**T.C.**
**BOLU ABANT İZZET BAYSAL ÜNİVERSİTESİ**
**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**
**BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

# SOSYAL ÖĞRENME DESTEKLİ DERİN PEKİŞTİRMELİ ÖĞRENME

**YÜKSEK LİSANS TEZİ**

**CEREN GÜLEN**

**TEZ DANIŞMANI**
**Doç. Dr. Mehmet Dinçer ERBAŞ**

**BOLU, ARALIK - 2023**

# KABUL VE ONAY SAYFASI

**Ceren GÜLEN** tarafından hazırlanan "**SOSYAL ÖĞRENME DESTEKLİ DERİN PEKİŞTİRMELİ ÖĞRENME**" adlı tez çalışması jürimiz tarafından Bilgisayar Mühendisliği Anabilim Dalı'nda Yüksek Lisans Tezi olarak oy birliği kabul edilmiştir. 27/12/2023

| **Jüri Üyeleri** | **İmza** |
|---|---|
| Danışman<br>Doç. Dr. Mehmet Dinçer ERBAŞ<br>Bolu Abant İzzet Baysal Üniversitesi | ..... |
| Üye<br>Dr. Öğr. Üyesi Musa MİLLİ<br>Milli Savunma Üniversitesi Deniz Harp Okulu | ..... |
| Üye<br>Dr. Öğr. Üyesi İsmail Hakkı PARLAK<br>Bolu Abant İzzet Baysal Üniversitesi | ..... |

**Lisansüstü Eğitim Enstitüsü Onayı**

**Prof. Dr. İbrahim KÜRTÜL**
**Lisansüstü Eğitim Enstitüsü Müdürü**

# ETİK BEYAN

Bolu Abant İzzet Baysal Üniversitesi, Lisansüstü Eğitim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu bildirir,

aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Teze ilişkin Turnitin adlı programında enstitü müdürlüğünce belirlenen filtrelemeler uygulanarak alınmış olan benzerlik raporuna göre, tezin benzerlik oranı ……'u geçmemektedir.

……..

**CEREN GÜLEN**

# ÖZET

**SOSYAL ÖĞRENME DESTEKLİ DERİN PEKİŞTİRMELİ ÖĞRENME**
**YÜKSEK LİSANS TEZİ**
**CEREN GÜLEN**
**BOLU ABANT İZZET BAYSAL ÜNİVERSİTESİ**
**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**
**BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**(TEZ DANIŞMANI: DOÇ. DR. MEHMET DİNÇER ERBAŞ)**

**BOLU, ARALIK - 2023**
**XI + 63**

Sosyal öğrenme, bireylerin çevrelerindeki diğer bireylerden bilgi edinme ve bu bilgileri kullanarak davranışlarını şekillendirme sürecidir. Bu tez çalışması, içsel geri bildirim kullanarak doğadaki kopyalama mekanizmasından esinlenerek, sosyal öğrenmeyi destekleyen özgün bir Derin Pekiştirmeli Öğrenme algoritması sunmaktadır. Grup içindeki bireylerin öğrenme hızını ve performansını artırmak amaçlanmaktadır. İyi bilinen bir pekiştirmeli öğrenme algoritması olan derin Q öğrenme kullanılmıştır. Pekiştirmeli öğrenme ile taklit kullanan diğer araştırmalara kıyasla, yöntemimiz öğrenen etmenlerin çift katmanlı bir kontrol sistemi tarafından kontrol edildiği, farklı hafıza türlerinin incelendiği, kopyalama ve uygulama davranışlarının doğaya uygun olarak derin sinirsel ağın gerçekleştirebileceği aksiyonlar arasından tanımlanarak dinamik olarak akıllı etmen tarafından kontrol edildiği özgün bir yaklaşımdır. Yaklaşımımız simülasyon ortamında uygulanmıştır. Simülasyon deneyleri deney sonuçları, öğrenme hızının arttığını ve kopyalama emrinin aksiyon olarak verilmesinin olumlu etki gösterdiği görülmüştür.

**ANAHTAR KELİMELER:** Arama / Toplama, Çok Etmenli Sistem, Derin Pekiştirmeli Öğrenme, Hafıza Yöntemleri, Sosyal Öğrenme, Sürü Robotlar, Taklit

# ABSTRACT

**SOCIAL LEARNING SUPPORTED DEEP REINFORCEMENT LEARNING**
**MSC THESIS**
**CEREN GÜLEN**
**BOLU ABANT IZZET BAYSAL UNIVERSITY**
**INSTITUTE OF GRADUATE STUDIES**
**DEPARTMENT OF COMPUTER ENGINEERING**
**(SUPERVISOR: ASSOC. DOC. MEHMET DİNÇER ERBAŞ)**

**BOLU, DECEMBER 2023**
**XI + 63**

Social learning is the process by which individuals acquire information from other individuals in their environment and use this information to shape their behavior. This thesis presents a novel Deep Reinforcement Learning algorithm that supports social learning inspired by the copying mechanism in nature using intrinsic feedback. The aim is to increase the learning speed and performance of individuals within a group. Deep Q learning, a well-known reinforcement learning algorithm, is used. Compared to other research using imitation with reinforcement learning, our approach is novel in that the learning agents are controlled by a dual-layer control system, different types of memory are examined, and copying and application behaviors are dynamically controlled by the intelligent agent by defining the actions that the deep neural network can perform according to its nature. Our approach is implemented in a simulation environment. The results of the simulation experiments show that the learning speed increases and the copy command as an action has a positive effect.

**KEYWORDS:** Foraging, Multi-Agent System, Deep Reinforcement Learning, Memory Methods, Social Learning, Swarm Robots, Imitation

# İÇİNDEKİLER

**Sayfa**

# ŞEKİL LİSTESİ

**Sayfa**

# KISALTMA VE SEMBOLLER LİSTESİ

**AI** : Yapay Zekâ (Artificial Intelligence)

**DIRL** : Derin Kopyalama ile Pekiştirmeli Öğrenme (Deep Imitation Reinforcement Learning)

**DL** : Derin Öğrenme (Deep Learning)

**DQN** : Derin Q Ağları (Deep Q-Networks)

**DRL** : Derin Pekiştirmeli Öğrenme (Deep Reinforcement Learning)

**GAIL** : Üretken Çekişmeli Kopyalama ile Öğrenme (Generative Adversarial Imitation Learning)

**GAN** : Üretken Çekişmeli Ağlar (Generative Adversarial Networks)

**IL** :Taklit Öğrenme (Imitation Learning)

**IRL** :Ters Pekiştirmeli Öğrenme (Inverse Reinforcement Learning)

**MDP** : Markov Karar Süreci (Markov Decision Process)

**RL** : Pekiştirmeli Öğrenme (Reinforcement Learning)

**SDRL** : Sosyal Öğrenme Destekli Derin Pekiştirmeli Öğrenme (Social Learning Supported Deep Reinforcement Learning)

**SL** : Denetimli Öğrenme (Supervised Learning)

**SÖ** : Sosyal Öğrenme (Social Learning)

**YSA** : Yapay Sinir Ağları (Artificial Neural Network)

**Q** : Eylem-Değer Fonksiyonu (Action-Value Function)

V : Durum-Değer Fonksiyonu (State-Value Function)

$\pi$ : Politika (Policy)

$\mathbf{a_t}$ : t Anında Seçilen Eylem (Action)

$\mathbf{r_t}$ : t Anındaki Ödül (Reward)

$\mathbf{s_t}$ : t Anındaki Durum (State)

$\mathbf{s_{t+1}}$ : t Anında a Hareketi ile Geçilen Yeni Durum (New State)

$\boldsymbol{\alpha}$ : Öğrenme Oranı (Learning Rate)

$\boldsymbol{\gamma}$ : İndirim Faktörü (Discount Factor)

$\boldsymbol{\varepsilon}$ : Epsilon

# TEŞEKKÜR

Lisansüstü eğitimimde, gerek almış olduğum dersler, gerekse tez çalışmalarım boyunca vermiş olduğu destekler, bana kazandırmış olduğu tecrübeler ve bilgiler için danışmanım Doç. Dr. Mehmet Dinçer ERBAŞ`a en içten dileklerimle teşekkür ederim.

Özellikle eğitimim konusunda hiç bir fedakârlıktan kaçınmayan; babam Durmuş GÜLEN, annem Sevgül GÜLEN ve ablam Çağla GÜLEN GÖKSU'a sonsuz teşekkürlerimi sunarım.

Bu tez çalışması, Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) tarafından 122E642 numaralı proje ile desteklenmiştir.

Ceren GÜLEN

# 1. GİRİŞ

Bu bölümde literatür taraması, tezin amacı, kapsamı, hipotezi ve tezde uygulanan yöntemler ile ilgili genel bilgiler verilmektedir.

## 1.1 Literatür Taraması

Pekiştirmeli öğrenme (RL) öğrenme problemlerinin çözümünde yapay bir etmenin çevresi ile arasındaki etkileşimleri kullanan, çevresel durumları maksimize edilmesi amaçlanan bir ödül değerine eşleyen, deneme/yanılma tabanlı bir bireysel öğrenme algoritmasıdır (Sutton & Barto, 2018). Etmenlerin gerçekleştirebilecekleri eylemler ölçüsünde çevreleriyle olan etkileşimlerinden yararlanır ve seçtiği eylemler sayesinde öğrenen etmenin optimal şekilde hedefine ulaşmasını sağlar. Bu yönüyle bakıldığında çevre ile etkileşim, olası eylemler ve hedef, pekiştirmeli öğrenmenin üç ana parçasıdır. Bu bilgiler doğrultusunda RL kullanan akıllı etmen, keşif/faydalanma dengesinden yararlanır. Böylece anlık ödülü maksimize etmeye çalışırken durum-eylem çiftleri üzerinden yeni ve daha verimli davranışları keşfetmeye çalışır. Son yıllarda RL ve farklı versiyonları, kontrol (Adam vd., 2012), veri madenciliği (Kheradmandian & Rahmati, 2009), ticaret ve finans (Charpentier vd., 2020), doğal dil işleme (Luketina vd., 2019) gibi birçok alanda kullanılmıştır. Ayrıca IoT temelli e-sağlık sistemlerinde kullanılan kablosuz ağlar için kanal erişim problemini ele almış ve bir Q-öğrenme (Q-learning) tabanlı yaklaşım sunulmuştur (Ali vd., 2019). Bu çalışmada öğrenme etmeni, çevresindeki kablosuz ağ durumlarına bağlı olarak, Q-matrisindeki en iyi eylemi seçmektedir. Bu yaklaşım sayesinde, ağdaki cihazların birbirleriyle çekişme yaşamasını önleyerek daha yüksek verimlilik sağlamaktadır. Siber güvenlik alanında RL algoritmalarını kullanan bir yapay zekâ etmeni geliştirilmiştir (Erdődi vd., 2021). Bu etmen, SQL enjeksiyonu açıklarını bulmak ve savunmalarını öğrenmek için tasarlanmıştır. Deneyler, önerilen yöntemin, manuel yöntemlere göre daha yüksek bir doğruluk oranı sağladığını göstermiştir. Bu çalışma, siber güvenlik alanında pekiştirmeli öğrenmenin nasıl kullanılabileceğine dair bir örnek sunmaktadır ve saldırganların zayıf noktaları keşfetmek için RL algoritmalarından yararlanılabileceğini göstermektedir. Endüstri 4.0 için öğrenme temelli otomasyon yöntemleri üzerine robotların öğrenmesi için bir yöntem önerilmektedir (Chen vd., 2021). Bu yöntem, görevleri modüller halinde ayırarak her bir modül için ayrı bir

RL modeli kullanmayı önerir. Otomatik olarak veri toplayarak, modeli güncelleyerek ve farklı görevler için yeniden kullanarak, endüstriyel ortamlarda adaptasyonu kolaylaştırmak için bir sürekli öğrenme yaklaşımı benimsemektedir. Robotlar üzerinde yapılan etkileşimli pekiştirmeli öğrenmenin insan rehberliği ve durum uzayı boyutunun performans üzerindeki etkisini incelenmektedir (Suay & Chernova, 2011). Makalede, bir insan rehberinin, bir robotun öğrenme performansını artırabileceği ve durum uzayı boyutunun arttıkça öğrenme performansının düştüğü gösterilmektedir.

Özellikle klasik RL metotlarının, eylem ve durum uzayı sınırlı ve kesikli olan çevrelerde kullanılabilir olmakla birlikte, bu özellikleri devamlı olan geniş, hatta sonsuz, durum-eylem uzayına sahip çevrelerde uygulanabilirliği azalmaktadır. Bu tür çevrelere uygulanabilirliği artırmak için farklı RL yöntemleri geliştirilmiştir (Örneğin (Schulman vd., 2015) ve (Babaeizadeh vd., 2016)). Son yıllarda, yapay sinirsel ağlar ve kullanım alanlarına olan ilginin artışı ile birlikte birçok farklı araştırma konusunda Derin Öğrenmenin (DL) kullanıldığı görülmektedir (Goodfellow vd., 2016). Derin öğrenmenin, pekiştirmeli öğrenmenin önemli bir kısıtlaması olan çok boyutlu durum-eylem uzayına sahip problemlere ölçeklenmesi konusunda yardımcı olabileceği fikri bu bağlamda düşünülmüştür. Sonuç olarak oldukça fazla sayıda problemin çözümü konusunda kullanılmış olan Derin Pekiştirmeli Öğrenme yöntemi (DRL) ortaya çıkmıştır (Arulkumaran vd., 2017). Bu yöntem dâhilinde öğrenen etmenler hem pekiştirmeli öğrenmenin deneme/yanılma tabanlı öğrenme metodunu kullanırken hem de derin sinirsel ağ sayesinde durum-eylem uzayının manuel biçimlendirilmesine gerek duymadan, yapılandırılmamış girdi verileri üzerinde eğitilebilmektedir. Son dönemde DRL`in yüksek sayıda uygulama geliştirilmiş olup bu alanlar arasında otonom araç kontrolü (S. Wang vd., 2018), görüntü işleme (Zhao vd., 2017), gerçek zamanlı robot kontrolü (Ibarz vd., 2021), doğal dil işleme (W. Y. Wang vd., 2018), oyunlar (Brown vd., 2020), sağlık hizmeti (Yu vd., 2021), siber güvenlik (Nguyen & Reddi, 2021), finans sektörü (Jiang vd., 2017), biyolojik veri madenciliği (Mahmud vd., 2018), ilaç tasarımı (Popova vd., 2018), yol planlama (Long & He, 2020), trafik sinyal kontrolü sistemleri (Zeng vd., 2018) ve çip yerleştirme (Mirhoseini vd., 2020) gösterilebilir. Bu alanlardan biri olan otonom araç kontrolü, çevresiyle nasıl etkileşime girdiğini öğrenmek için güçlendirme öğrenmesi kullanılarak ele

alınmıştır (S. Wang vd., 2018). Otonom araçların karmaşık çevrelerde de işlevsel güvenliği koruması gerekmektedir. Bu zorluklarla başa çıkmak için, öncelikle karmaşık durum ve harekvdanlarını ele alabilen derin deterministtik politika gradyanı (DDPG) algoritmasını benimsenmiştir. Görüntü işleme alanında trafik kameraları tarafından yakalanan görüntülere dayalı olarak araç sınıflandırılmasına odaklanılmıştır (Zhao vd., 2017). Odaklanmış bir görüntünün sınıflandırma olasılığının bilgi entropisini bir sonraki odaklanmış konumu bulmayı öğrenmek için RL etmenine rehberlik edecek ödül olarak alınmaktadır. Gerçek zamanlı robot kontrolü üzerine Ibarz ve arkadaşları (Ibarz vd., 2021) nesne manipülasyonu, hareket ve otonom sürüş gibi gerçek dünya robotik görevlerine RL'nin başarılı uygulamalarını gösteren birkaç vaka çalışmasını sunmaktadır. Her vaka çalışması, karşılaşılan zorlukları ve bunları ele almak için kullanılan belirli RL algoritmaları ve teknikleri tartışıyor. Oyun alanında yayımlana bir makalede, ReBeL (Recursive Belief-based Learning) olarak adlandırılan bir genel RL + Arama çerçevesi tanıtılmaktadır (Brown vd., 2020). ReBeL, iki oyunculu sıfır toplamlı oyunlarda bir Nash dengesine yakınsayan önceki çalışmalara dayanmaktadır. "Durum" kavramı, tüm etmenlerin ortak bilgi gözlemleri ve politikalarına dayanarak, kendilerinin hangi durumda olabileceği konusundaki olasılık dağılımlarını da içerecek şekilde genişletilmektedir. Belirtilen algoritma, genişletilmiş durumlar için bir değer ağı ve bir politika ağı eğiterek, kendini oynatan güçlendirme öğrenmesi yoluyla çalışmaktadır. Ayrıca, algoritma kendini oynatma sırasında arama için değer ve politika ağlarını kullanmaktadır. Siber güvenlik alanında oyun teorisi yaklaşımı ve bulanık Q-öğrenme temelli iki aşamalı bir sızma tespit ve önleme sistemi önerilmiştir (Nguyen & Reddi, 2021). İlk aşamada, veri toplayıcı düğüm, bulanık Q-öğrenmeyi kullanarak saldırganın kurban düğümlere neden olan anormallikleri tespit edilmiştir. Zararlı bilgi, veri toplayıcı düğüm tarafından önceden işlenmiş ve bir eşik değeriyle kontrol edilmiştir. Bu aşamadan sonra, ikinci aşamada, üst istasyon, bulanık Q-öğrenmeyi kullanarak en uygun savunma hareketlerini seçmiştir.

Görüleceği üzere DRL karmaşık özelliklere sahip, gerçek zamanlı ve geniş/sınırsız durum-eylem uzayına sahip birçok problemde kullanılmaya çalışılmaktadır. Bu tür problemlerin önemli özelliklerinden biri, birçok çevrede ödüllerin seyrek olmasıdır. Bu durum özellikle eğitim başlangıcında etmenin ödül

almasını ve bu sayede eğitimin doğru şekilde yönlendirilmesini zorlaştırmaktadır. Sonuç olarak, öğrenen etmenin öğrenme süresi uzayabilmekte veya etmenin en uygun olmayan çözümlere yakınsaması gözlemlenebilmektedir. Bu bağlamda, aynı ortamı paylaşan bir başka etmen veya uzmanın davranışlarının kopyalanması olarak tanımlayabileceğimiz sosyal öğrenme (SÖ) DRL algoritmalarının ilk aşamalarında öğrenme aktivitesini yönlendirebilmek için kullanılabilecek bir destekleyici yöntem olarak görülmüştür. SÖ, akıllı etmenlerin çevrelerinden öğrenmelerine olanak verir. Bu yönüyle, RL, denetimli öğrenme (SL) veya genetik algoritmalar gibi kişisel öğrenme yöntemlerinden farklılık gösterir. Önemli bir SÖ olan kopyalama veya imitasyon, bireyler arası davranışların transferine olanak verir.

Kopyalamanın DRL algoritmalarını destekleyici bir yöntem olarak kullanılması fikri son dönemde birçok araştırmanın konusu olmuştur. Bu konuda denenmiş ilk SÖ yöntemi davranışsal klonlamadır (Pomerleau, 1991). Bu yöntem ile aynı ortamı paylaşan bir uzmanın eylemlerinin içeren durum-eylem çiftleri öğrenen etmen tarafından uygulanır. Bu sayede direk olarak uzmanın eylem gösterimlerinin öğrenilmesi amaçlanır. Öte yandan tamamen gösterilen eylemlerin tekrarlanmasına dayanan davranışsal klonlamanın farklı problemlere genelleşmesinin düşük olduğu tespit edilmiştir (Ross vd., 2011). Bu sebeple sadece yüksek miktarda uzman davranışının klonlanmasına izin veren, belli özelliklere sahip problemlerde uygulanabilir olduğu görülmüştür.

DRL ile kullanılan bir diğer yöntem ise Ters Pekiştirmeli Öğrenmedir (IRL), (Ng & Russell, 2000). IRL ile gözlemlenen uzmanın eylem gezingesini diğer gezingelere göre önceliklendiren bir maliyet fonksiyonunun öğrenilmesi amaçlanır. Bu yöntem birçok farklı problem üzerinde denenmiş ve başarılı sonuçlar alınmıştır (Ratliff vd., 2009; Ziebart vd., 2008). Öte yandan, IRL algoritmaları oldukça yüksek işlem gücüne ihtiyaç duymaktadır. Bu durum robotik gibi gerçek zamanlı kontrol yöntemlerinin öğrenilmesinin gerektiği karmaşık çevrelerde önemli bir sorundur. Ayrıca IRL, direk olarak eylem önerilerinde bulunmak yerine uzmanın eylem seçimlerinin dolaylı yoldan içeren bir maliyet fonksiyonunun öğrenilmesini amaçlamaktadır. IRL algoritmalarının karmaşık çevrelerde uygulayabilmek için yeni araştırmalar yapılmaktadır (Finn vd., 2016).

