. (1999) Bounding the suboptimality of reusing subproblems, *Proceedings of IJCAI-99*, (pp. 1340-1345). New York: Morgan Kaufmann.

Filar, J. & Vrieze, K. (1997) *Competitive Markov decision processes.* New York: Springer Verlag.

Kaebling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement Learning: A survey. *Journal of AI Research, 4*, (pp. 237–285).

Matarić, Maja (1997). Reinforcement Learning in the Multi-Robot Domain. *Autonomous Robots, 4(1)*, (pp. 73–83).

McCallum, A. K. (1995). *Reinforcement Learning with Selective Perception and Hidden State.* Ph. D. dissertation, University of Rochester, Rochester, New York.

Mitchell, T. M. (1997). *Machine Learning.* New York: McGraw-Hill.

Noda, I. (1995). Soccer Server: a simulator for Robocup, *JSAI AI-Symposium 95: Special Session on RoboCup.*

Parker, L. (1992). Adaptive Action Selection for Cooperative Agent Teams, *Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, (pp. 442–450).

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development, 3*, (pp. 211–229).

Watkins, C. (1989). *Learning from delayed rewards.* Ph.D. dissertation, King's College, Cambridge, England.

## Appendix

In order to prove Theorem 2, we begin with Brouwer's Theorem. Then we develop the concept of symmetry in Euclidean spaces, and prove a result which will be used as a foundation of the proof of Theorem 2.

**Definition 15** *A set $K \subseteq \mathbb{R}^n$ is **bounded** if there exists an $N \in \mathbb{R}$ such that for all $\vec{x} \in K$, $d(\vec{x}, \vec{0}) < N$. A set $K \subseteq \mathbb{R}^n$ is **compact** if it is closed and bounded.*

**Theorem 4** ***Brouwer's Theorem*** *Given a nonempty, compact, convex set $K \subseteq \mathbb{R}^n$, a continuous function $f : K \to K$, there exists a vector $\vec{x} \in K$ such that $f(\vec{x}) = \vec{x}$.*

A proof can be found in (Border, 1985).

**Definition 16** *If $E$ is an equivalence relation on $\{1, \dots, n\}$, then a vector $\vec{x} \in \mathbb{R}^n$ is **symmetric** with respect to $E$ if for all $(i,j) \in E$, $x_i = x_j$. Suppose that $f$ is a function from $B \subseteq \mathbb{R}^n$ to $C \subseteq \mathbb{R}^n$. $f$ is **symmetric** with respect to $E$ if for all symmetric $\vec{x} \in B$, $f(\vec{x})$ is symmetric.*

**Theorem 5** *The set of all symmetric vectors in $\mathbb{R}^n$ with respect to an equivalence relation $E$ on $\{1 \dots n\}$ is a linear subspace, and therefore closed and convex.*

Proof: Define $V$ to be the set of all symmetric vectors in $\mathbb{R}^n$ with respect to $E$. A subset of $\mathbb{R}^n$ is a linear subspace if it is closed under addition, scalar multiplication, and contains the zero vector. $V$ is closed under addition, because if $\vec{x}$ and $\vec{y}$ are symmetric, then for all $(i,j) \in E$, $x_i = x_j$, and $y_i = y_j$, hence $x_i + y_i = x_j + y_j$. If $\vec{x}$ is symmetric and $\lambda$ is a scalar, then for all $(i,j) \in E$, $x_i = x_j$, and $\lambda x_i = \lambda x_j$. For $\vec{0}$, for all $(i,j) \in E$, $0_i = 0 = 0_j$, implying $\vec{0}$ is symmetric. Thus the set of all symmetric vectors is a linear subspace. ∎

Here we present a theory of symmetry which will be at the core of the proof to Theorem 2.a.

**Theorem 6** *Given $B \in \mathbb{R}^n$ is a compact, convex set, a continuous, symmetric function $f : B \to B$, $E$ is an equivalence relation on $\{1 \dots n\}$. If there exists a symmetric vector in $B$ with respect to $E$, then there exists a symmetric vector $\vec{x} \in B$ with respect to $E$ such that $f(\vec{x}) = \vec{x}$.*