Son dönemde davranışsal klonlama ve Ters Pekiştirmeli Öğrenme yöntemlerine alternatif olacak, modelden bağımsız, farklı problemlere genelleşebilen, düşük işlem maliyetine sahip ve direk olarak eylem önerebilen kopyalama yöntemlerinin geliştirilmesine çalışılmıştır. Geliştirilen yöntemlerden en bilineni, üretken çekişmeli ağlar (GAN) yönteminin bir versiyonu olan üretken çekişmeli kopyalama ile öğrenme (GAIL) (Ho & Ermon, 2016) algoritmasıdır. Bu yöntem, uzman eylemlerinin girdi olarak verildiği bir derin sinirsel ağın eğitimini amaçlar. Eğitilen ağ tarafından seçilen eylemler git gide uzmanın eylemlerine yakınsar. Eğitilen ağın önerdiği eylemler ile uzmanın sergilediği gezingeler arasında fark kalmadığında eğitim tamamlanmış olur. Bu yöntem farklı problemlerin çözümünde başarı göstermekle birlikte fazlasıyla uzmandan gelen gösterim bilgisine bağımlıdır. Uzmandan gelen gösterimlerin en uygun olmadığı durumlarda lokal çözümlere yakınsama durumu gözlemlenebilmektedir. Ayrıca ilgili yöntem fazlasıyla sosyal öğrenmeye ağırlık vermekte ve etmenlerin kişisel tecrübelerinden yararlanarak gelişmelerine olanak vermemektedir. Bu noktadan hareketle Yi ve diğerleri (Yi vd., 2018) bu yöntemi geliştirerek derin kopyalama ile pekiştirmeli öğrenme (DIRL) isminde bir algoritma geliştirmiştir. Buna göre ayrı bir derin ağ eğitilerek çevreden alınan gösterim bilgisinin uzman veya uzman olmayan şeklinde ayırt edilmesi amaçlanmıştır. Böylece GAIL algoritmasının uzman bilgisine bağımlılığının azaltılması planlanmıştır. Ayrıca, uzman olarak tanımlanan eylemler için etmenin kontrolünü sağlayan derin ağda ekstra ödül tanımları yapılmış ve bu ödüllerin etmenin eğitimi süresince giderek azalan şekilde etki etmesi hedeflenmiştir. Bu sayede tecrübe kazanan etmenlerin SÖ beraberinde kişisel öğrenmeden yararlanmaları amaçlanmıştır. Öte yandan ilgili çalışmada, eğitimin ilk aşamalarında uzman veya uzman olmayan davranışların ne şekilde ayrılacağı net olarak belirtilmemiştir. Ayrıca ekstra ödüllerin etkisinin eğitim süresince belli değerler arasında azaltılacağı söylenirken, azaltma işleminin hangi gözlem doğrultusunda yapılacağı açıklanmamıştır. Bir başka çalışmada ise (Mao vd., 2022), otonom su altı araçların kontrolü için GAIL algoritması kullanılmıştır. Bu amaçla, uzman veya uzman olmayan gösterimlerin ayrımı için ayrıştırıcı ismi verilen bir sinirsel ağ eğitilmiştir. Önceki çalışmalara ek olarak bu çalışmada birden fazla otonom su altı aracının eğitimi aynı anda yapılmış ve farklı araçların çevreden ve kopyalama ile elde ettikleri tecrübeler aynı hafıza alanında saklanmıştır. Bu

sayede en uygun olmayan uzman kaynaklı lokal çözümlere yakınsama probleminin çözülmesi amaçlanmıştır.

## 1.2 Tezin Amacı, Kapsamı, Hipotezi ve Katkıları

Son dönemde DRL destekleyici şekilde farklı kopyalama yöntemlerinin kullanıldığı görülmektedir. Ancak bu yöntemler doğal sistemlerde kopyalama ile öğrenme kullanımına uyum göstermemektedir. Doğal sistemlerde kopyalama ile elde edilen bilgi uzman veya gösterici tarafından belirli durumlarda gerçekleştirilen eylem gösterimleridir (Dautenhahn & Nehaniv, 2ss002). Gözlemlenen eylemlerin hangi durum ve şartlar altında yarar göstereceği öğrenen etmen tarafından keşfedilmelidir. Bu durum ancak belli bir deneme/yanılma mekanizması dâhilinde kopyalanan eylemlerin uygulanması ve bu eylemlerin sonucunda hesaplanacak bir takviye sinyali sayesinde yararlı eylem gösterimlerinin desteklenmesi, diğerlerinin ise daha az ihtimalle tekrarlanması şeklinde mümkündür. Bu sayede etmenin öğrenme aktivitesine olumlu etki eden eylem gösterimleri etmenin kişisel öğrenme aktivitesinin bir parçası haline gelmelidir. Bu yaklaşım RL`nin deneme/yanılma tabanlı yöntemine uyumluluk göstermektedir ve bazı araştırmalarda deneme/yanılma yöntemine dayanan sosyal öğrenmenin saf RL algoritmasının öğrenme hızını artırabileceği gösterilmiştir (Erbas vd., 2014). Ayrıca fiziksel sistemlerde kopyalama sırasında öğrenen etmenin göstericiyi izler vaziyette kalması ve ifa etmekte olduğu göreve ara vermesi gerekmektedir. Bu durum kopyalama ile alakalı bir maliyete sebep olmaktadır. Mevcut çalışmalarda kopyalama, uzman etmenin davranışlarının öğrenen etmene transferi şeklinde yapılmaktadır. Bu davranışlar ya öğretici olarak kabul edilmekte veya bir sinirsel ağ tarafından kategorize edilmektedir. Yapılan hiçbir çalışmada kopyalama kaynaklı maliyetlerin etkisi incelenmemiştir.

Bu tez kapsamında doğada kopyalamanın kullanımından esinlenen, SÖ destekli, modelden bağımsız, bir grup içerisinde öğrenen bireylerin öğrenme hız ve performanslarını artıran, özgün bir DRL geliştirilmiştir. Literatürde SDRL algoritmasının yöntemlerinde hafıza üzerinde durulmamıştır. Dolayısıyla bu çalışma kapsamında farklı hafıza mimarileri ile birlikte etkin deneme/ölçüm yöntemlerinin kullanılmasına yönelik bir yöntem önerilmektedir. Önerilen SDRL algoritmasının farklı problemler üzerinde denenmesi ve akıllı etmenlerin öğrenme hızı ve problem çözme başarısında artış olduğunun gösterilmesi hedeflenmiş ve

akıllı etmenlerin öğrenme süreçleri esnasında sosyal öğrenmeden yararlanması amaçlanmıştır. Geliştirilmiş algoritma, belli bir görevi yerine getiren bir etmen sürüsü içerisinde denenmiştir. Kopyalanan davranışlar, öğrenen etmenlerin öğrenme sürecini yöneten çift katmanlı bir kontrol sistemi tarafından seçilmesiyle farklı zamanlarda olasılıksal olarak etmenin hareketlerini belirleyen derin sinirsel ağa önerilmiştir. Sinirsel ağdaki zamansal değişimler incelenmiş ve kopyalanan davranışlar arasında öğrenmeye olumlu etki edenler desteklenmiştir. Böylece etmenin farklı gösterici etmenlerden, farklı davranışlar öğrenmesinin mümkün olacağı görülmüştür. Bu durumun, ortamda bir uzmanın olmadığı durumlarda dahi etmen sürüsünün kümülatif öğrenme hızına olumlu etki etmesi beklenmektedir. Ayrıca kopyalama sırasında ortaya çıkan ekstra maliyetler ve bu maliyetlerin sürünün öğrenme hızına etkileri incelenmiştir.

Bu araştırmayı teşvik eden bir hipotez önerilmiştir:

Hipotez 1 Deneme/yanılma tabanlı bir kopyalama metodu ile aynı ortamı paylaşan farklı tecrübe düzeyine sahip etmenlerin birbirlerinden sosyal şekilde öğrenmesi ve böylece DRL algoritmasının performansının artırılması mümkündür.

Bu tezde sunulan çalışma aşağıdaki katkıları içermektedir:

Topluluk içerisinde öğrenen etmenlerin, davranışı sergileyen etmenin uzman olup olmamasından bağımsız olarak, farklı etmenlerden öğrenebilmelerine olanak verilmiştir.

Tez kapsamında geliştirilen yöntem ile öğrenme algoritmasının verimliliğini arttırarak, teknolojik araçlar olan otonom akıllı etmenlerin/robotların kontrolü konusunda yaşam kalitesinin ve güvenliğinin arttırılması amaçlanmaktadır.

Tez kapsamında bazı çevrelerde gözlemlenen seyrek ödül probleminin çözümlenmesi için alternatif bir yöntem kullanılmıştır. Bununla birlikte takviye sinyali ile davranışların içsel geri bildirimin değerlendirilmesine olanak tanınmıştır.

Benzetim ortamında geliştirilmiş ve sınanmış olan özgün yöntemin, gerçek otonom robotlar üzerinde sınanması/geliştirilmesini içerecek yeni araştırma projelerinin oluşturulması mümkün olacaktır.

Algoritma ve yöntemin sonraki projelerde yapılacak araştırma/geliştirme faaliyetleri ile otonom etmenlerin gerçek zamanlı kontrol ve eğitimi amacıyla özgün algoritmaların geliştirilmesine önayak olabileceği düşünülmekte ve geliştirilmiş sistemlerin ekonomik açıdan değişiklik göstereceği düşünülmektedir.

### 1.3 Tez Çerçevesi

Bu tez, bir giriş bölümü de dâhil olmak üzere altı bölümden oluşmaktadır. İkinci bölümde, genel bilgiler çerçevesinde sosyal öğrenme, taklit, etmenlerde ve robotlarda taklit, sürü robotlar, derin öğrenme, pekiştirmeli öğrenme ve derin pekiştirmeli öğrenme konuları ele alınmaktadır. Üçüncü bölüm, önerilen yöntemin detaylı bir açıklamasını sunmaktadır. Bu bölüm, yeni yöntemin arkasındaki fikirleri ve yöntemleri anlamak için görevin karmaşıklığını açıklamaya yardımcı olur. Dördüncü bölüm, yapılan deneylerin neler olduğu ve sonuçlarını sunmaktadır. Beşinci bölüm, deneylerden elde edilen bulguları sunmaktadır. Altıncı bölümde ise gelecekteki çalışmalar için potansiyel iyileştirmeler ve sonuçlar tartışılır.

# 2. GENEL BİLGİLER

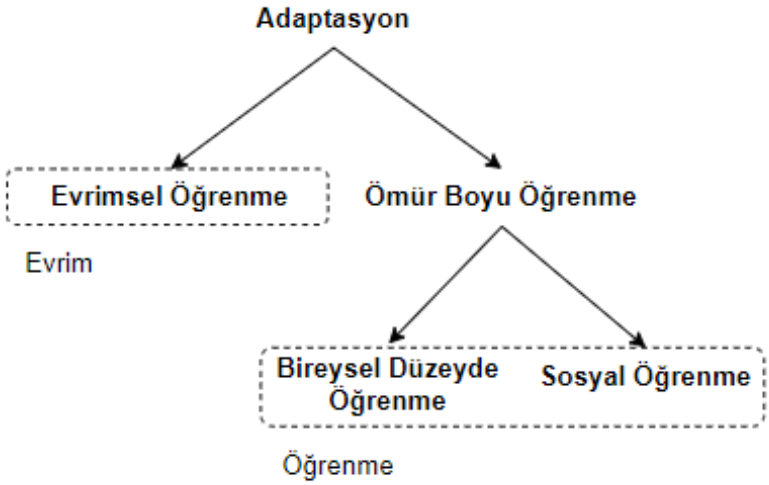
Bu bölüm, ilerideki bölümlerde ele alınacak konulara genel bir bakış sunmaktadır. İlk olarak, sosyal öğrenme (SÖ), etmenlerde ve robotlarda taklit ve sürü robotları kavramlarını açıklıyoruz. Ardından, derin öğrenme (DL), pekiştirmeli öğrenme (RL) ve derin pekiştirmeli öğrenme (DRL) yöntemlerinin genel bir özetini sunuyoruz.

## 2.1 Sosyal Öğrenme

Yaşamın tanımını yapabilmek ne kadar zor olsa da zamana göre değişim içerisinde olduğu mutlak bir gerçektir. Tüm canlı organizmalar çevreleriyle, birbirleriyle ve değişen koşullarla sürekli mücadele etmektedirler. Bu mücadele süresince adapte olabilenler yaşamlarını sürdürmektedir. Adaptasyon evrimsel öğrenme ve ömür boyu bireysel düzeyde öğrenme olmak üzere iki seviyeden (Şekil 2.1) oluşmaktadır.

Evrimsel anlamda adaptasyon bir canlının bulunduğu habitata, daha uygun hale geldiği süreçtir (Darwin, 1859). Evrim, çok uzun zaman dilimi içerisinde birçok neslin geçmesiyle gerçekleşir. Son yıllarda yeni teknolojik gelişmelerin ve hesaplama gücünün artmasıyla birlikte, doğadaki evrimden ilham alınarak robotik üzerine yapılan çalışmalar artarak devam etmektedir. Genetik algoritmalar ve hayvan sürüsü davranışı fikirlerine dayanan parçacık sürüsü optimizasyonu gibi evrimsel algoritmalar, robotikteki birçok probleme uygulanmıştır.

Bireylerin ömür boyu öğrenmelerini içeren süreçte öğrenerek adaptasyon, evrime kıyasla çok daha kısa sürede ve bireysel yaşamlarla sınırlı olarak gerçekleşmektedir. Ömür boyu öğrenme, bireysel öğrenme ve bireylerin başkalarından öğrendiği sosyal öğrenme (SÖ) olmak üzere iki kategoride incelenmektedir. Tüm canlılar çevre ile etkileşime girerek motor becerilerini etkin bir şekilde kullanmayı öğrenmektedirler.




**Şekil 2.1.** Canlı organizmaların adaptasyon süreçleri iki ana seviyeye ayrılır: evrimsel öğrenme seviyesi ve ömür boyu öğrenme seviye. Ömür boyu öğrenme seviyedeki süreçler ayrıca iki alt kategoriye ayrılır: bireysel öğrenme ve sosyal öğrenme. (Eiben vd., 2007)'den uyarlanmıştır.

SÖ, bireylerin çevrelerindeki diğer insanların davranışlarını gözlemleyerek, deneme-yanılma yoluyla bu davranışları kendi davranışlarına uyarlayarak yeni bilgi ve beceriler edindikleri bir öğrenme süreci olarak tanımlanabilir. Bu teori, 1800'lü yılların başlarından günümüze kadar birçok filozof, psikolog ve sosyolog tarafından geliştirilmiştir. İlk SÖ teorileri, John Locke tarafından ortaya atılmıştır. Locke, insan davranışının çevresel faktörler tarafından şekillendiğini savunmuştur (Locke, 1960). Daha sonra, Albert Bandura gibi psikologlar, öğrenmenin sadece ödüllendirme ve cezalandırma yoluyla gerçekleşmediğini, aynı zamanda gözlem ve taklit yoluyla da gerçekleşebileceğini savunmuştur (Bandura, 1962). Clark Hull, insan davranışının ödüllendirme ve cezalandırma yoluyla şekillendiğini savunan bir öğrenme teorisi olan davranışsal bir yaklaşım geliştirmiştir (Hull, 1943). Julian Rotter, SÖ teorisini özellikle klinik psikoloji alanında uygulayarak, bireylerin davranışlarının çevresel faktörlerle ve kişisel faktörlerle nasıl etkilendiğini incelemiştir (Rotter, 1947). Ayrıca, "içsel kontrol odağı" ve "dışsal kontrol odağı" kavramlarını ekleyerek, insanların hangi faktörlere daha fazla dikkat ettiklerinin, nasıl düşündüklerinin ve davrandıklarının belirlenmesinde bu faktörlerin önemli bir rol oynadığını savunmuştur. Kurt Lewin'in SÖ üzerine yaptığı çalışmalar arasında, sosyal psikolojinin temel kavramlarından biri olan "alan teorisi" yer almaktadır (Levin, 1951). Lewin'e göre, insan davranışı, bireyin içinde bulunduğu sosyal alan

ile etkileşim halindedir ve bu alan bireyin davranışını şekillendirmektedir (Levin, 1951). Katz ve Kahn, organizasyonlardaki bireylerin hem diğer bireylerin davranışlarını gözlemleyerek hem de bu davranışları taklit ederek öğrendiklerini savunmuşlardır (Katz & Kahn, 1966). Ayrıca, organizasyonlardaki bireylerin öğrenme sürecindeki motivasyonları ve beklentilerinin de SÖ teorisi tarafından açıklanabileceğini belirtmişlerdir. Lazarus, bireylerin stresle başa çıkmak için kullandıkları stratejilerin etkililiğinin, bu stratejileri kullanmadan önceki deneyimlerine ve beklentilerine bağlı olduğunu savunmuştur (Lazarus, 1966). Bandura, öğrenmenin sadece ödüllendirme ve cezalandırma yoluyla gerçekleşmediğini, aynı zamanda gözlem ve taklit yoluyla da gerçekleşebileceğini savunmaktadır (Bandura, 1977). Taklit ile birlikte bireylerin sadece kendi deneyimlerine dayanmak zorunda kalmadan, çevrelerindeki diğer insanların deneyimlerinden de yararlanmalarını sağlamaktadır. Daha sonraki zamanda rus psikolog Vygotsky, öğrenmeyi öğrenenin ilgisine bağlı olarak, ancak öğretenlerin rehberliğinde gerçekleşen bir sosyal etkileşim faaliyeti olarak tanımlamıştır (Vygotsky, 1978). Jean Lave ve Etienne Wenger, SÖ teorileriyle bilinmektedirler. Özellikle, topluluklar aracılığıyla öğrenme konusundaki çalışmaları ile tanınırlar. Bu yaklaşım, öğrenmenin sadece bireysel değil, toplumsal bir süreç olduğunu vurgulamışladır (Lave & Wenger, 1991) . SÖ sembolik iletişim veya dil olmaksızın beceri ve davranışların aktarılmasına izin vermektedir. Bu nedenle bir taklittir (Erbaş, 2012)

### 2.1.1 Taklit

Taklit psikologlar ve biyologlar tarafından geniş çapta incelenmiştir ve SÖ`nin önemli bir parçasıdır. Taklit üzerine yapılan psikolojik araştırmalar taklitin mekanizmaları ve işlevleriyle ilgilenirken yapılan biyolojik araştırmalar büyük ölçüde taklitin organizma için uyarlanabilir değerine odaklanmıştır (Zentall, 2001). Taklidin insanlara özgü olup olmadığı konusunda devam eden tartışmalar vardır. Meltzoff ve Moore (Meltzoff & Moore, 1977), bebeklerin ağzını açma ve basit parmak hareketleri gibi eylemleri taklit edebildiğini ortaya çıkarmışlardır. Taklit, bireyler arasında bilgi aktarımını sağladığı için kültürel öğrenmenin önemli bir parçası olarak görülmüştür (Tomasello vd., 1993). Blackmore (Blackmore, 1999), taklidin insanları hayvanlar arasında benzersiz kıldığı ve evrimsel olarak bir uyum faktörü olarak seçilmiş olabileceği öne sürmektedir. Bazı araştırmacılar herhangi

bir kültürel fikir, sembol ve uygulama birimini genlerin biyolojik evrimde oynadığı role benzeterek insanlar arasında taklit ile aktarılmasına izin verdiği için kültürel evrimde önemli bir işlevi olduğunu öne sürmektedirler (Blackmore, 1999; Dawkins, 1976). Bunların yanı sıra hayvanlarda taklitçi davranış araştırmalarında önemli ilerlemeler kaydedilmiştir (Heyes, 1996). Bu ilerlemeler, tepki eşleştirmede taklit örneklerini diğer sosyal etkilerden ayırt etmek için kullanılan prosedürel yöntemlerin geliştirilmesine bağlanmıştır (Zentall, 1996). Bu yöntemlere iki eylemli yöntem prosedürü (Dawson & Foss, 1965), çift yönlü kontrol prosedürü ve çapraz hedef prosedürü (Heyes & Dawson, 1990) örnek verilebilir. Taklitçi davranış araştırmalarına örnek vermek gerekirse Edward Thorndike'ın ünlü kedi deneyinde aç bir kedi bir yapboz kutusuna yerleştirilmiş ve kedinin bir kola basarak veya bir ip çekerek kaçmayı nasıl öğrendiğini gözlenmiştir (Thorndike, 1898). Deney, kedinin yavaş yavaş belirli bir davranışı kutudan kaçmak gibi istenen bir sonuçla ilişkilendirmeyi öğrendiğini göstermiştir. Deney, hayvan öğrenimi ve davranışı tarihinde bir dönüm noktası olarak kabul edilmektedir. Hayvan zekâsı ve öğrenimi üzerine daha ileri çalışmaların yolunu açmıştır (Thorndike, 1898). Daha sonraları kuşlar, Japon bıldırcınları, güvercinler, maymunlar ve farelerde taklidi tanımlamak için faydalı olmuştur (Akins & Zentall, 1996; Campbell vd., 1999; Custance vd., 1999; Dawson & Foss, 1965; Heyes vd., 1994; Zentall vd., 1996).