Proof: Define $B'$ to be the set of all symmetric vectors in $\mathbb{R}^n$ with respect to $E$. $B'$ is closed and convex by Theorem 5. Consider the set $B'' = B' \cap B$. Since $B'$ and $B$ are closed, $B''$ is closed, and since $B'$ and $B$ are convex, $B''$ is convex. Since $B$ is bounded, $B''$ is bounded, and because $B$ contains a symmetric vector, $B''$ is nonempty. If we restrict $f$ to $B''$, $f$ is still continuous and because $f$ is symmetric the new codomain will be a subset of $B''$. Hence, Brouwer's Theorem applies to $f$ on $B''$, and there exists an $\vec{x} \in B''$ such that $f(\vec{x}) = \vec{x}$. $\vec{x}$ is a symmetric vector in $B$. ∎

Before presenting the proof of Theorem 2, we present a proof of the existence of an optimal value function, originally proved by Bellman (1957). We show this proof because it shares many elements with the proof of Theorem 2, and will help elucidate that argument.

**Theorem 7** *Given an MDP $(\mathcal{S}, \mathcal{A}, T, R)$, where $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$ and $\mathcal{A}$ are finite, then there exists a value function $V^* : \mathcal{S} \to \mathbb{R}$ satisfying Bellman's equation.*

Proof: Bellman's equation may or may not have a solution because it has a value $V^*$ on both sides of the equation. So, let us slightly modify Bellman's equation by introducing $U : \mathbb{R}^{|\mathcal{S}|} \to \mathbb{R}^{|\mathcal{S}|}$, replacing $V^*(s_j)$ on the left side of Bellman's equation with $U(\vec{x})_j$, and on the right side with $x_j$.

$$U(\vec{x})_i = \max_{a \in \mathcal{A}} \left( R(s_i, a) + \gamma \sum_{j=1}^{n} T(s_i, a, s_j) x_j \right)$$

Observe that if $U(\vec{x}) = \vec{x}$, then $V^*(s_j) = x_j$ would be a solution to Bellman's equation.

Observe that the value in any state will be bounded above by the case where the agent receives the maximum value at every step, and bounded below by the case where it receives the minimum value. Define $R_{max} = \max_{s \in \mathcal{S}, a \in \mathcal{A}} R(s,a)$, $R_{min} = \min_{s \in \mathcal{S}, a \in \mathcal{A}} R(s,a)$. Define the set $K = [\frac{1}{1-\gamma} R_{min}, \frac{1}{1-\gamma} R_{max}]^{|\mathcal{S}|}$. Observe that $K$ is a hypercube, both compact and convex.

Also, for all $\vec{x} \in K$, $U(\vec{x}) \in K$. Observe that:

$$U(\vec{x})_i \leq \max_{a \in \mathcal{A}} \left( R_{max} + \gamma \sum_{j=1}^{n} T(s_i, a, s_j) \frac{1}{1-\gamma} R_{max} \right)$$

$$U(\vec{x})_i \leq \max_{a \in \mathcal{A}} \left( R_{max} + \frac{\gamma}{1-\gamma} R_{max} \sum_{j=1}^{n} T(s_i, a, s_j) \right)$$

By the definition of an MDP, $\sum_{j=1}^{n} T(s_i, a, s_j) = 1$, so:

$$U(\vec{x})_i \leq \max_{a \in \mathcal{A}} \left( R_{max} + \frac{\gamma}{1-\gamma} R_{max} \right)$$

$$U(\vec{x})_i \leq \frac{1}{1-\gamma} R_{max}$$

Similarly, $U(\vec{x})_i \geq \frac{1}{1-\gamma} R_{min}$.

$U$ is continuous, because it is formed by the composition of concatenation, maximization, multiplication and addition, which are all continuous functions. Therefore, $K$ is a compact, convex set and $U$ is a continuous function from $K$ to $K$, and hence by Brouwer's Theorem there exists a fixed point $\vec{x}$ such that $U(\vec{x}) = \vec{x}$. Therefore there exists a function $V^*$ satisfying Bellman's equation. ∎

Proof of Theorem 2.a: Consider the symmetry $F = \{(i,j) \in \{1 \ldots n\}^2 : (s_i, s_j) \in E_{\mathcal{S}}\}$. Define $K$ and $U$ as before. Observe that a vector in $K$ is symmetric with respect to $F$ if and only it represents a symmetric value function. Also, $(R_{min} \ldots R_{min})$ is a symmetric vector in $K$. Observe that if $U$ is symmetric, then by Theorem 6, there exists a symmetric value function in $K$ satisfying Bellman's equation, implying that the optimal value function is symmetric. Now we will prove that $U$ is a symmetric function with respect to $F$. For all $(i,k) \in F$:

$$U(\vec{x})_i = \max_{a \in \mathcal{A}} \left( R(s_i, a) + \gamma \sum_{j=1}^{n} T(s_i, a, s_j) x_j \right)$$