Makak maymunları üzerinde yapılan deneylerde, ventral kortekslerindeki nöronların hem maymunun bir eylemi gerçekleştirdiğinde hem de aynı eylemin başka bir hayvan tarafından gerçekleştirildiğinde aktif olduğu görülmüştür (Gallese vd., 1996). Bu nöronlar ayna nöronlar olarak adlandırılmış ve taklitten sorumlu bir mekanizma olarak düşünülmüştür. (Gallese vd., 1996). İlerleyen yıllarda, insan beyninde de ayna nöronların varlığına dair kanıtların ortaya çıkmasına neden olmuş ve farklı sosyal davranışların nöral temellerini anlamak için kullanılmıştır (Zentall, 2001). Ancak, ayna nöronların tam işlevi hala tam olarak anlaşılamadığı ve tartışmaların devam ettiği iddia edilmiştir (Zentall, 2001). Daha sonra, Demiris (Demiris, 2002) ayna nöronların çalışma mekanizmasından ilham alarak, dağıtık davranış tabanlı bir robot taklit yöntemi geliştirmiştir. Bu yöntem, robotların diğer robotların veya insanların hareketlerini taklit etmelerine olanak tanımıştır. Ayna nöronlarla ilgili yapılan araştırmalar ve Demiris'in yönteminin geliştirilmesiyle birlikte, robotlar ve yapay sinir ağları (YSA) tasarımlarında daha gelişmiş öğrenme

yöntemleri ve daha doğal insan-robot etkileşimleri gibi birçok uygulama potansiyeli ortaya çıkmıştır.

Mitchell (Mitchell, 1987) taklit için aşağıdaki adımları önermiştir:

1. Modelin kaydedilmesi veya algılanması.
2. Kopyanın üretilmesi.
3. Kopyanın modelle karşılaştırılması.
4. Kopyanın modelle benzerlik derecesinin değerlendirilmesi.

Bu adımlar, taklitçinin orijinal modele benzer bir kopya oluşturma niyetine sahip olduğunu ima etmektedir. Bu tanımlara uygun olarak ve doğada kopyalamanın kullanımından esinlenilerek etmen-etmene hareket taklit algoritması üçüncü bölümde geliştirilmiştir. Taklit algoritmasıyla birlikte bu bölümde kopyalama türleri ve hafıza tiplerinin yöntemleri anlatılmıştır.

## 2.2 Etmenlerde ve Robotlarda Taklit

Son yıllarda, etmen taklit çalışmaları da oldukça popüler hale gelmiştir. Etmen taklidi, bir etmenin diğer etmenlerin davranışlarını taklit etmesi ve öğrenmesi anlamına gelir. Bu yöntem, özellikle çoklu etmenler arasındaki işbirliği ve rekabeti optimize etmek için kullanılan DL teknikleri gibi yeni yapay zekâ teknolojilerinin gelişmesiyle birlikte daha fazla dikkat çekmektedir. Ayrıca, etmen taklidi, özellikle robotik ve yapay zekâ alanlarında, öğrenme sürecini hızlandırmak ve performansı artırmak için kullanılan bir tekniktir.

Taklit öğrenme hem robotlar hem de insanlar için önemli bir öğrenme yöntemidir. İnsanlar, başkalarının davranışlarını taklit ederek birçok beceri ve davranışı öğrenmektedirler. Robotlar da, programlanmış algoritmalar veya örnek olaylardan öğrenerek taklit etme yoluyla becerilerini geliştirebilirler.

Benzer şekilde, etmenler veya robotik programlar da taklit öğrenme yöntemlerini kullanarak bir görevi gerçekleştirmek için başarılı olmuş diğer etmenlerin davranışlarını taklit edebilirler. Bu, yapay zekâ sistemleri için önemli bir öğrenme yöntemidir, çünkü birçok görev, insanların benzer görevleri nasıl yaptığını taklit ederek başarılı bir şekilde gerçekleştirilebilir.

## 2.3 Sürü Robotlar

Kolektif robotik, doğal kolektif sistemlerden ilham alarak, robotların bir arada çalışmasıyla birlikte daha karmaşık görevleri yerine getirmesi için benzer prensipleri uygulayan bir araştırma alanıdır. Bu alandaki çalışmalar, özellikle sosyal böceklerin davranışlarından ilham almaktadır. Sosyal böcek kolonileri, çevreye dağılmış işbirliği ve otonom bireylerden oluşan merkezi olmayan sistemlerdir. Bu sistemlerde, bireysel böcekler sınırlı yeteneklere sahip olsa da, koloni olarak karmaşık görevleri yerine getirebilirler. Örneğin, karınca kolonileri yiyecek toplama ve yuva inşa etme gibi görevleri başarıyla yerine getirebilirler. Kolektif robotik, bu gibi doğal kolektif sistemleri inceleyerek benzer prensipleri robotik sistemlere uygular. Kolektif robotik araştırmaları, özellikle büyük ölçekli ve karmaşık görevlerin yerine getirilmesinde faydalıdır. Birden fazla robotun bir arada çalışarak daha verimli ve hızlı bir şekilde işler yapması, iş gücü maliyetlerini düşürebilir ve üretim süreçlerini optimize edebilir. Ayrıca, kolektif robotik araştırmaları, robotik sistemlerin insanlarla daha etkileşimli ve işbirlikçi olması için de kullanılabilir.

Sürü robotiği, kolektif robotiğin en bilinen alanıdır. Amacı, bir hedef için büyük sayıda basit fiziksel robotun koordinasyonunu sağlamaktır. Sürü robotiği, kolektif robotik alanının en tanınmış alanıdır ve amacı, bir hedef için bir araya getirilmiş birçok basit fiziksel robotu koordine etmektir (Şahin, 2005). Bu tanıma göre, sürü robotlarının istenilen kolektif davranışı, etmenler arasındaki yerel etkileşimler ve etmenler ile çevreleri arasındaki etkileşimlerden kaynaklanacak şekilde ortaya çıkması gerekmektedir. Bu da, her bir etmenin basit bir tasarıma sahip olması ve çevreleriyle etkileşimleri yoluyla kolektif bir davranış sergilemesi gerektiği anlamına gelmektedir. Sürü robotiği, doğal afetlerde arama ve kurtarma çalışmalarında, inşaat işlemlerinde ve çevre izleme görevlerinde kullanılabilir. Sürü robotiği tarafından yaygın bir şekilde incelenen problemler arasında sürü davranışı (flocking), kümelenme (aggragation), kümeleme (clustering), birlikte taşıma (cooperative transport) ve arama toplama (foraging) gibi konular yer almaktadır.

Sürü davranışı, sürü robotik sistemleri aracılığıyla incelenen ilk sosyal hayvan davranışlarından biriydi. Reynolds (Reynolds, 1987) tarafından yapılan kuş sürüsü makalesinde bireysel unsurların bir arada hareket ederek sürü veya sürüler oluşturmasını simüle etmek için bir algoritma önerilmektedir. Sürü davranışı üç basit yerel kural tarafından kontrol edilmektedir. Aynı zamanda, birliği sağlamak

için komşularının belirli bir yarıçap içindeki diğer kuşların konumlarına göre yönlenirler. Bu algoritma, sürülerin doğal davranışlarını taklit etmek için oldukça etkilidir ve birçok uygulama alanı bulmuştur. Benzer şekilde, Vicsek ve arkadaşları (Vicsek vd., 1995) tarafından geliştirilen Vicsek Modeli, bu kuralların kullanımıyla sürü davranışlarının matematiksel olarak modellenmesini sağlamaktadır. Bu model, farklı yoğunluklardaki robot gruplarının nasıl hareket ettiğini inceleyerek, farklı uygulama alanları için fikirler sağlamaktadır. Mataric (Mataric, 1994) mobil robot takımı kullanarak basit yerel kurallarla ortaya çıkan sürü davranışları incelenmiştir. Robotlar, kümelenme ve dağılma gibi basit kuralları birleştiren bir şema tabanlı rutin yöntemiyle kontrol edilmiştir. Robot takımı, ortaya çıkan bir sürü davranışı sergilenmiştir.

Sürü robotiğinde kümelenme ve kümeleşme, farklı amaçlar için kullanılabilen farklı robotik yaklaşımlardır. Sürü robotiğinde kümelenme, robotların tek bir hedefe doğru hareket ederek oluşturduğu bir sistemdir. Kümeleşme ise, belirli bir görev için koordine edilmiş robotların birbirleriyle etkileşim halinde olduğu bir sistemdir. Kümeleşme yaklaşımı doğal olarak gözlemlenen sürü davranışı, koloni davranışı veya karınca kolonisi gibi bazı biyolojik sistemlerden esinlenmiştir. Sürü robotiği alanında, kümelenme, robotların daha etkili bir şekilde hedeflerine ulaşmak için bir araya gelerek hareket etmelerini ifade eder. Bu davranışı, Trianni ve diğerleri (Trianni vd., 2003) simüle edilen robotlar üzerinde incelenmiştir. Bu çalışmada, her bir robot bir sinir ağı tarafından kontrol edilmiş ve genetik algoritmalar ile evrimleştirilmiştir. Robotların uygunluğu, tüm robotların ağırlık merkezine olan uzaklığı hesaplanarak belirlenmiştir. İki tür kümelenme davranışı gözlemlenmiştir: statik kümelenme davranışı ve dinamik kümelenme davranışı. Statik kümelenme davranışı, robotların sıkı ve istikrarlı bir küme oluşturmasına izin verirken, dinamik kümelenme davranışı daha esnek ve ölçeklenebilir bir küme oluşturmasına olanak tanımaktadır.

Sürü robotiğinde birlikte taşıma, bir grup hareketli robotun bir nesneyi bir araya getirerek taşımasını içeren bir robotik yaklaşımdır. Bu yaklaşım, doğada karınca kolonilerinde gözlemlenen işbirliği taşıma davranışı gibi biyolojik modellerden ilham almaktadır. Mataric ve arkadaşları (Mataric vd., 1995), iki robotun bir kutuyu hedefe doğru itmesini gerektiren bir görevde, robotların birbirleriyle iletişim kurabilmesi için basit bir iletişim protokolü kullanılmıştır. Bu

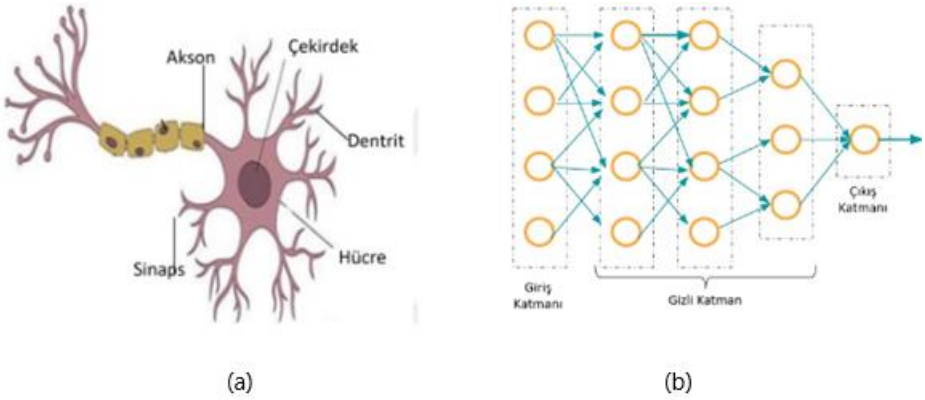
çalışma, robotların işbirliği yapması ve bir hedefe doğru ittikleri kutuyu başarıyla hareket ettirmesi konusunda olumlu sonuçlar vermiştir. Kube ve Bonabeau (Kube & Bonabeau, 1998) çalışmasında, bir grup hareketli robotun, karınca kolonilerinde gözlemlenen işbirliği taşıma davranışı gibi, bir nesneyi bir araya getirerek taşımaları amaçlanmıştır. Robotlar arasındaki koordinasyon ve iletişim, stigmergy yoluyla gerçekleştirilmiştir.

Sürü robotiğinde arama/toplama, bir grup hareketli robotun bir yüzeydeki kaynakları toplamak için işbirliği yaparak hareket ettiği bir robotik yaklaşımdır. Krieger ve Billeter (Krieger & Billeter, 2000),12 gerçek mobil robotla yürütülen bir arama/toplama görevi deneyi sunulmuştur. Deneylerinde, her robot farklı rastgele seçilmiş bir etkinleştirme eşiğine sahiptir, yani bir robotun yiyecek toplamak için tetikleneceği yuva enerji seviyesi, takımın aktivitesini düzenlemek için kullanılmıştır. Campo ve diğerleri (Campo vd., 2010) 20 gerçek robot sürüsü üzerinde matematiksel bir modele parametreleştirerek arama/toplama davranışını artırmayı amaçladılar. Stirling ve Floreano (Stirling & Floreano, 2010) ise bilinmeyen bir ortamda uçuş robotları sürüsünü dağıtmak için stratejiler geliştirilmiştir. Deneylerinde, uçuş robotları sürüsü enerji tasarrufu yapmak için dinamik bir sensor ağı oluşturdu. Robotların algılama ve iletişim yeteneklerine dayalı farklı stratejileri test edilmiştir.

### 2.4 Derin Öğrenme

Derin öğrenme, canlı organizmaların doğdukları andan itibaren çevrelerindeki nesneleri, sesleri ve kelimeleri tanıma gibi karmaşıklığı yüksek ve zekâ düzeyi gerektiren eylemleri ilham kaynağı olarak alan bir yapay zeka dalıdır. Canlı organizmalar, karmaşık bilgileri işlemek için milyonlarca nöron aracılığıyla anlamlı bilgi çıkarmaya çalışmaktadırlar. Benzer şekilde, sinir ağları da karmaşık bilgileri anlamlandırmak amacıyla nöron ağını taklit ederek çok katmanlı sinir ağları kullanılmaktadır. Tipik bir biyolojik nöron hücresinin temsili (Jeremiah vd., 2021) ile yapay sinir ağı hücresinin karşılaştırılması, Şekil 2.2`de gösterilmiştir. Her katman, birçok yapay nöron içermektedir. Bu yapay nöronlar, girdi verisini alarak bir aktivasyon fonksiyonu uygulamakta (Feldman vd., 1988) ve çıktı verisini üretmektedir. Çıktı verisi, bir sonraki katmana girdi olarak iletilmektedir. Bu şekilde, her katman veriden farklı seviyelerde özellikler çıkarmaktadır. Bu, sinir

ağlarının karmaşık görevleri gerçekleştirme yeteneklerini arttırmaktadır ve derin öğrenmenin başarısının temelini oluşturmaktadır.




**Şekil 2.2.** Tipik bir biyolojik nöron hücresinin temsili (a) (Jeremiah vd., 2021) ile yapay sinir ağı hücresinin (b) karşılaştırılması

DL bu çalışmada DRL algoritmasının kritik bir bileşeni olarak önemli bir rol oynamaktadır. RL'nin bir alt dalı olan Q-öğrenme içinde, durumlar ve eylemler arasındaki ilişkiyi haritalayabilen işlev genellikle karmaşık bir matematiksel formülasyonla ifade edilmesi zor bir işlemdir. Bu nedenle, bu haritalamayı yaklaştırmak için DL kullanılarak Q-ağlarını oluşturmaktadır. Bu ağlar, Q-tablosunun yerini alarak Derin Q Ağları (DQN) yöntemini ortaya çıkarmaktadır.

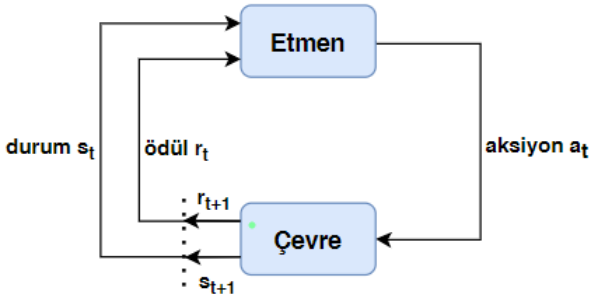
## 2.5 Pekiştirmeli Öğrenme

Canlı bir organizma için çevresini algılamak hayati bir öneme sahiptir ve bununla birlikte hayatta kalabilmek için değişen çevre koşullarına adapte olmaları gerekmektedir. Pekiştirmeli Öğrenme (RL), etkileşimli bir ortamda deneme yanılma yöntemiyle bir yapay etmenin, çevresel durumlarını maksimize edilmesi amaçlayan ve bir ödül değerine eşleyen bir bireysel öğrenme algoritmasıdır (Richard S. Sutton & Andrew G. Barto, 2018). RL`in amacı, en iyi sonuçları elde etmek için optimum eylem dizisini haritalamaktır. Bu hedef, etmenin çevreyi keşfetmesine, etkileşime girmesine ve çevreden öğrenmesine olanak sağlayarak gerçekleştirmektir. Bu nedenle, temel RL formülasyonu, algılama, eylem ve hedef olmak üzere üç temel bileşeni içermektedir (Richard S. Sutton & Andrew G. Barto, 2018). RL, denetimli (SL) ve denetimsiz öğrenme ile birlikte makine öğrenmesinin üç temel yaklaşımından biridir. SL, etmen, bazı durumlar için en iyi yanıtları bilen

bir denetleyicinin sağladığı örneklerden öğrenmektedir. Ancak RL, doğru yanıtları öğrenen etmenine herhangi bir durum için vermemektedir. Bunun yerine, öğrenen etmen, çevresi ile etkileşimde bulunarak bu yanıtları keşfetmesi beklenmektedir. RL, tamamen çevre ile etkileşime dayandığı için keşif ve sömürü arasında denge bulmayı gerektirir. Anında yüksek bir ödül elde etmek için etmen, daha önce denediği ve istenen sonuçları üreten bir eylemi tercih edebilir. Ancak aynı zamanda yüksek ödül sunan eylemleri keşfetmek için daha önce denenmemiş eylemleri de denemelidir. Bu nedenle, gelecekte daha iyi eylem seçimleri yapabilmek için etmen, mevcut bilgilerini kullanmalı ve aynı zamanda diğer olası eylemleri de keşfetmelidir. Keşif ve sömürü dengesi, RL literatüründe yaygın bir şekilde incelenmiştir. RL sisteminde, bir politika, bir ödül fonksiyonu, bir değer fonksiyonu ve bir çevre modeli olmak üzere dört ana bileşen tanımlanabilir (Richard S. Sutton & Andrew G. Barto, 2018).

RL çerçevesinde etmen ile çevre arasındaki etkileşim Şekil 2.3'de verilmiştir ve şöyle tanımlanabilir. Her karar adımında t, etmen çevrenin durumunu gözlemleyerek $s_t$ (t anındaki durum) elde etmektedir. Mevcut $s_t$ durumuna dayanarak, etmen bir eylem $a_t$ (t anında seçilen eylem) gerçekleştirmektedir ve eylemin çevre üzerindeki etkisinin bir göstergesi olarak bir $r_t$ ödülü almaktadır. Bu eylemin sonucu olarak, çevrenin mevcut durumu değişir ve etmen bir sonraki durum olan $s_{t+1}$ ve bir sonraki zaman adımında sayısal bir ödül olan $r_{t+1}$ gözlemleyecektir. Algoritma, her adım kararında eylemin ve çevrenin mevcut ömrünün değerlerini koruyarak bu bilgiyi bir sonraki adımı değerlendirmek için kullanılmıştır (Kaelbling vd., 1996). Yeterli bir öğrenme seviyesiyle birçok tekrarlamadan sonra, etmen uzun vadede toplam ödülü maksimize ($V\pi(s)$) etmek için eylem almaktadır. Etmenin asıl amacı, bir değer fonksiyonu olarak tanımlanan uzun vadeli hedefleri en üst düzeye çıkaran bir politika ($\pi$) keşfetmektir. Politika ($\pi$), etmenlerin eylemlerini gerçekleştirmesinde rehber görevini üstlenen bir fonksiyondur. RL`in temel amacı, bir problemin çözümü için en uygun $\pi$`yi bulmaktır. Bu, durum-değer fonksiyonu ve eylem-değer fonksiyonunun kullanımı yoluyla gerçekleştirmektedir. Durum-değer fonksiyonu (V), $\pi$'yi takip ederek elde edilen ödül miktarını ölçerek bir durumun değerini belirlerken, eylem-değer fonksiyonu (Q), $\pi$'yi takip ederek bir durumda bir eylem gerçekleştirildiğinde elde

edilen ödül miktarını ölçerek bir eylemin değerini belirlemektedir. En uygun π, V`nin maksimize edildiği durumda elde edilmektedir.




**Şekil 2.3.** RL çerçevesinde etmen ile çevre arasındaki etkileşim (Kaelbling vd., 1996)

RL yöntemleri, etmenin en yüksek ödülü elde etmeyi amaçladığı durumlar ve eylemlere odaklanarak, bu hedefe ulaşmak için ödül ve ceza işaretleri kullanılmaktadır. Bu tür problemlerde genellikle ardışık karar sistemleri olarak modelleme yapılmakta ve bu nedenle ilgili sorunlar RL ortamlarında Markov Karar Süreçleri (MDP) olarak tanımlanmıştır. MDP, RL görevlerinde, bir eylemin bir durumda yarattığı etkilerin sadece o duruma ve alınan eyleme bağlı olduğu durumları tanımlamaktadır. Bu nedenle, MDP`ler RL araştırmalarında büyük öneme sahiptir, çünkü çoğu yöntem bu tür MDP temelli sorunları çözmeye odaklanılmıştır. MDP nin temel özelliği, gelecek durum $s_{t+1}$ `in geçmiş durumlara değil, yalnızca mevcut durum olan $s_t$`ye bağlı olmasıdır.

RL problemlerini çözmek için üç temel yöntem sınıfı bulunmaktadır. İlk sınıf, Dinamik Programlama (Bellman, 1957) olarak adlandırılmıştır. Bu yaklaşım, hesaplama yükünün azaltılmasına katkıda bulunan ve Richard Bellman tarafından geliştirilen bir matematiksel optimizasyon yöntemidir (Bellman, 1957). Çevrenin mükemmel bir modeli verildiğinde, her durumun değerini yineleyerek hesaplamak için kullanılmıştır. İkinci sınıf yöntem, Monte Carlo Yöntemleri'dir ve çevrenin tam bir bilgisine ihtiyaç duymadan çalışabilirler. Monte Carlo yöntemleri, deneyimlerden öğrenmek için örnek dönüşlerin ortalamasını alarak kullanılmaktadır. Üçüncü sınıf ise Temporal Difference (Zamansal Fark) (TD) öğrenmedir. Dinamik programlama ile Monte Carlo yöntemlerinin bir sentezi olarak kabul edilmektedir. TD öğrenme yöntemleri, deneyimden doğrudan

öğrenirler ve çevrenin tam bir modeline ihtiyaç duymadan tahminleri güncelleyebilirler. Bu, önceki öğrenmelere dayalı tahminlerin tamamlanmış bölümleri beklemeksizin güncellenebileceği anlamına gelmektedir. Ayrıca, TD yöntemleri sonraki tahminlerin genellikle belirli bir derecede ilişkili olduğunu göz önünde bulundurur.

TD öğrenmeye dayalı iki tür kontrol yöntemi bulunmaktadır. SARSA (Rummery & Niranjan, 1994) bir politika temelli kontrol yöntemidir. Bir politika verildiğinde, SARSA algoritması kullanan bir etmen, çevresi ile etkileşime girerek politikasını güncelemektedir. Algoritma, durum-eylem çiftleri için tahminlerini güncellerken şu adımları takip eder:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \quad (2.1)$$

Denklem 2.1`de $s_t$ mevcut durumu, $a_t$ seçilen eylemi, $r_{t+1}$ eylem gerçekleştirildiğinde etmenin aldığı ödülü, $s_{t+1}$ etmenin yeni durumunu, $a_{t+1}$ yeni durumdaki seçilecek eylemi, α öğrenme oranını ve γ ise indirim faktörünü ifade etmektedir. Denklemde belirtildiği gibi, bir durum-eylem çiftinin Q değeri, öğrenme oranı α tarafından düzeltilen bir hata ile güncellenmektedir.

TD `ye dayalı ikinci kontrol yöntemi politika dışı (off-policy) kontrol algoritması olan Q-Öğrenme`dir (Watkins, 1989). Eylemleri gerçekleştirmek için kullanılan politika ile güncellenen politika arasındaki farklılıktan dolayı politika dışı olarak nitelendirilmektedir. Bu yöntem, takip edilen politikadan bağımsız olarak öğrenilen deneyimlere dayalı olarak doğrudan durum-eylem çiftlerinin Q değerini tahmin etmektedir. Bir adımlık Q-öğrenme aşağıdaki gibi tanımlanır:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma max_a Q(s_{t+1}, a) - Q(s_t, a_t)] \quad (2.2)$$

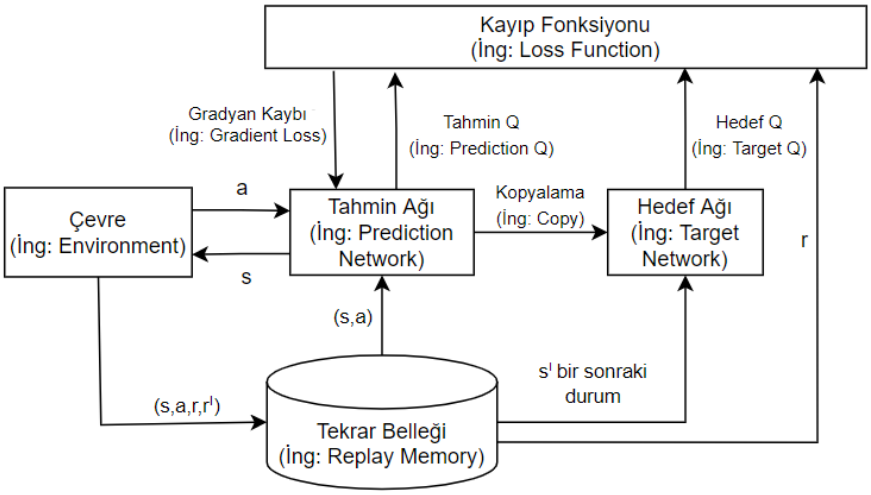
Q-Öğrenme, RL alanında en önemli gelişmelerden biri olarak kabul edilmektedir. Her bir durumda gerçekleştirilebilecek eylemlerin beklenen faydasını karşılaştırabilme yeteneğine sahiptir ve bu, çevre modeline ihtiyaç duymadan gerçekleştirilebilir.

## 2.6 Derin Pekiştirmeli Öğrenme

RL ve DRL, yapay zekanın doğal zekaya yaklaşmasını sağlayan iki güçlü yöntemdir. Bu yöntemler, canlıların çevrelerinden gelen geri bildirimlere göre

davranışlarını değiştirmeleri ve öğrenmeleri prensibine dayanmaktadır. RL, etmenin durum ve eylem arasındaki değer ilişkisini öğrenmesini sağlayan bir Q fonksiyonu kullanılmaktadır. Bu fonksiyon, basit veya karmaşık bir şekilde tanımlanabilmektedir. Q öğrenmenin karmaşık ve doğrusal olmayan ortamlarda, çok sayıda hücreyi içeren bir tabloyu oluşturarak işleyebilmesi zor olabilmektedir. Mevcut bilinen durumların sayısının artmasıyla, yeni durumların Q değerlerini hesaplamak oldukça zorlaşabilir. Ayrıca, gereken Q-tablosunu oluşturmak ve her bir durumu keşfetmek için gereken süre gerçekçi bir zaman çerçevesinde uygulanamayabilir. Bu nedenlerle, değerlerin matriste tutulması yerine sinir ağı ile işlenerek kullanılması, daha pratik bir yaklaşım olabilir. Bu yaklaşım, DQN yapısının temelini oluşturmaktadır.

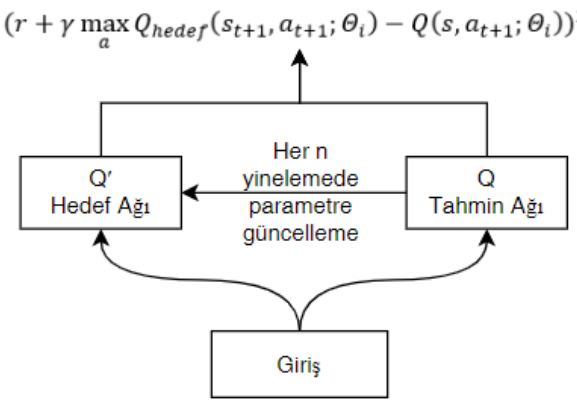
DQN, bir ortamdaki en uygun eylemleri öğrenmeye yönelik bir algoritma olan Q öğrenmeyi derin sinir ağlarıyla birleştiren bir takviyeli öğrenme tekniğidir. DQN temelinde matematiksel bir ifadeden oluşmaktadır ve (2.2) de görülen Bellman denklemi ile ifade edilmektedir. DQN formülü, bir etmenin belirli bir durumda belirli bir eylemi gerçekleştirmesinin beklenen değerini göstermektedir. Bu değer, etmenin aldığı ödüllerin ve gelecekteki durumlardan elde edebileceği maksimum Q değerlerinin toplamıdır. Bellman denklemini ve sinir ağlarını temel alan DQN veri akışı (Arwa & Folly, 2020), Şekil 2.4`de gösterilmiştir. Etmenin çevreyle etkileşime girmesiyle Q öğrenme algoritmasındaki gibi aksiyon seçilerek tahmin ağına ($Q(s_t, a_t)$) verilmektedir. Ancak kullanılacak sinir ağlarını eğitmek için çok fazla veri gerekmektedir. Bu durumda etmen bilinmeyen bir ortamda keşif yapacağından ötürü önceden eğitim setinin hazırlanması imkânsızdır. Bu soruna çözüm olarak tekrar belleği kullanılmaktadır. Hareket halinde olan bir etmenin oluşturduğu deneyim verileri alınarak bu bellekte saklanmaktadır. Her yinelemede eski bellek yenisiyle değiştirilmektedir. Böylelikle sinir ağlarının eğitimi için veri seti oluşturulmaktadır. Oluşturulan tekrar belleği içerisinden, tahmin ağı şu anki durum bilgisini kullanırken hedef ağ ise bir sonraki durum bilgisini kullanmaktadır. Tahmin ağ, etmenin hareketleri için tahmin edilen Q değerlerini hesaplamak için kullanılırken, hedef ağ etmenin doğru Q değerlerini hesaplamak için kullanılmaktadır. Kullanılan çift ağ yapısı, DQN algoritmasının daha istikrarlı ve verimli bir şekilde eğitilmesini sağlamaktadır.




**Şekil 2.4.** DQN Veri Akış Şeması (Arwa & Folly, 2020)

Ağ yapılarına verilen değerlerin sonucunda elde edilen tahmin Q değeri, hedef Q değeri ve tekrar belleğindeki verilerden gelen ödül değerinin birleşimi ile kayıp fonksiyonu oluşmuştur. Kayıp fonksiyonu (2.3) `de gösterilmektedir. Kayıp fonksiyonu, Q değerlerinin güncelleme kuralından türetilmiştir ve Q öğrenme algoritmasındaki TD hata hesaplamasının yerini almıştır. TD hatası genişletildiğinde, yeni Q değerinin hedef ağı tarafından üretilen Q değerine eşit olduğunu ve eski Q değerinin tahmin ağında hesaplanan Q değerine eşit olduğunu göstermektedir. Kayıp fonksiyonun hesaplanması ve parametrelerin güncellenmesinin mantığı (El Gourari vd., 2021) şekil 2.5'te gösterilmiştir.

$$L(\theta) \leftarrow (r + \gamma \max_{a} Q_{hedef}(s_{t+1}, a_{t+1}; \theta_i) - Q(s, a_{t+1}; \theta_i))^2 \quad (2.3)$$



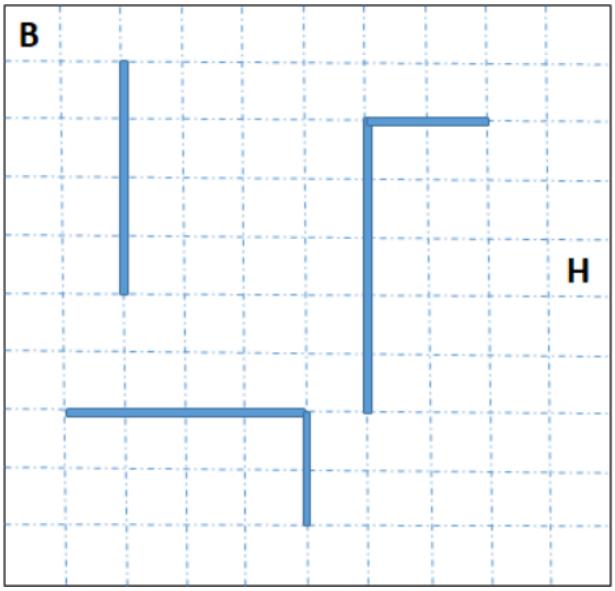
**Şekil 2.5.** Belirli frekans aralığıyla hedef ağın güncellenmesi (El Gourari vd., 2021)

Hesaplanan bu loss değerini en aza indirmek için geri yayılım algoritmasını veya gradyan inişi kullanılarak ağ üzerindeki parametreler güncellenmektedir. Her yinelemede hedef işlem bir süreliğine sabit tutulduğu için tahmin ağındaki parametrelerin hedef ağa kopyalanması daha istikrarlı bir eğitime yol açmaktadır. Tahmin ağının parametreleri, her bir eğitim adımı sırasında güncellenmektedir. Ancak, parametreleri hedef Q değerleri hesaplamak için kullanılan sabit bir hedef ağın parametrelerine göre güncellenmektedir. Bu, ağın eğitimi sırasında hedef Q değerlerindeki ani değişikliklerin neden olduğu aşırı tahmin sorununu engellemektedir.

Kopyalama yoluyla doğal sistemlerde elde edilen bilgi, uzman veya gösterici tarafından belirli durumlarda gerçekleştirilen eylem gösterimlerini ifade etmektedir (Dautenhahn & Nehaniv, 2002). Gözlemlenen eylemlerin ne zaman ve hangi koşullar altında yararlı olabileceği, öğrenen etmen tarafından keşfedilmelidir. Bu keşfin mümkün olabilmesi, belirli bir deneme/yanılma mekanizmasına dayanmaktadır. Bu deneme/yanılma mekanizması sayesinde, bazı eylemler kopyalanıp uygulanır ve bunların sonuçlarına göre bir takviye sinyali hesaplanmaktadır. Bu sinyal, yararlı olan eylemleri pekiştirmektedir, yararsız olanları ise daha seyrek yapılmasını sağlamaktadır. Bu yaklaşım RL`in yapısına uyumluluk göstermektedir. Ayrıca, kopyalama işlemi sırasında etmenin gözlemlerine devam etmesi veya ara vermesi gerekmektedir. Bu süreçteki davranışlar, bir sinirsel ağ tarafından kategorize edilmektedir. Yapılan çalışmada DQN yöntemi kullanılmıştır.

# 3. MATERYAL VE YÖNTEM

Bu tez çalışmasının amacı doğada kopyalamanın kullanımından esinlenen, SÖ destekli özgün bir DRL algoritmasının geliştirilmesidir. İlgili algoritma ortamda bir uzmanın varlığından bağımsız olarak farklı etmenlerden kopyalama yoluyla öğrenmeye olanak vermektedir. Öğrenen etmenler çift katmanlı bir kontrol sistemi tarafından yönetilmiş ve kopyalama davranışının kontrol sistemi tarafından seçilebilecek olası davranışlardan biri olacaktır. Bu sayede kopyalama davranışının, etmenin öğrenme performansına etkisi ile birlikte, bu davranışın öğrenme sürecinde dinamik olarak değişimi incelenmiştir. Geliştirilen SDRL algoritmasının sürü zekâsı konusunda incelenmiş arama/toplama problem tipi üzerinde denenmesi ve akıllı etmenlerin öğrenme hızı ve problem çözme başarısında artış olduğunun gösterilmesi hedeflenmiştir. Şekil 3.1'de bu amaçla dizayn edilmiş bir benzetim ortamı görülmektedir.



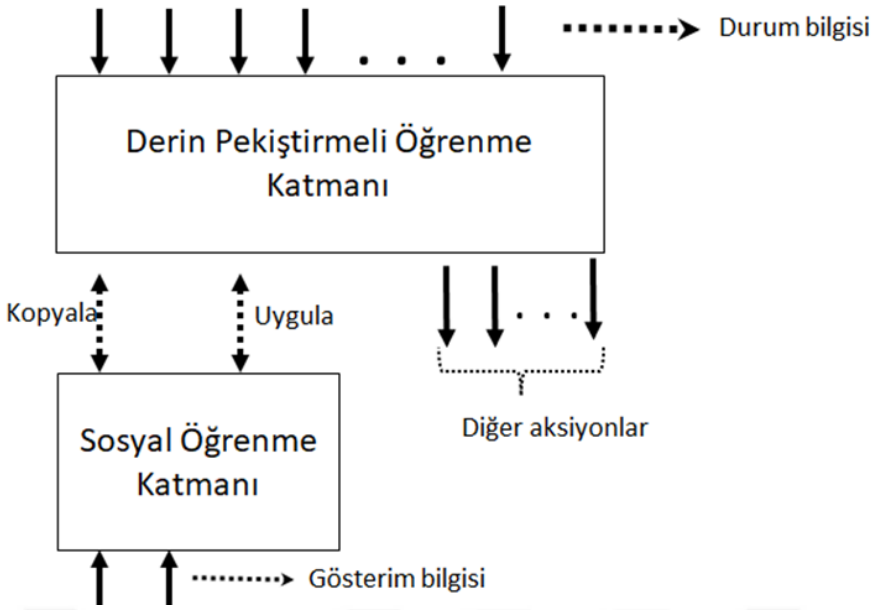

**Şekil 3.1.** Arama/toplama görevinin öğrenilmesi amaçlı tasarlanan benzetim ortamı

Deney ortamı bir hücreler dünyası şeklinde dizayn edilmiştir. Bu ortamda öğrenen etmenlerden beklenen B başlangıç pozisyonundan H hedef lokasyonuna giden en kısa yolu bulmalarıdır. Deney ortamına ilgili ortamın boyutlarını belirleyen kenarların dışında etmenin hareketlerini kısıtlayan bazı engeller yerleştirilmiştir. Öğrenen etmen her birim zamanda bulunduğu lokasyonu ve

etrafındaki engellerin varlığını algılayabilmektedir. Buna bağlı olarak öğrenen etmen, kullanmakta olduğu algoritmaya göre yapacağı aksiyona karar verebilmekte ve bulunduğu hücrenin komşu hücrelerinden birine hareket edebilmektedir. Bu deney ortamında etmenin ödül kazanacağı tek durum hedef lokasyona ulaşılmasıdır. Bu durum, tez sürecinde incelenmesi hedeflenmiş olan seyrek ödül içeren problemler tanımına uymaktadır. Deney ortamının boyutları, başlangıç ve hedef noktalarının lokasyonları ile engellerin sayısı ve dağılımı parametrik olarak tanımlanmış ve bu sayede farklı özelliklere sahip deney ortamlarının oluşturulması mümkün olmuştur.

Geliştirilmiş olan deney ortamında saf DRL algoritması kullanan etmenler ile tez kapsamında geliştirilmiş olan SDRL algoritması kullanan etmenlerin performansı karşılaştırılmıştır. Bu amaçla literatüre uygun olarak, Şekil 3.1'de görülen arama/toplama problemi çözen etmenler için belli aralıklarla sinirsel ağda öğrenilmiş en yüksek seçilme olasılığına sahip aksiyon dizileri belirlenmiştir.