Choose an $a' \in \mathcal{A}$ that maximizes $R(s_i, a') + \gamma \sum_{j=1}^{n} T(s_i, a', s_j) x_j$. Note that:

$$U(\vec{x})_i = R(s_i, a') + \gamma \sum_{j=1}^{n} T(s_i, a', s_j) x_j$$

Since $(i,k) \in F$, $(s_i, s_k) \in E_{\mathcal{S}}$. Thus, there exists a $a'' \in \mathcal{A}$ such that $((s_i, a'), (s_k, a'')) \in E_{\mathcal{A}}$. Since the reward function is symmetric:

$$U(\vec{x})_i = R(s_k, a'') + \gamma \sum_{j=1}^{n} T(s_i, a', s_j) x_j$$

Define $P = \{F(i) | i \in \{1 \ldots n\}\}$, the set of equivalence classes of states represented as indices. This set is finite, so it can be represented as $\{P_1, P_2, \ldots, P_m\}$. Observe this is a partition, so:

$$U(\vec{x})_i = R(s_k, a'') + \gamma \sum_{p=1}^{m} \sum_{j \in P_p} T(s_i, a', s_j) x_j$$

For each $P_p$, choose some $j \in P_p$ and define $v_p = x_j$. Observe that for all $l \in P_p$, $x_l = x_j$ and hence $x_l = v_p$.

$$U(\vec{x})_i = R(s_k, a'') + \gamma \sum_{p=1}^{m} \sum_{j \in P_p} T(s_i, a', s_j) v_p$$

Now we can bring this value out of the innermost sum.

$$U(\vec{x})_i = R(s_k, a'') + \gamma \sum_{p=1}^{m} v_p \sum_{j \in P_p} T(s_i, a', s_j)$$

Because the transition function is symmetric:

$$U(\vec{x})_i = R(s_k, a'') + \gamma \sum_{p=1}^{m} v_p \sum_{j \in P_p} T(s_k, a'', s_j)$$

Afterwards, we unwrap these manipulations:

$$U(\vec{x})_i = R(s_k, a'') + \gamma \sum_{j=1}^{n} T(s_k, a'', s_j) x_j$$

$$U(\vec{x})_i \leq max_{a \in \mathcal{A}} \left( R(s_k, a) + \gamma \sum_{j=1}^{n} T(s_k, a, s_j) x_j \right)$$

$$U(\vec{x})_i \leq U(\vec{x})_k$$

By reversing $i$ and $k$, we can prove $U(\vec{x})_k \leq U(\vec{x})_i$. Thus, $U(\vec{x})_i = U(\vec{x})_k$. This implies that if $\vec{x}$ is symmetric, $U(\vec{x})$ is symmetric, which means we can use Theorem 6 to prove there is a symmetric optimal value function. Because the optimal value function is unique, the optimal value function is symmetric. ∎

Proof of Theorem 2.b: For all $(s,a), (s',a') \in \mathcal{S} \times \mathcal{A}$ such that $(s',a') \in E_{\mathcal{A}}(s,a)$:

$$Q(s,a) = R(s,a) + \gamma \sum_{s'' \in \mathcal{S}} T(s,a,s'') V^*(s'')$$

Observe that $V^*$ is symmetric in the same way that $\vec{x}$ was. Thus, we can use the same technique we used in Theorem 2.a. ∎

Proof of Theorem 2.c: Now we will prove that the optimal actions sets are symmetric. Take an $((s,a),(s',a')) \in E_{\mathcal{A}}$ such that $a$ is an optimal action for the state $s$. Thus $(s,s') \in E_{\mathcal{S}}$. Therefore, $Q(s',a') = Q(s,a) = V^*(s) = V^*(s')$, so $a'$ is an optimal action for the state $s'$. ∎

Proof of Theorem 2.d: Finally, we shall prove there exists a symmetric optimal policy. For each $P_i$, choose a representative $s_i \in P_i$. For each $s_i$, choose an optimal action $a_i \in \mathcal{A}$ for $s_i$. For each $s \in P_i$, choose an $a \in \mathcal{A}$ such that $((s,a),(s_i,a_i)) \in E_{\mathcal{A}}$, and set $\sigma(s) = a$. Observe that, by definition, $\sigma$ is symmetric. Since the set of optimal actions is symmetric, $\sigma$ is optimal. ∎