Kopyalamanın doğada kullanımından esinlenen sosyal öğrenme destekli, modelden bağımsız ve özgün bir DRL algoritması geliştirmiştir. Bu bağlamda kullanılması planlanan yaklaşım, davranış bilimi konusunda dil ve diğer davranışların sosyal bireyler arasında öğrenimini açıklayan kullanım tabanlı teoriden esinlenmektedir. Bu yaklaşıma göre, bireylerin belli davranışları öğrenebilmeleri için, öğrenilecek davranışın bireylerin çevrelerinde sergilenmesi, ilgili davranışın öğrenen birey tarafından kopyalanması ve sonrasında kopyalanan davranışların öğrenen birey tarafından gerçekleştirilmesi gerekmektedir. Gerçekleştirme sonrası elde edilen içsel geri bildirim ve çevresel ipuçları sayesinde gözlemlenen davranışlar bireyin davranış repertuvarına eklenmiştir. Bu noktada kopyalamanın öğrenen birey tarafından gerçekleştirilebilecek bir aksiyon olarak tanımlanması ve kopyalanan davranışların değerlendirilmesine olanak veren bir içsel geri bildirim mekanizmasının oluşturulması yüksek önem arz etmiştir. İlgili yaklaşıma uygun olarak Şekil 3.2'de görüleceği üzere çift katmanlı bir kontrol mimarisi geliştirilmiştir.




**Şekil 3.2.** Sosyal öğrenme destekli Derin Pekiştirmeli Öğrenme algoritması

Geliştirilen ilgili mimari, üst seviyede DRL katmanı ile alt seviyede SÖ katmanından oluşacak şekilde tasarlanmıştır. Üst katmanda, DRL algoritması çalışacak ve etmenin ifa ettiği görev doğrultusunda eğitilmiştir. Üst katmanda sekiz aksiyon ile birlikte "kopyala" ve "uygula" aksiyonlarıda seçilebilcektir.

Üst katmanın seçebileceği sekiz adet aksiyon sinirsel ağ tarafından ateşlendiğinde iki katman arasında aşağıda belirtilen etkileşimler gerçekleşir:

- SÖ katmanına "NW, N, NE, W, E, SW, S, SE" yönlerinden seçilen bir emir gönderilir.
- Etmenin kontrolü alt katmana geçer, etmen bu süreçte bu emri gerçekleştirmeyi dener.
- Bu işlemlerin sonunda etmenin kontrolü üst seviye kontrol katmanına geri verilir.

Üst katmanın seçebileceği aksiyonlardan biri olan "kopyala" aksiyonu sinirsel ağ tarafından ateşlendiğinde iki katman arasında aşağıda belirtilen etkileşimler gerçekleşir:

- SÖ katmanına "kopyala" emri gönderilir.
- Etmenin kontrolü alt katmana geçer, etmen belli bir süre hareketsiz kalır ve aynı ortamda bulunan bir başka etmenin davranış gösterimlerini izler.

- Gözleme işlemi bittiğinde, gözlemlenen davranış gösterimleri alt katmanda bulunan hafıza alanına kaydedilir.
- Bu işlemlerin sonunda etmenin kontrolü üst seviye kontrol katmanına geri verilir.

Üst katmanın seçebileceği bir diğer aksiyon ise "uygula" aksiyonudur. Bu aksiyon sinirsel ağ tarafından ateşlendiğinde iki katman arasında aşağıda belirtilen etkileşimler gerçekleşir:

- SÖ katmanına "uygula" emri gönderilir.
- SÖ katmanı, içerdiği hafıza alanına "kopyala" emri sonucu kaydedilmiş olan bir davranış gösterimini olasılıksal olarak seçer ve bu gösterimi üst seviye kontrol katmanına gönderir.
- Üst seviye katman gönderilmiş olan davranış gösterimini uygular.
- Üst seviye katman, ilgili gösterimin uygulanması sonrası elde edilen ödülleri ve durum-aksiyon çiftlerindeki değişimi inceler ve alt seviye katmana bir takviye sinyali gönderir.
- SÖ katmanı, olumlu takviye alan davranış gösterimlerinin seçilme ihtimalini artırırken negatif takviye alan gösterimlerin seçilme ihtimalini azaltır.

Mevcut araştırmalarda iki farklı yaklaşım bulunmaktadır. Bunlardan ilki, uzman tarafından gerçekleştirildiği bilinen davranış tecrübelerinin belli aralıklarla yapay sinirsel ağa gönderilmesidir (örneğin (Ho & Ermon, 2016)). Bu yöntemde kopyalama davranışının ne şekilde ve ne sıklıkla gerçekleşeceği önceden tanımlanmıştır. İlgili literatürdeki diğer yöntem ise uzman ile uzman olmayan etmenlerden elde edinilmiş davranış tecrübelerinin, ikinci bir derin sinirsel ağ ile kategorize edilmesi ve uzmandan alındığı tespit edilen davranışların asıl eğitimi yapılan derin sinirsel ağa gönderilmesi şeklindedir (örneğin (Ho & Ermon, 2016)). Geliştirilmiş olan özgün kontrol sisteminde ise kopyalama davranışının ne zaman ve hangi sıklıkla ateşleneceği, etmenin ifa ettiği görev doğrultusunda eğitilen derin sinirsel ağa bırakılmıştır. Bu yöntem sayesinde kopyalama davranışının etmenin eğitimi süresince değişiminin incelenmesi mümkün olmuştur. Ayrıca DRL katmanı tarafından "kopyala" aksiyonunun ateşlenmesi durumunda yukarıda açıklandığı

üzere öğrenen etmen bir süre hareketsiz kalmaktadır. Bu durum kopyalama sebebiyle bir miktar zaman kaybedilmesine ve buna bağlı olarak öğrenme hızı/performansı açısından bir maliyet oluşmasına sebep olabilmektedir. Bu maliyetlerin SÖ kullanan etmenin öğrenme hızına etkisi ilk kez incelenmiştir.

Üst katmanda bulunan Derin Sinirsel ağın seçebileceği diğer aksiyonlar etmenin öğrenmekte olduğu görev ile alakalı davranışlarını tetiklemektedir. Görüldüğü üzere, kopyalamanın doğada kullanımına uygun olarak, "kopyala" ve "uygula" aksiyonları etmenin asıl kontrolünü sağlayan DRL katmanı tarafından seçilebilecek aksiyonlar içerisindedir. Bu tür bir dizayn sayesinde akıllı etmenlerin eğitimlerinin başlangıç aşamalarında ilgili aksiyonları daha sık ateşleyerek kopyalama tabanlı sosyal öğrenmeden yararlanırken, ilerleyen aşamalarda diğer aksiyonları seçerek saf DRL tabanlı kişisel öğrenmeye ağırlık vermeleri mümkün olmuştur. Yapılmış olan deneyler sırasında kopyalama davranışının zaman içinde değişimi incelenmiş ve böylece, literatürde bulunan diğer araştırmalara kıyasla (örneğin (Ho & Ermon, 2016), (Yi vd., 2018)), kopyalama davranışının dinamik olarak akıllı etmen tarafından kontrol edildiği özgün bir DRL algoritması oluşturulmuştur. Bu doğrultuda, yöntem, birçok farklı parça üzerinde durularak geliştirilmiştir. Bunlardan ilki, SÖ katmanının içerdiği hafızanın yapısıdır. İlgili hafızanın yapısı ile alakalı aşağıda belirtilen farklı türler denenmiştir.

- Çok bölmeli statik hafıza: Bu hafıza yapısı, belirli sayıda davranış gösterimini içerir ve bu gösterimler, başlangıç ve bitiş noktası arasındaki en kısa yolların belli bir oranıyla alınan davranış gösterimlerinden oluşur. Kopyalama zamanı veya uygulama durumunda hafıza yapısında bir değişiklik olmadığından öğrenen etmen sabit olarak belirlenmiş hafıza üzerinde işlemlerini gerçekleştirir.
- Çok bölmeli dinamik hafıza: Bu hafıza yapısında belli bir sayıda kopyalanmış gösterim hafızada sıralı şekilde tutulur. İlgili sıralama kopyalama zamanı veya uygulama durumunda alınan geri bildirime göre oluşturulur. Hafıza alanının dolu olması ve yeni bir gösterimin kopyalanması durumunda, sıralama yöntemine göre en eski veya en düşük değerde geri bildirime sahip gösterim hafızadan atılır ve yeni gösterim boşalan pozisyona yerleştirilir. DPÖ katmanı "uygula" aksiyonunu ateşlendiğinde ve hafızadan bir gösterim seçilmesi gerektiğinde, bu seçim alınan olumlu veya olumsuz geri bildirimlerin oranına göre olasılıksal

olarak yapılır. Buna göre yüksek oranda olumlu geri bildirim alan gösterimler yüksek ihtimal ile uygulanmak üzere DPÖ katmanına önerilirken, yüksek oranda negatif geri bildirim alan gösterimler düşük ihtimal ile seçilir.

Yöntemin oluşturulması sırasında üzerinde durulan diğer bir konu, kopyalama sırasında hangi tür bilgilerin transfer edilmesi gerektiğidir. Tez kapsamında transfer edilmesi gereken bilginin içeriğine göre tek bir kopyalama türünden bahsetmek mümkündür:

- Sadece aksiyonları kopyalama: Bu yöntem ile etmen, aksiyonların gerçekleştirildiği çevresel durumlardan bağımsız olarak, sadece gösterilen aksiyonları kopyalar. Kopyalanan aksiyonların anlamlı olduğu çevresel durumlar, öğrenen etmen tarafından keşfedilir.

Üzerinde durulan son konu, SÖ katmanının içerdiği hafıza ile DPÖ katmanının kendi içindeki hafıza yapısının ilişkilendirilmesidir.

- DPÖ katmanında, etmen çevreyle etkileşime girdiğinde seçilen aksiyon sinir ağlarını eğitmek için kullanılır. Bu nedenle hafıza yapısının kullanılması, çok miktarda veri gerektiği için önemlidir. Aksiyonlar, veri eğitimi sırasında kullanıldığı için ve tez kapsamında sadece aksiyonların transferi yapıldığından DPÖ katmanındaki hafıza yapısı dolaylı yoldan etkilemektedir.
- DPÖ katmanından "kopyala" emri verildiğinde, SÖ katmanına geçen etmenin kontrolü sırasında etmenin hareketsiz kaldığı bir süreç boyunca, aynı ortamda bulunan ve DPÖ katmanında işlem yapan başka bir etmenin aksiyonlarını belirli bir süre boyunca hatırlayarak bir davranış gösterimini izler. Gözleme işlemini sonlandırdığında, gözlemlenen davranış gösterimleri SÖ katmanında bulunan hafıza alanına kaydedilir.

DPÖ katmanı tarafından “uygula” aksiyonunun seçilmesi ve kopyalanan gösterimin gerçekleştirilmesi durumunda SÖ katmanına sağlanacak olan içsel geri bildiriminin hesaplanmasıdır. Kopyalama esnasında tek bir aksiyon yerine bir aksiyon serisinden oluşan davranış bütünü kopyalanmıştır. İlgili davranış bütünü

bir vaka olarak alınmıştır ve vakanın başlangıcı ile sonu arasındaki değişim incelenmiştir. Bu bağlamda gözlemlenen ilk metrik kopyalanan davranışın gerçekleştirilmesi esnasında elde edilen ödüller olmuştur. Ancak sadece ödüllerin incelenmesi, tez kapsamında belirtildiği üzere bazı çevrelerde gözlemlenen seyrek ödül problemini çözmeyecektir. Bu sebeple, vaka öncesi ve sonrasında Derin Pekiştirmeli sinirsel ağdaki durum-aksiyon değerleri incelenmiştir ve aksiyon bazlı entropi hesabı yapılmıştır.

Entropi, bir sistemin düzensizlik veya belirsizlik derecesini ölçen ölçüdür ve bir sistemin olasılık dağılımının bir fonksiyonudur. Bir vakanın gerçekleşmesi durumunda, vaka boyunca ziyaret edilen tüm durumlar için kümülatif entropi hesabı yapılmıştır. Ziyaret edilen durumların her biri için seçebileceği 8 yönün Q değerleri DQN ağından elde edilerek entropi hesabına katılmıştır. Bu noktada dikkat edilmesi gereken husus DQN ağının negatif Q değerlerini üretebilmesidir. Fakat negatif değerler, belirsizlik veya düzensizliğin azalmasını göstermektedir. Bu nedenle, negatif değerler entropi kavramına aykırıdır. Geliştirilen yöntem kapsamında negatif Q değerlerini işleyebilmek için pozitif değerlerinde varlığıyla birlikte bir normalizasyon fonksiyonu oluşturulmuştur. Uygulanan normalizasyon fonksiyonunun adımları aşağıdaki gibidir:

1. Normalize edilecek 8 yöne ait Q değerleri bir dizi olarak alınır.
2. Elemanlarının negatif olup olmadığını kontrol edilir.
3. Negatifse tersi alınır.
4. Dizinin minimum ve maksimum değerleri bulunur ve toplamı alınır.
5. Bu değerleri kullanarak dizi [0, 1] aralığında normalize edilir.
6. Normalize edilen dizi entropi hesabında kullanılmak üzere geri döndürülür.

Kullanılan normalizasyon fonksiyonunun sözde kodu algoritma 1`de gösterilmiştir.

**Algoritma 1** Kullanılan normalizasyon işleminin sözde kodu

```
METHOD normalize_array
    PARAMETERS array
    array_ = boş liste
    Döngü, array'in içindeki her değer için
```

```
    Eğer value < 0
        Append (1 / (-1 * value)) to array_
    Değilse
        Append value to array_
min_value = array_ in içindeki minimum değer
max_value = array_ in içindeki maximum değer
max_ = max_value + min_value
normalized_array = boş liste
Döngü, array_'in içindeki her değer için
    Eğer max_ >= 0
            normalized_array' e (value / max_) ekle
    Değilse
        Eğer value < 0
            Append ((1 / (-1 * value)) / max_) to normalized_array
normalized_array listelerini bir çift olarak döndür
```

Buna göre belirli bir s durumundaki aksiyon seti A için entropi aşağıdaki şekilde hesaplanmıştır:

$$Entropi_s = -\sum_{a \in A} P_{s,a} \log(P_{s,a}) \tag{3.1}$$

Formülde $P_{s,a}$ , S durumunda a aksiyonunun seçilme olasılığını temsil etmektedir. Bir vakanın gerçekleşmesi durumunda, vaka boyunca ziyaret edilen tüm durumlar için kümülatif entropi hesabı yapılmıştır. Ziyaret edilen durumlarda aksiyon bazlı entropide düşüş yaşanması durumunda, durum-aksiyon uzayının vakanın gerçekleştirildiği kısmında, belli aksiyonların öne çıktığı ve akıllı etmenin yönlendirildiği şeklinde değerlendirilmiştir. Bu durum olumlu bir takviye sinyaline sebep olmuştur. Aksi durumda bir entropi artışı gözlemlendiğinde ise beklenilen şekilde bir yönlendirmenin oluşmadığı bir durum olarak değerlendirilmiş ve negatif takviye sinyali oluşmuştur. Ayrıca öğrenen etmenlere, SÖ ile elde edilen davranışların kişisel öğrenme ile edilen durum-aksiyon değerleri ile çelişmesi halinde, bu davranışları reddetme ve olumsuz takviye sinyali oluşturma imkânı verilmiştir. İlgili ölçümün, etmenin mevcut durum-aksiyon değerleri üzerindeki

değişim üzerinden yapılması, SÖ kaynaklı elde edilen bilgi ile etmenin kişisel öğrenme kaynaklı tecrübelerinin birleştirilmesini sağlaması amaçlanmıştır.

Entropi hesabı sırasında kullanılan vaka uzunluğu, SÖ katmanının içerdiği hafıza alanından "kopyala" emri sonucu kaydedilmiş olan bir davranış gösterimini "uygula" emri sırasında hafızadan olasılıksal hesaba istinaden seçilmektedir. "Uygula" emri sırasında vaka uzunluğundaki davranışların uygulanmasından sonra başlangıçtaki ile sondaki kümülatif entropi hesapları karşılaştırılmaktadır ve entropi düşüşü gözlemlediği sırada entropi düşüş sayacı 1 arttırılmaktadır. Entropi düşüş sayıları her yola karşılık gelen bilgiye istinaden bir dizide tutulmaktadır. Bununla birlikte entropi düşüşü olsa da olmasa da bu vaka uzunluğu uygulandığı için uygulanma sayacı 1 arttırılmaktadır. Entropi düşüşünde olduğu gibi uygulanma sayıları her yola karşılık gelen bilgiye istinaden bir dizide tutulmaktadır. Belirtilen iki dizi yapısı hafızadaki yolların seçilme olasılığında kullanılmaktadır. Kullanılan formül aşağıdaki gibidir:

$$(Entropi\ Düşüş\ Sayısı + 1)\ /\ (Uygulanma\ Sayısı + 1) \quad (3.2)$$

Kullanılan entropi hesabının sözde kodu algoritma 2`de gösterilmiştir.

**Algoritma 2** entropi hesabının sözde kodu

```
Method entropy_calculation(self, entropy):
    Parametre entropy
    temp = boş liste

    normalize_array = Method self.normalize_array
    entropy_value = Normalize edilmiş dizinin logaritması ile çarpılmış halinin
                                    toplamının -1 ile çarpılması şeklinde hesapla
    entropy_value listelerini bir çift olarak döndür
```

Seçilme olasılığı hesabı aşağıda belirtilen adımlar uygulanarak yapılmaktadır.

1. Entropi düşüş ve uygulanma sayısı değer çiftleri için her biri için oranları hesaplanır. Oranlar, (3.2) de belirtilen formülü kullanılarak hesaplanır.
2. Her oranı, tüm oranların toplamına bölerek normalleştirme işlemi yapılır.

3. 0 ile 1 arasında rastgele bir ondalıklı sayı üretilir.
4. Rulet tekerleği seçim mantığını uygulanır. Normalleştirilmiş oranların üzerinde döndürülür, belirlenen bir değişkende biriktirilir ve bu değişken üretilen rasgele sayıyı aştığında indeksi seçer.
5. Seçilen indekse hafızada karşılık gelen yol seçilir.

Özetle, rulet tekerleği sistemi uygunluk puanı yüksek olan vakaların daha yüksek bir olasılıkla seçilmesini sağlayacak şekilde yapılmaktadır. Tekerlek döndürüldüğünde, daha yüksek uygunluk puanına sahip vakaların daha geniş bir alana sahip olduğu ve daha yüksek bir olasılıkla seçileceği bir olasılık dağılımı oluşturmaktadır. Kullanılan seçilme olasılığı hesabının sözde kodu algoritma 3`de gösterilmiştir.

**Algoritma 3** Seçileme olasılığı hesabının sözde kodu

```
Metod probability_calculation
    Parametre index
    full_entropi = boş liste
    full_uygulama = boş liste
    full_directions, full_directions_index = Metod is_full_directions

    Döngü i, her bir indeks için full_directions_index içinde
       full_entropi listesine self.entropi_dususu[index][i] ekle
       full_uygulama listesine self.uygulama_sayisi[index][i] ekle

    self.ratios = [(entropi + 1) / (uygulama + 1) for entropi, uygulama in
                                          zip(full_entropi, full_uygulama)]
    randnum = random.uniform(0, 1)
    oran_sum = 0

    Döngü i, her bir indeks için self.ratios içinde
       oran_sum = oran_sum + self.ratios[i]

    Döngü i, her bir indeks için self.ratios içinde
```

```
        self.ratios[i] = self.ratios[i] / oran_sum

    temp = 0
    action = -1

    Döngü i, her bir indeks için self.ratios içinde
        temp = temp + self.ratios[i]
        Eğer temp > randnum ise
            action = i
            Döngüyü Kır

    yol = action
    yol_bilgisi = self.directions[yol]
    yol, yol_bilgisi listelerini bir çift olarak döndür
```

Seçileme olasılığı hesabında kullanılan full directions sınıfının sözde kodu algoritma 4`de gösterilmiştir.

**Algoritma 4** Seçileme olasılığı hesabınında kullanılan full directions sınıfının sözde kodu

```
Metod full_directions
    full_directions = boş liste
    full_directions_index = boş liste

    Döngü i, satir için her bir eleman için
        Eğer satir doluysa:
            satiri full_directions listesine ekle
            satirin indeksini full_directions_index listesine ekle
    full_directions ve full_directions_index listelerini bir çift olarak döndür
```

Görüleceği üzere tez kapsamında, kopyalamanın doğada kullanımına uygun olarak davranışların içsel geri bildirim ile değerlendirilmesi gerçekleştirilmektedir. Bu sayede farklı etmenlerin sergilemesi sonucu kopyalanan

davranışların etmenin öğrenme hızını destekleyici olanlarının tespit edilmesi amaçlanmaktadır.

# 4. DENEYLER

Bu bölümde özgün SDRL algoritmasının performansı ölçülmüştür. Bu ölçümler, malzeme ve yöntem kısmında belirtilen benzetim ortamında gerçekleştirilmiştir. Geliştirilecek olan kontrol sisteminin kopyalama türü, hafıza yapısı ve vaka uzunluğu bazlı farklı parametreleri bulunmaktadır. İlgili parametrelere eşlenecek olan değerlere/ayarlara bağlı olarak kontrol sisteminin farklı versiyonları oluşturulmuştur (örneğin, sadece aksiyonları kopyalayan çok bölmeli statik hafıza kullanan, vaka uzunluğu 5 aksiyon olan kontrol sistemi). İlgili kontrol sistemlerinin versiyonları ve benzetim ortamı Python programlama dili/ortamı kullanılarak oluşturulmuştur. Kontrol sisteminin her farklı versiyonu benzetim ortamında gerçekleştirilir ve malzeme ve yöntem bölümünde tanıtılan sürü zekâsı konusunda incelenmiş arama/toplama problemi üzerinde sınanmıştır. Bu amaçla her farklı versiyonun kullanıldığı, istatistiksel analize imkân verilerek yüksek sayıda (örneğin 100 kez) deney koşusu tamamlanmıştır. Bu sayede, incelenen her problem için ikili t-test ve benzeri istatistiksel yöntemler kullanılarak en yüksek performansı gösteren kontrol sistemi parametreleri tespit edilmiştir. Sonrasında uygulanan deneylerin aynı benzetim ortamında saf DRL algoritması ile karşılaştırılmıştır.

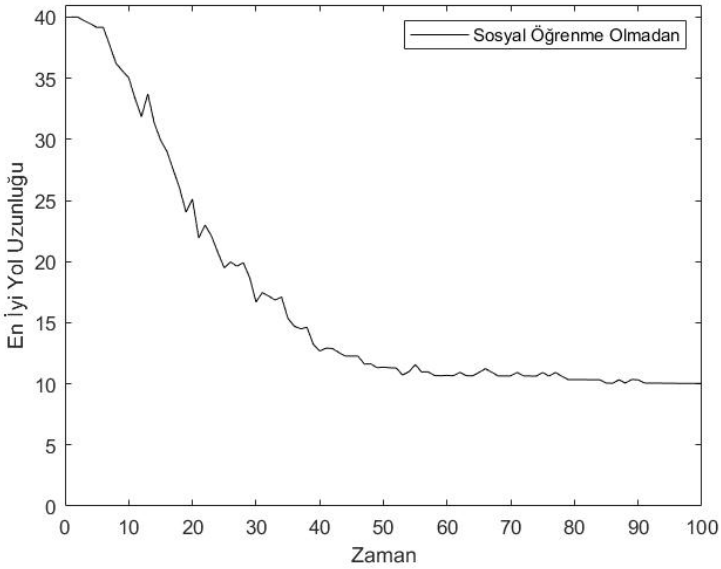
Her deney Şekil 3.1`deki benzetim ortamında değerlendirilmiştir. Etmenlerin her biri sol üst köşede bulunan B noktasına yerleştirilir ve hedef öğesi ise H noktasına yerleştirilir. Benzetim ortamında önceden belirlenen engeller bulunmaktadır.

Tez kapsamında deney koşularına dair üretilen, etmenlerin deney koşularına dair iç hesaplamaları ve SÖ kullanımına dair istatistiksel veriler sonradan farklı analiz araçları ile incelenmeye uygun olacak şekilde kayıt altına alınmıştır.

Derin pekiştirmeli öğrenme modelinin saf analizi, uzman izleme yoluyla kurulan öğrenme modelinin analizi, 8 yolun verilerek kurulan öğrenme modelinin analizi, tecrübeli etmeni gözlemleyerek öğrenme modelinin analizi ve birbirini gözlemleyerek öğrenme modelinin analizi olmak üzere 5 deney ortamında incelemeler yapılmıştır. Tezin buradan sonraki kısmında adım adım deney ortamları anlatılmıştır.

### 4.1 Derin Pekiştirmeli Öğrenme Modelinin Saf Analizi

Sosyal öğrenme olmaksızın DRL modeli test edilmiştir. Şekil 3.1 de görülen B başlangıç noktasından H hedef noktasına ulaşmak için derin Q-öğrenme kullanılarak bir etmen simüle edilmiştir. Etmen, benzetim ortamının sol üst köşesinden başlamakta ve her bir zaman biriminde sekiz komşu hücreden birine hareket edebilmektedir. Bir simülasyon çalışması, etmen benzetim ortamının H ile işaretlenmiş olan hedef öğeyi bulana kadar devam edebilmektedir. Öğrenen etmen, her adımda mevcut konumu için en yüksek Q değerini tahmin etmek için bir sinir ağı kullanılmıştır. Ardından, bu tahmine göre en yüksek Q değerine sahip olan eylemi seçmek için bir e-greedy algoritması kullanılmıştır. DQN, ε olasılığıyla rastgele bir eylem seçmektedir. 1 - ε olasılığıyla ise, Bellman denkleminin güncellenmiş Q değerini kullanarak en iyi eylemi seçmektedir. Bu strateji, etmenin keşfetme ve öğrenme yeteneklerini bir dengeleyerek, rastgele keşfetmeyi teşvik ederken aynı zamanda öğrenilen bilgileri kullanma olasılığını arttırmaktadır. Bu sayede, etmen geniş bir eylem uzayında etkili bir şekilde öğrenebilmekte ve optimize edilebilmektedir. Deney çalışması için, etmen eylemlerini seçerken rastgele keşfetme ve öğrenme ile daha önce öğrendikleri arasında bir denge kurmak için kullanılan epsilon (ε) değerin 0.3 olarak belirlenmiştir. Gelecekteki ödüllerin şu anki ödüllere göre ne kadar değerli olduğunu belirten gamma (γ) faktörü 0.9 olarak belirlenmiştir. Tahmin ve hedef ağları iki katmanlı bir yapay sinir ağından oluşmaktadır. Hedef ağı, tahmin ağına benzer, ancak eğitilmemektedir. Tahmin ağın güncellenme adımlarında kullanılan öğrenme oranı (α) 0.01 olarak belirlenmiştir. Tahmin ağın eğitiminde kullanılan batch size 128 olarak belirlenmiştir. Deneyim tekrar kullanımını sağlamak için kullanılan hafıza boyutu 21000 olarak belirlenmiştir. Her bir deneme 10000 zaman adımından oluşmaktadır. Her 100 lük zaman periyotlarında bir ölçüm yapılmıştır. Bu 100 lük periyotlardaki ortalama en iyi yol uzunluğunun değişimi Şekil 4.1`de gösterilmiştir.

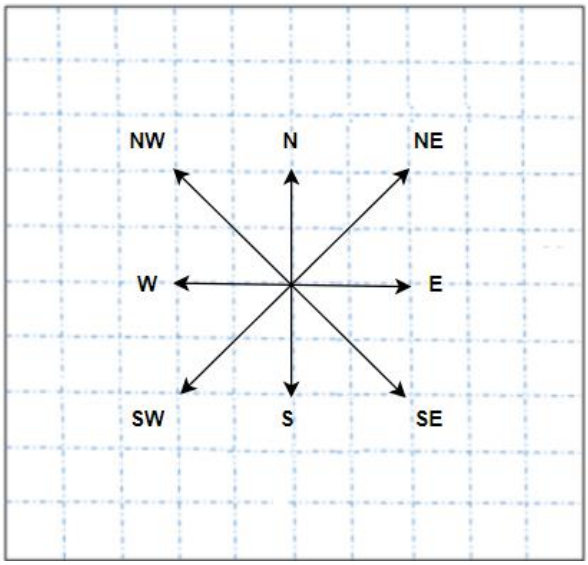



**Şekil 4.1.** Sosyal öğrenme olmaksızın DRL algoritmasının zamana göre en kısa yol uzunluğunun değişimi

### 4.2 8 Yolun Verilerek Kurulan Öğrenme Modelinin Analizi

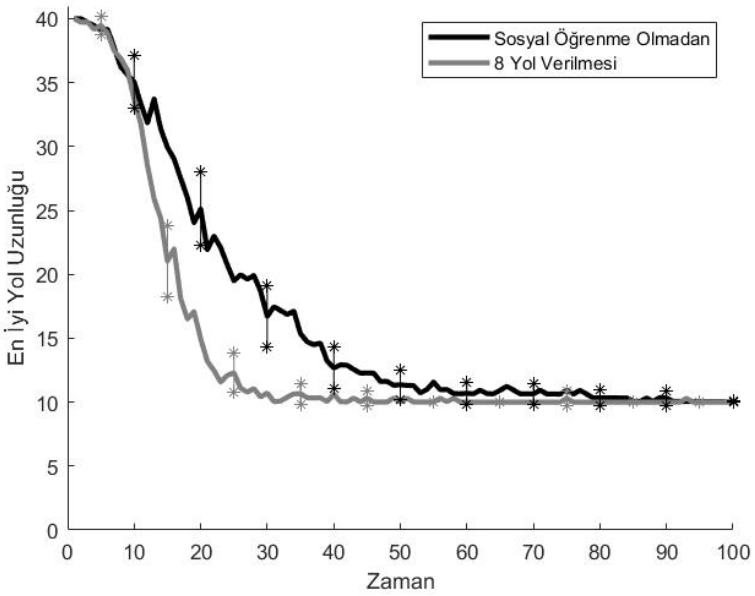
Algoritmanın faydalı yolları seçip seçemediğini ve bunları kullanarak öğrenme hızını artırıp artırmadığını kontrol etmek için öğrenme aracısına sekiz yol verilmiştir. Bunların her biri, Şekil 4.2'de gösterildiği gibi, pusulanın 8 farklı yönündeki 5 ardışık hareketi temsil edecek şekilde seçilmiştir. Deney sırasında aracının hafızasında aşağıdaki eylem setleri bulunmaktadır:

- $Yol_1$: N, N, N, N, N
- $Yol_2$: NE, NE, NE, NE, NE
- $Yol_3$: E, E, E, E, E
- $Yol_4$: SE, SE, SE, SE, SE
- $Yol_5$: S, S, S, S, S
- $Yol_6$: SW, SW, SW, SW, SW
- $Yol_7$: W, W, W, W, W
- $Yol_8$: NW, NW, NW, NW, NW



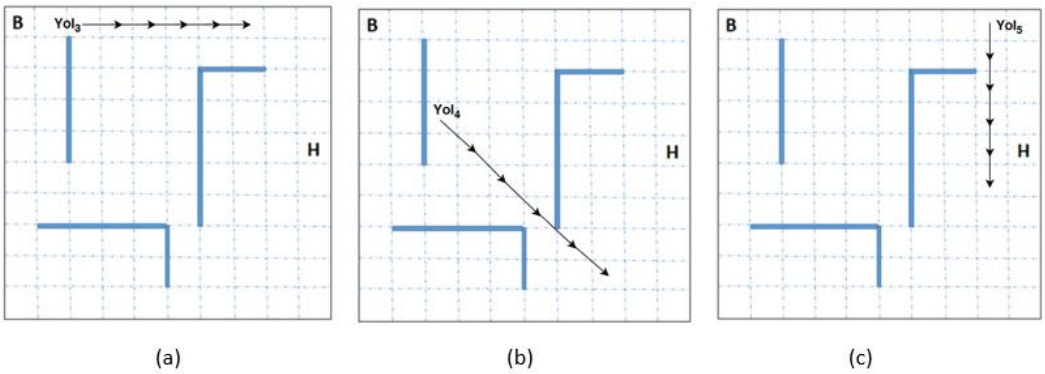

**Şekil 4.2.** Etmene verilen sekiz yol. Her eylem listesi tek yönde ardışık 5 hareketten oluşur: N, NE, E, SE, S, SW, W, NW

Kopyalama emri ateşlendiğinde başka bir etmen gözlemlenerek davranış bütünü taklit edilmeyeceğinden ve hafızada bir değişiklik yapılmayacağından dolayı kopyalama zaman maliyeti göz ardı edilmiştir. Uygula emri ateşlendiğinde çok bölmeli statik hafızada bulunan yolların yöntem kısmında bahsedilen formül ile olasılık hesabı sonucunda bir tanesi elde edilerek uygulanmakta ve zaman maliyeti göz önünde bulundurulmaktadır. 100 adet simülasyon denemesi yapılmıştır. Her bir deneme 10000 zaman adımından oluşmaktadır. Her 100 lük zaman periyotlarında bir ölçüm yapılmıştır. 8 yol verilerek taklit eden etmen ile taklit etmeyen (saf DRL) etmen arasındaki karşılaştırma ikili t-test ile ölçülmüştür. Taklit destekli öğrenme etmeninin performansını gösteren Şekil 4.3'e bakıldığında, etmenin sadece Q-öğrenmeye sahip öğrenmeyi önemli ölçüde daha önce en kısa yolu keşfettiği görülmektedir. Taklit ile öğrenme etmeni, yararlı yolları seçebildiği ve performansını arttırabildiği görülmektedir. İkili t-test, taklit eden etmenin 1300 ile 4800 zaman adımları arasında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. 100 deney çalışmasından elde edilen ortalama en iyi yol uzunluğu, her etmen için her 100 zaman biriminde mevcut Q değerleri üzerinde açgözlü bir algoritma kullanılarak hesaplanmıştır.



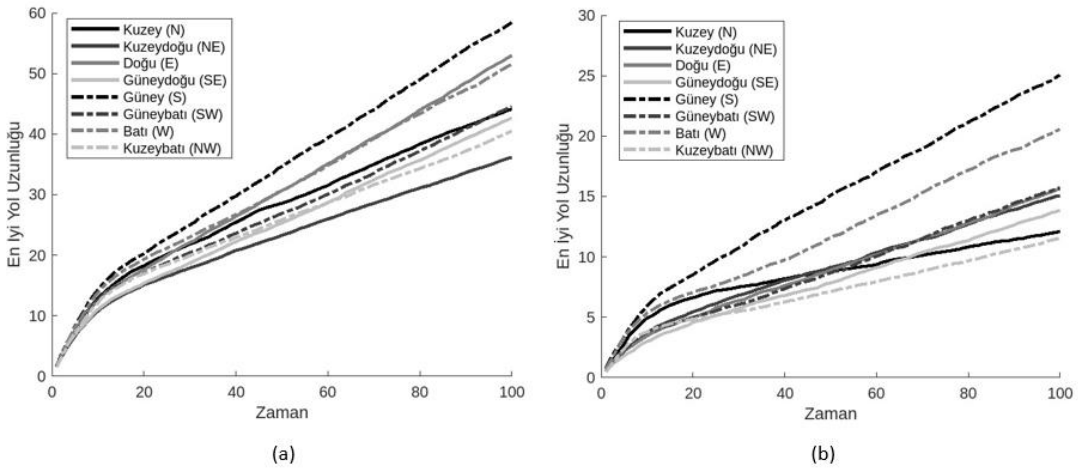

**Şekil 4.3.** 8 yol verilerek taklit eden ve taklit etmeyen etmen için en iyi yol uzunluğu. Sonuçlar, 100 deneme çalışmasının ortalaması olarak elde edilen en iyi yol uzunlukları ile birlikte %95 güven aralıklarını içerir.

Etmenin hafızasından olasılıksal olarak seçilen ve uygulanma sayısı ile entropi düşüş sayıları tutularak, hangi yolun daha çok tercih edildiği bilgisi olasılıksal olarak incelenmiştir. Şekil 4.4`de, 100 denemeden herhangi birinde en fazla olasılıkla seçilen üç yol örneği ($Yol_3$, $Yol_4$, $Yol_5$), benzetim ortamında rastgele belirlenmiş konumlar üzerinde gösterilmiştir. Doğu yolunun en yüksek uygulanma sayısına ve entropi düşüşüne sahip olduğu görülmüştür. Bu nedenle seçilme olasılığının daha yüksek olduğu görülmüştür. E, SE ve S yolları, diğer 5 yol ile karşılaştırıldığında göreceli olarak daha yüksek olasılık değerlerine sahiptir. Bu, etmenin performansını iyileştirmeye daha yatkın olan 3 yolu seçebildiğini doğrulamıştır.



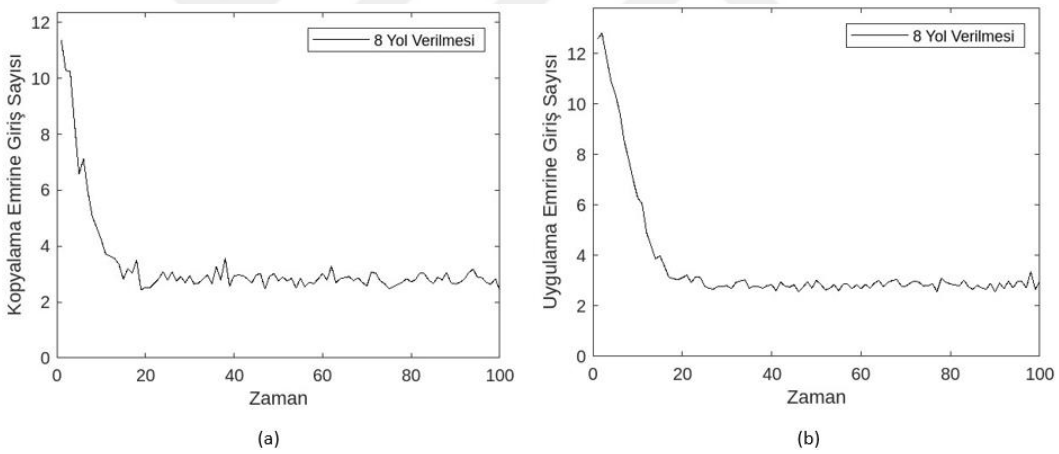

**Şekil 4.4.** 8 yol verilerek taklit eden etmenin benzetim ortamında rastgele belirlenmiş konumlar üzerinde, 100 denemeden herhangi birinde en fazla olasılıkla seçilen üç yol örneği. Hafızada bulunan $Yol_3$, 121 kez uygulanmış bunların 64 adetinde entropi düşüşü gözlenmiştir (a). $Yol_4$, 109 kez uygulanmış bunların 52 adetinde entropi düşüşü gözlenmiştir (b). $Yol_5$, 97 kez uygulanmış bunların 35 adetinde entropi düşüşü gözlenmiştir.

Yöntemimiz kapsamında kullandığımız entropi hesabının başarısını ölçebilmek için uygulanma ve entropi düşüşlerinin yollar üzerindeki etkisi incelenmiştir. Şekil 4.5`de yolların uygulanma sayısının etkisi incelendiğinde sırasıyla güney ve doğu yönlerinin en çok uygulandığı görülmüştür. Bununla doğru orantılı olarak yolların entropi düşüş oranları incelendiğinde yine güney ve doğu yönlerinin en çok olduğu görülmüştür. Dolayısıyla entropi hesabının yöntemimize katkısının olduğu ve seçilen yollardan kesitlerin başarılı olduğu görülmüştür.



**Şekil 4.5.** Entropi hesabının başarısını ölçebilmek için uygulanma ve entropi düşüşlerinin yollar üzerindeki etkisi. Yolların uygulanma sayısının etkisi (a) ve entropi düşüşün etkisi (b).

Kopyala ve uygula emirlerinin derin sinir ağı tarafından kontrol edilmesine dayalı yöntemimizin öğrenmeye etkisini incelemek amacıyla, kopyala ve uygula emirlerinin değişim grafikleri Şekil 4.6`da incelenmiştir. Aksiyon olarak kontrol edilmesi öğrenme sırasında negatif bir etkiye sebep olmadığı pozitif bir etkiye sebep olarak öğrenmeye destek olduğu görülmüştür. Kopyala ve uygula aksiyonlarının seçilme olasılıklarının zaman içinde azaldığı gözlemlenmiştir. Sonuçlar, 100 deneme çalışmasından elde edilen ortalama kopyala ve uygulama emirlerini yerine getirme sayılarını içermektedir. Ölçümler, her 100 zaman biriminde gerçekleştirilmiştir.




**Şekil 4.6.** 8 Yol verilerek taklit eden etmenin kopyala emri (a) ve uygula emrini (b) girişlerinin zaman içindeki değişimi

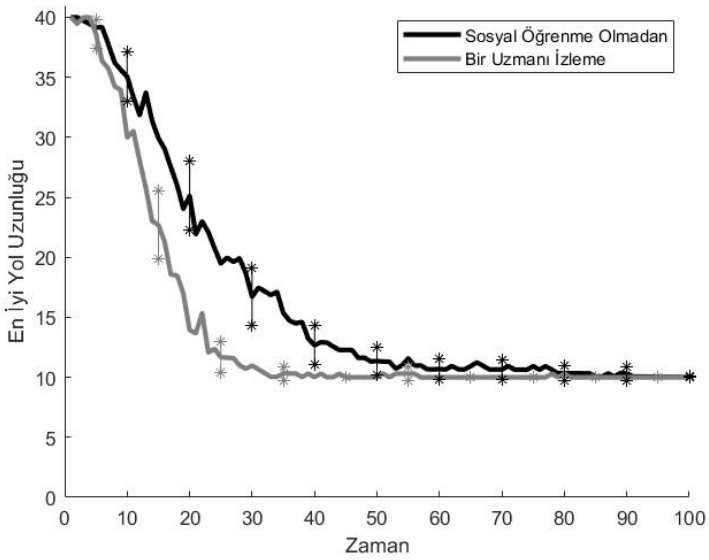
### 4.3 Uzman İzleme Yoluyla Kurulan Öğrenme Modelinin Analizi

Taklit etmenin önemli yönlerinden biri olan deneyimin taklit yoluyla diğer bireylere aktarılması olanağından yola çıkılarak oluşturulmuştur. Şekil 3.1 de görülen B başlangıç noktasından H hedef noktasına en kısa ulaşılan yollardan 8 farklı 5 ardışık hareketi temsil edecek şekilde faydalı kesitler seçilmiştir. Deney sırasında aracının hafızasında aşağıdaki eylem setleri bulunmaktadır:

- $Yol_1$: E, E, E, E, E

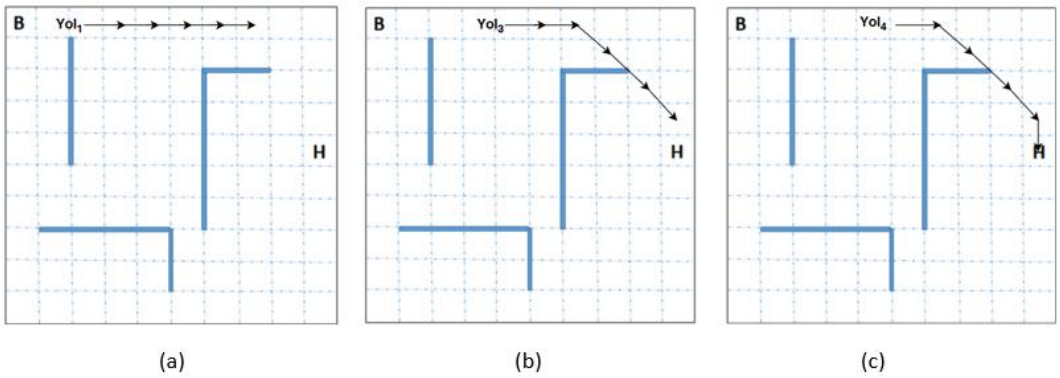
- $Yol_2$: E, E, E, E, SE
- $Yol_3$: E, E, SE, SE, SE
- $Yol_4$: E, SE, SE, SE, S
- $Yol_5$: SE, NE, SE, NE, SE,
- $Yol_6$: NE, SE, NE, SE, E
- $Yol_7$: SE, NE, SE, E, E
- $Yol_8$: SE, E, E, SE, SE

Unutulmaması gereken bir nokta, en kısa yollardan alınan kesitlerin, taklit eden etmenin bu eylemlerin hangi durumda (veya benzetim ortamının hangi konumunda) anlamlı olduğu konusunda herhangi bir bilgiye sahip olmamasıdır. Benzetim ortamında tek bir etmen çalışmaktadır. Kopyalama emri ateşlendiğinde başka bir etmenden davranış bütünü kopyalanmayacağı ve hafızada bir değişiklik yapılmayacağından dolayı kopyalama maliyeti göz ardı edilmiştir. Taklit eden etmen en fazla 8 yolu hafızasında saklayabilmektedir. Uygula emri ateşlendiğinde çok bölmeli statik hafızada bulunan yolların olasılık hesabı sonucunda bir tanesi elde edilerek uygulanmıştır. 100 adet simülasyon denemesi yapılmıştır. Her bir deneme 10000 zaman adımından oluşmaktadır. Her 100 lük zaman periyotlarında bir ölçüm yapılmıştır. Bir uzmanı taklit eden etmen ile taklit etmeyen (saf DRL) etmen arasındaki karşılaştırma ikili t-test ile ölçülmüştür. Şekil 4.7`de görüleceği üzere taklitle geliştirilmiş öğrenme etmeni en kısa yolu sadece Q-öğrenmesine sahip etmenden çok daha önce bulmaktadır. İkili t-test, taklit eden etmenin 1200 ile 5000 zaman adımları arasında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. Sonuçlar 100 deney çalışmasından elde edilen ortalama en iyi yol uzunluğudur. Her bir etmen için en iyi yol, mevcut Q değerleri üzerinde açgözlü bir algoritma kullanılarak her 100 zaman biriminde hesaplanmıştır.



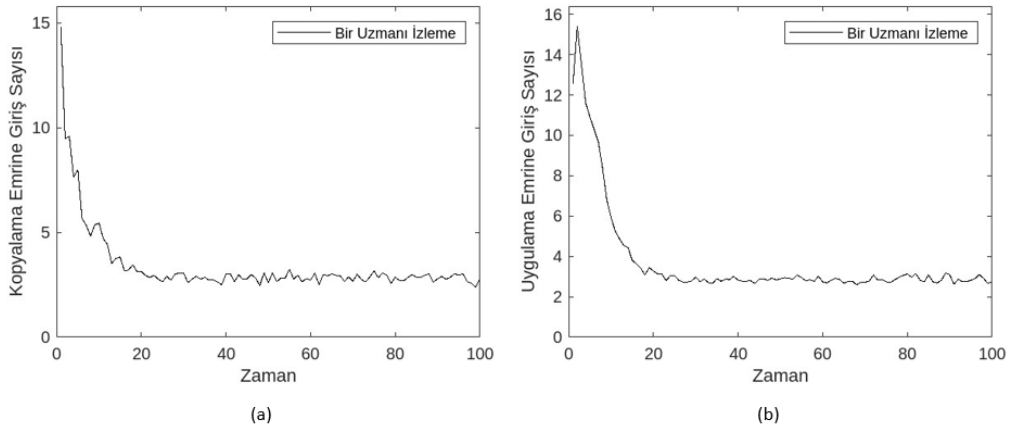

**Şekil 4.7.** Bir uzmanı taklit eden ve taklit etmeyen etmen için en iyi yol uzunluğu. Sonuçlar, 100 deneme çalışmasının ortalaması olarak elde edilen en iyi yol uzunlukları ile birlikte %95 güven aralıklarını içerir.

Etmenin hafızasından olasılıkla seçilen yolların uygulanma sayısı ve entropi düşüş sayıları incelenerek, hangi yolun daha çok tercih edildiği belirlenmiştir. Şekil 4.8`de, benzetim ortamında rastgele belirlenmiş konumlar üzerinde, 100 denemeden herhangi birinde en fazla olasılıkla seçilen üç yol örneği ($Yol_1$, $Yol_3$, $Yol_4$) gösterilmiştir. $Yol_4$ en yüksek uygulanma sayısına ve entropi düşüşüne sahip olduğu görülmüştür. Statik hafızadaki yollar, 8 yoldan farklı olarak faydalı kesitlerden oluştuğu için, uzman izleme deneyinde uygulanma ve entropi düşüş değerleri birbirlerine yakındır. Bu nedenle, seçilme olasılıklarındaki yakın farklara rağmen, yol seçiminde daha yüksek olasılıkla seçilen yolların olduğu görülmüştür. Bu, etmenin performansını iyileştirmeye daha yatkın olan yolu seçebildiğini doğrulamıştır.




**Şekil 4.8.** Bir uzmanı taklit eden etmenin benzetim ortamında rastgele belirlenmiş konumlar üzerinde, 100 denemeden herhangi birinde en fazla olasılıkla seçilen üç yol örneği. Hafızada bulunan $Yol_1$, 100 kez uygulanmış bunların 41 adetinde entropi düşüşü gözlenmiştir (a). $Yol_3$, 91 kez uygulanmış bunların 33 adetinde entropi düşüşü gözlenmiştir (b). $Yol_4$, 102 kez uygulanmış bunların 41 adetinde entropi düşüşü gözlenmiştir.

Kopyala ve uygula emirlerinin derin sinir ağı tarafından kontrol edilmesine dayalı yöntemimizin öğrenmeye etkisini incelemek amacıyla, kopyala ve uygula emirlerinin değişim grafikleri Şekil 4.9`da incelenmiştir. Sonuçlar 100 deney çalışmasından elde edilen ortalama kopyala ve uygulama emirlerini yerine getirme sayılarıdır. Her 100 zaman biriminde yapılan ölçümlerle, kopyala ve uygula aksiyonlarının seçilme olasılıklarının zaman içinde azaldığı gözlemlenmiştir. Aksiyon olarak kontrol edilmesi öğrenme sırasında negatif bir etkiye sebep olmadığı pozitif bir etkiye sebep olarak öğrenmeye destek olduğu görülmüştür.



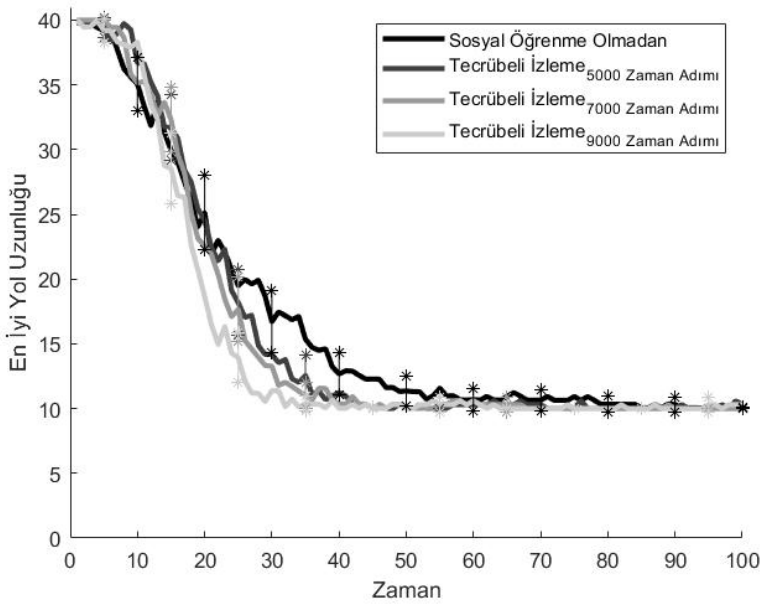

**Şekil 4.9.** Bir uzmanı taklit eden etmenin kopyala emri (a) ve uygula emrini (b) girişlerinin zaman içindeki değişimi

### 4.4 Tecrübeli Etmeni Gözlemleyerek Öğrenme Modelinin Analizi

Tecrübeli bir etmeni taklit eden durumda, bir uzmanın yerine hafif derecede tecrübeli bir etmenin varlığının etkisi test edilmiştir. Başlangıçta tek bir etmen, bir süre boyunca (5000 - 7000 - 9000 zaman adımı) benzetim ortamında eğitilmiştir. Bu nedenle ilk etmen taklit olmadan çalıştırılmıştır. Daha sonra ikinci bir etmen devreye sokulmuştur. İkinci etmen tecrübeli ilk etmeni takip edebilmektedir. Taklit ile geliştirilmiş olan DRL algoritması kullanılarak takip işlemi gerçekleştirilmiştir. Unutulmamalıdır ki, tecrübeli etmen en iyi yolu bulsa bile, e-greedy algoritması nedeniyle hala rastgele hareketler yapabilmektedir. İlk etmen için bir hafıza alanı bulunmamaktadır. İkinci etmen ilk kez devreye girdiği sırada boş bir hafıza ile öğrenmeye başlamakta ve ilk etmeni gözlemleme yeteneğine sahiptir. Taklit eden etmen en fazla 10 yolu hafızalarında saklayabilmektedir. Hafızası dolduğu anda, yeni gösterim kopyalanmak istendiğinde hafızasındaki en düşük olasılıklı yol silinmekte ve hafızasında boşalan pozisyona yeni yol kayıt edilmektedir. Hafıza yapısının bu şekilde kullanılması, çok bölmeli dinamik hafıza yapısına bir örnektir. Her iki etmen içinde her bir deneme 10000 zaman adımından oluşturulmuştur. Ancak ikinci etmen, belirli bir süre sonra çalışmaya başlayacağından dolayı, ilk etmen zaman adımını doldurup ikinci etmenin hala öğrenmeye devam etme olasılığı ortaya çıkmıştır. Bu durum ikinci etmen için risk oluşturmaktadır. Çünkü ilk etmen deneme işlemini sonlandırdığı için ikinci etmenin gözlemleme yapabileceği bir etmen benzetim ortamında kalmayacaktır. Bu soruna bir çözüm üretebilmek amacıyla ilk etmenin 10000 lik zaman adımı sonlandığında ikinci etmen kalan zamanında gözlemleme işlemini yani kopyalama işlemini yapmaktan vazgeçecektir. Kopyala emri devre dışı kalacaktır. Sadece hafızasında bulunan yolları uygula emri ateşlendiğinde uygulayabilir konuma gelecektir.

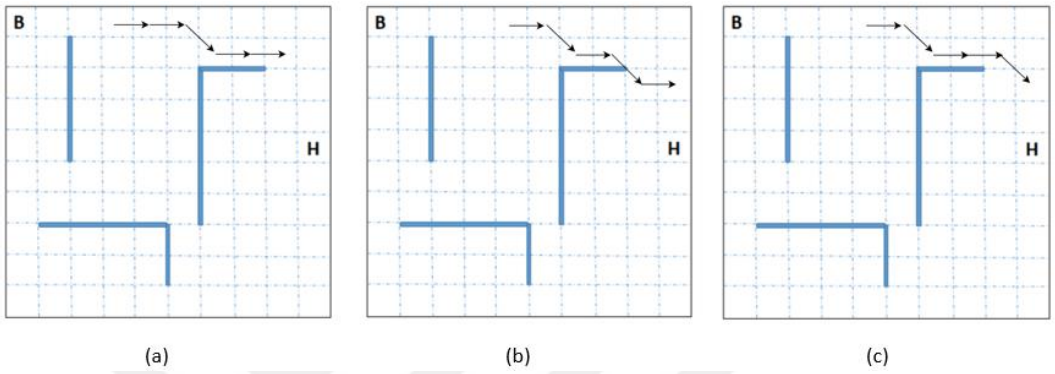
Üç farklı zaman adımı üzerinde deney çalışması yapılmıştır. Tecrübeli bir etmeni taklit eden etmen ile taklit etmeyen etmen arasındaki karşılaştırma ikili t-test ile ölçülmüştür. Ortalama en iyi yol uzunluğu, her etmen için her 100 zaman biriminde, 100 deney çalışmasının sonuçlarından hesaplanmıştır. Şekil 4.10`da ölçümler incelendiğinde 5000, 7000 ve 9000 zaman adımları için yapılan karşılaştırmalar görülmektedir. 5000 zaman adımı sonra benzetim ortamına bırakılan etmen, 3100 ile 3400, 3600 ile 3900 ve 4200 ile 4800 zaman adımları aralıklarında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür.

7000 zaman adımı sonra benzetim ortamına bırakılan etmen, 2600 ile 4100 ve 4200 ile 4700 zaman adımları aralıklarında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. 9000 zaman adımı sonra benzetim ortamına bırakılan etmen, 2000 ile 4700 zaman adımları aralıklarında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. 5000 ile 7000 zaman adımları arasında bir karşılaştırma yapıldığında tek bir nokta (9000. zaman adımı) dışında fark olmadığı görülmüştür. 5000 ile 9000 zaman adımları arasında bir karşılaştırma yapıldığında 1800 ile 3200 zaman adımları aralıklarında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. 7000 ile 9000 zaman adımları arasında bir karşılaştırma yapıldığında 2500 ile 2900 zaman adımları aralıklarında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. Belirtilen karşılaştırmalın incelenmesinin ardından 9000 zaman adımı sonra benzetim ortamına bırakılan etmenin 7000 ve 5000 zaman adımı deneylerinden daha iyi sonuç verdiği görülmüştür. Bu nedene 9000 zaman adımı ile tecrübeli öğrenen etmenin kullanılmasına karar verilmiştir. Ortalama en iyi yol uzunluğunun zamanla azaldığı görülmüştür.



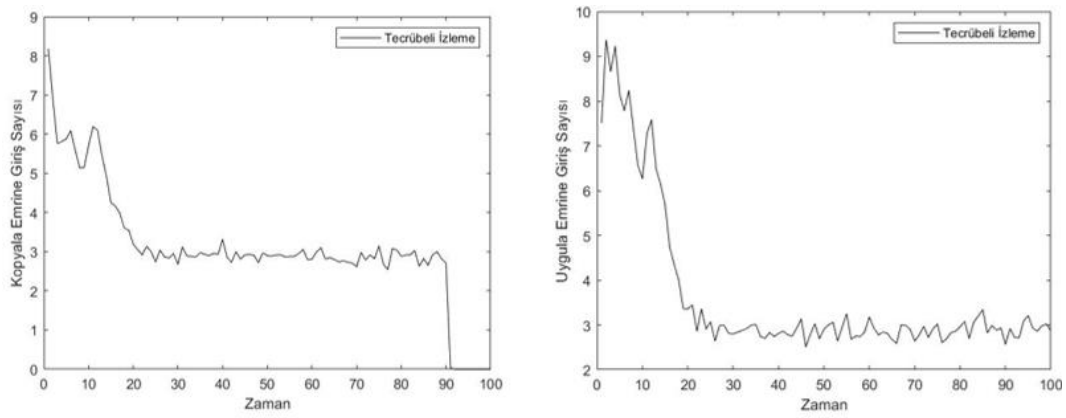

**Şekil 4.10.** Tecrübeli bir etmeni taklit eden etmenin zamana göre en kısa yol uzunluğunun değişimi. Sonuçlar, 100 deneme çalışmasının ortalaması olarak elde edilen en iyi yol uzunlukları ile birlikte %95 güven aralıklarını içerir.

Hangi yolun daha çok tercih edildiği olasılıksal olarak incelenmesi için uygulanma sayısı ve entropi düşüş değerleri tutulmuştur. Şekil 4.11 benzetim ortamında rastgele belirlenmiş konumlar üzerinde, 9000 zaman adımı sonra simülasyona başlayan ikinci etmen için 100 denemeden herhangi birinde deneme sonunda en fazla olasılıkla seçilen üç yol örneğini gösterilmektedir.




**Şekil 4.11.** Tecrübeli bir etmeni taklit eden etmenin benzetim ortamında rastgele belirlenmiş konumlar üzerinde, 100 denemeden herhangi birinde en fazla olasılıkla seçilen üç yol örneği. 83 kez uygulanmış bunların 38 adetinde entropi düşüşü gözlenmiştir (a). 69 kez uygulanmış bunların 29 adetinde entropi düşüşü gözlenmiştir (b). 61 kez uygulanmış bunların 24 adetinde entropi düşüşü gözlenmiştir.

Kopyala ve uygula emirlerinin derin sinir ağı tarafından kontrol edilmesine dayalı yöntemimizin öğrenmeye etkisini incelemek amacıyla, Şekil 4.12`de etmenin kopyala ve uygula emirlerinin değişim grafikleri incelenmiştir. Sonuçlar, 100 deneme çalışmasından elde edilen ortalama kopyala ve uygulama emirlerini yerine getirme sayılarını içermektedir. Ölçümler, her 100 zaman biriminde gerçekleştirilmiştir. Grafik incelendiğinde kontrolün aksiyonlar üzerinde etkili bir şekilde kullanılmasının öğrenme sürecinde olumsuz bir etkisi olmadığını, aksine öğrenmeye olumlu bir katkıda bulunduğunu göstermektedir. Kopyala ve uygula aksiyonlarının seçilme olasılıklarının zaman içinde azaldığı gözlemlenmiştir. Fakat kopyala emrine giriş grafiğinde kopyala işlemini belirli bir süre sonra bıraktığı görülmüştür. Takip eden etmenin belirli bir süre sonra simülasyon ortamına alınarak başlatılmasından kaynaklı olarak belirli bir süre sonra takip edebileceği bir deneyimli etmen ortamda kalmayacağından ötürü kopyala emrini seçemeyeceğini ifade etmektedir.

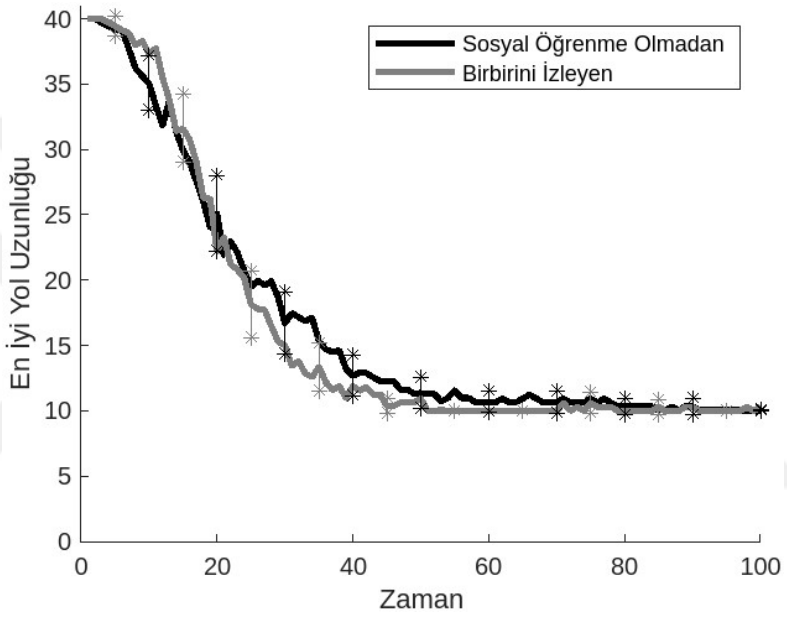



**Şekil 4.12.** Tecrübeli bir etmeni taklit eden etmenin kopyala emri (a) ve uygula emrini (b) girişlerinin zaman içindeki değişimi

## 4.5 Birbirini Gözlemleyerek Öğrenme Modelinin Analizi

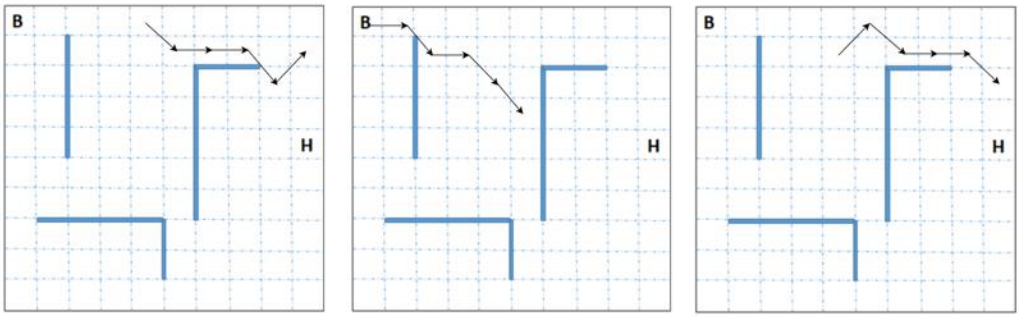
Etmenlerin birbirini kopyaladığı durumu düşünelim. Ancak deneyimlerini paylaşacak bir uzman etmen olmadığında ne olurdu? Bu senaryoyu simüle etmek amacıyla, benzetim ortamında iki etmen bulunmaktadır; her ikisi de aynı anda öğrenmeye başlamakta ve birbirlerini gözlemleme yeteneğine sahiptirler. Etmenler arasındaki ufak farklılıklar, birbirini taklit eden etmenlerin deneyim boyunca birbirlerini etkilemelerinden kaynaklanmaktadır. Bu durum, birinin biraz daha iyi performansa sahip olması diğerini iyileştirmesine olanak tanıyabilmektedir. Sürekli etkileşimleri, bir noktadan sonra birbirlerine daha az yarar sağlamaya başlamaktadır. Herhangi bir etmenin kopyala emri ateşlendiğinde diğer etmenin kopyala emri pasif konuma gelmekte ve kopyala emrindeki etmen işlemine sonlandırdığında aktif konuma gelmektedir. İki etmende kendilerine özgü boş hafıza ile başlamakta ve kopyala emri ile gözlemledikleri davranış gösterimlerini hafızalarına kayıt etmektedirler. Birbirini taklit eden etmenler en fazla 10 yolu hafızalarında saklayabilirler. Hafızaları dolduğu anda, yeni gösterim kopyalanmak istediğinde hafızasındaki en düşük olasılıklı yol silinmekte ve hafızasında boşalan pozisyona yeni yol kayıt edilmektedir. Hafıza yapılarının bu şekilde kullanılması çok bölmeli dinamik hafıza yapısına örnektir. Bu süreçte hafızasında silinen yola karşılık gelen entropi düşüş ve uygulanma sayısını tutan verilerde sıfırlanmaktadır. Böylece yeni eklenen yol için olasılık hesabı sıfırlanarak başlanmış olur. Her iki etmende her 100 simülasyon çalışmasında birbirlerinden yeni bir yol kopyalar yani taklit ederler. İki etmeninde taklit ettiği yolları hafızasında saklamaktadır. Her bir

deneme 10000 zaman adımından oluşmaktadır. Taklit edene ile taklit etmeyen etmen arasındaki karşılaştırma ikili t-test ile ölçülmüştür. Ortalama en iyi yol uzunluğu, her etmen için her 100 zaman biriminde, 100 deney çalışmasının sonuçlarından hesaplanmıştır. Şekil 4.13, taklit eden etmenin, 3100 ile 4000 ve 3700 ile 3900 zaman adımları arasında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. Ortalama en iyi yol uzunluğunun zamanla azaldığı görülmüştür.



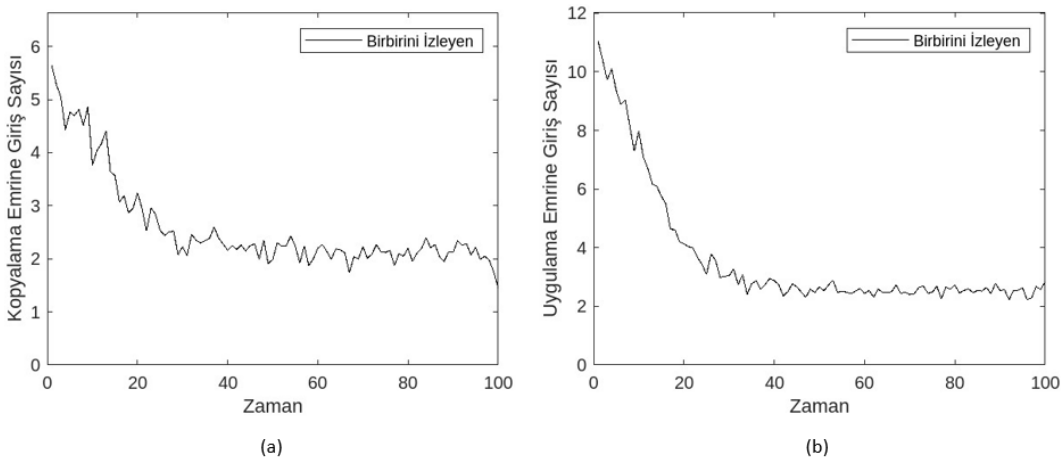

**Şekil 4.13.** İki ajanın birbirini taklit ettiği ve taklit etmeyen etmen için en iyi yol uzunluğu. Sonuçlar, 100 deneme çalışmasının ortalaması olarak elde edilen en iyi yol uzunlukları ile birlikte %95 güven aralıklarını içerir.

Etmenin hafızasından olasılıksal olarak seçilen yolların uygulanma sayısı ve entropi düşüş değerleri tutulmuştur. Böylece, hangi yolun daha çok tercih edildiği olasılıksal olarak incelenmiştir. Şekil 4.14`de benzetim ortamında rastgele belirlenmiş konumlar üzerinde, etmen için 100 denemeden herhangi birinde deneme sonunda en fazla olasılıkla seçilen üç yol örneği gösterilmiştir. Taklit eden etmenlerin bu eylemlerin hangi durumda veya benzetim ortamının hangi konumunda anlamlı olduğu konusunda hiçbir bilgisi olmadığını unutmamak önemlidir. Bu nedenle hafızada tutulan birçok faydalı yol olabileceğinden yakın olasılıklara sebep olabildiği görülmüştür.




**Şekil 4.14.** İki etmenin birbirini taklit ettiği benzetim ortamında rastgele belirlenmiş konumlar üzerinde, ilk etmen için 100 denemeden herhangi birinde en fazla olasılıkla seçilen üç yol örneği.

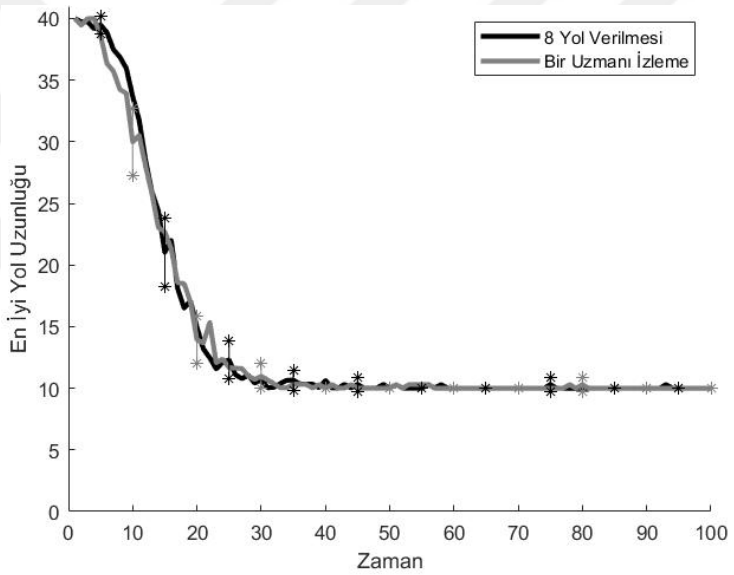
Kopyala ve uygula emirlerinin derin sinir ağı tarafından kontrol edilmesine dayalı yöntemimizin öğrenmeye etkisini incelemek amacıyla, Şekil 4.15`de iki etmenin kopyala ve uygula emirlerinin değişim grafikleri incelenmiştir. Bu grafikler, aksiyon olarak kontrol edilmesinin öğrenme sırasında negatif bir etkiye neden olmadığını, aksine pozitif bir etkiye neden olarak öğrenmeye destek olduğunu göstermektedir. Kopyala ve uygula aksiyonlarının seçilme olasılıklarının zaman içinde azaldığı gözlemlenmiştir. Sonuçlar, 100 deneme çalışmasından elde edilen ortalama kopyala ve uygulama emirlerini yerine getirme sayılarını içermektedir. Ölçümler, her 100 zaman biriminde gerçekleştirilmiştir.




**Şekil 4.15.** İki etmenin birbirini taklit ettiği ilk etmenin kopyala emri (a) ve uygula emrini (b) girişlerinin zaman içindeki değişimi

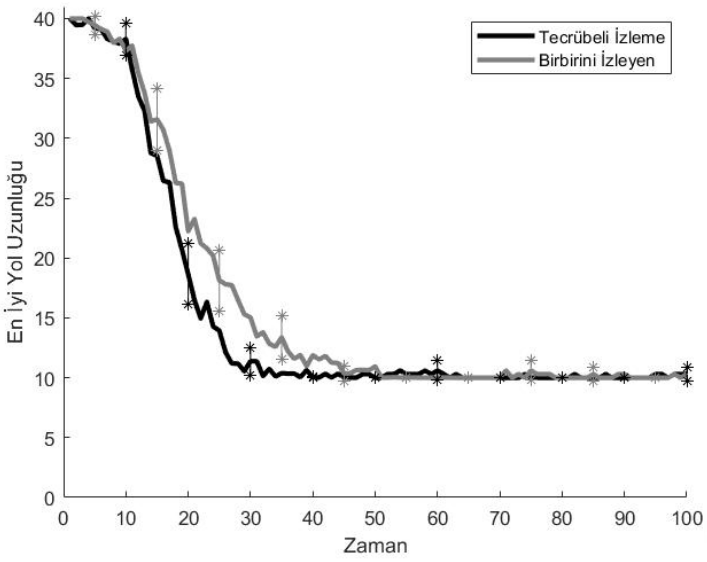
# 5. BULGULAR

Çalışmamızda uyguladığımız beş farklı yöntem ve iki farklı hafıza tipi için 3 farklı karşılaştırma yapılmıştır. Çok bölmeli statik hafıza yapısına uygun olacak şekilde düzenlenen 8 yol ve uzman izleme deneylerinin karşılaştırması Şekil 5.1'de gösterilmiştir. Karşılaştırma yapıldığında uzmanı izleyen etmenin 8 yol verilerek öğrenen etmen göre tek bir nokta (1000. zaman adımı) dışında istatiksel farkı olmadığı görülmüştür. 8 yol verilerek öğrenen etmenin uzmanı izleyen etmene göre tek bir nokta (2200. zaman adımı) dışında istatiksel farkı olmadığı görülmüştür. Bu durumda uzman ve 8 yolun hemen hemen benzer başarı verdikleri görülmüştür.



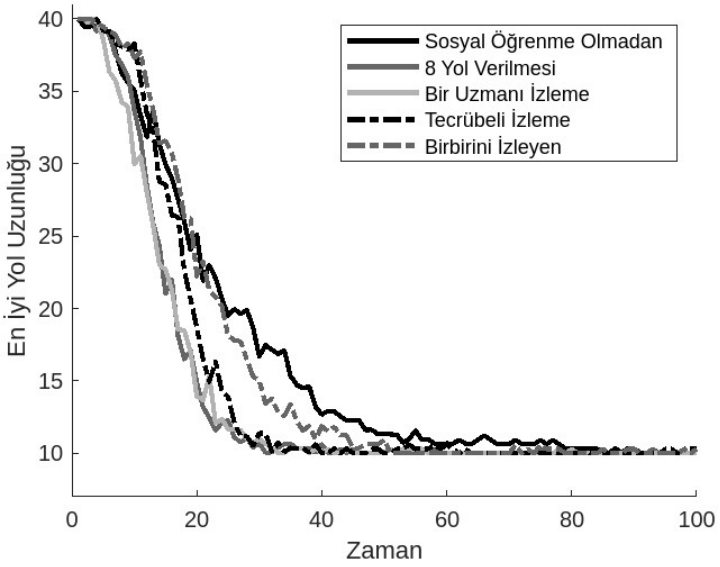

**Şekil 5.1.** 8 yol ve uzman izleme deneylerinin karşılaştırması. Sonuçlar, 100 deneme çalışmasının ortalaması olarak elde edilen en iyi yol uzunlukları ile birlikte %95 güven aralıklarını içerir.

Çok bölmeli dinamik hafıza yapısına uygun olacak şekilde düzenlenen tecrübeli etmeni gözlemleme ve birbirinden gözlemleyerek öğrenme deneylerinin karşılaştırması Şekil 5.2'de gösterilmektedir. Tecrübeli bir etmeni izleyen etmenin, 2100 ile 3000 ve 3200 ile 3600 zaman adımları aralıklarında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. Belirtilen aralıklarda tecrübeli bir etmeni izleyen etmenin en kısa yolu birbirini gözlemleyerek öğrenen etmene göre daha önce bulduğu görülmüştür.



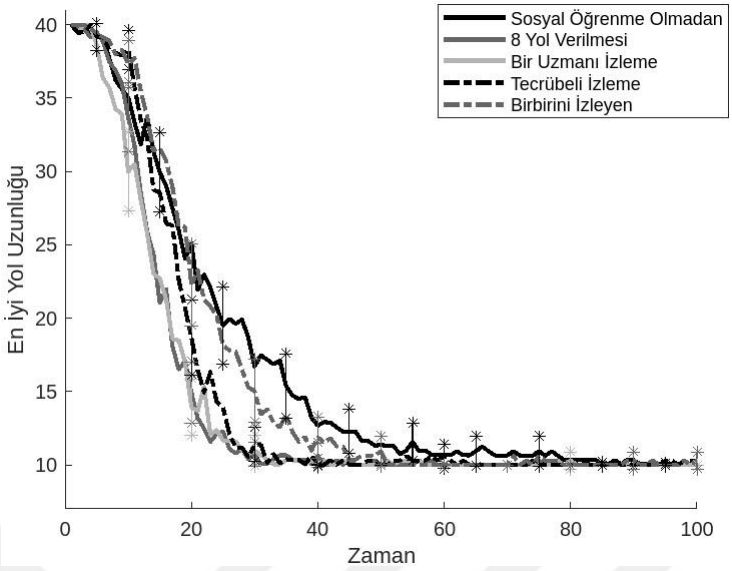

**Şekil 5.2.** Tecrübeli etmeni gözlemleme ve birbirinden gözlemleyerek öğrenme deneylerinin karşılaştırılması. Sonuçlar, 100 deneme çalışmasının ortalaması olarak elde edilen en iyi yol uzunlukları ile birlikte %95 güven aralıklarını içerir.

Tüm senaryolar Şekil 5.3` de karşılaştırmaktadır: bir etmenin sosyal öğrenme olmaksızın DRL algoritmasında çalışılan durum, bir etmenin sekiz yol verildiği durum, bir uzmanı izleyen bir etmen, tecrübeli bir etmeni izleyen bir etmen ve birbirlerini gözlemleyen bir etmen. Bir uzmanı izleyen etmen ile 8 yol verilerek öğrenen etmenin başarıları hemen hemen aynı olduğu görülmüştür. Bunlardan biraz daha az geç kısa yol bulan tecrübeli bir etmeni izleyen etmen deneyi olmuştur. Başlarda aralarında benzerlikler olan fakat tecrübeli etmeni izleyene göre daha geç en kısa yolu bulan birbirini izleyen etmen deneyi olmuştur. Dört deney setinde de, saf DRL modelinin en düşük performans gösterdiği durumdan belirgin bir şekilde daha üstün bir performans sergilediği gözlemlenmiştir.




**Şekil 5.3.** Beş senaryo deneyi için en iyi yol uzunluğu. Zamana göre en kısa yol uzunluğunun değişimi

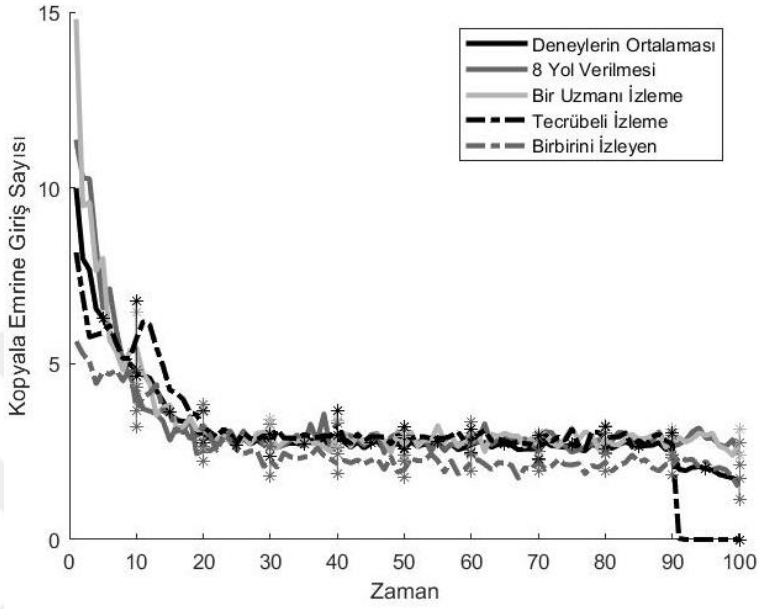
Tüm senaryolar t-test ölçümü ile karşılaştırılmış hali Şekil 5.4` de gösterilmiştir. Beklendiği gibi, birbirini gözlemleyen etmenin, sosyal öğrenme olmadan öğrenen etmene göre 3100 ile 4000 ve 3700 ile 3900 zaman adımları arasında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. Tecrübeli bir etmeni izleyen etmen, birbirini gözlemleyerek öğrenen etmene göre 2100 ile 3000 ve 3200 ile 3600 zaman adımları arasında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. 8 yol verilerek taklit eden etmen, tecrübeli bir etmeni izleyen etmene göre 1000 ile 2100 zaman adımları arasında istatistiksel olarak daha iyi bir performansa sahip olduğu görülmüştür. Bir uzmanı izleyen etmen, 8 yol verilerek taklit eden etmene göre iki nokta (600. zaman adımı ve 1000. zaman adımı) dışında istatiksel farkı olmadığı görülmüştür. Bir uzmanı taklit eden etmen, çok az bir fark ile diğer tüm deneyleri geride bıraktığı görülmüştür. Sekiz yol verilerek öğrenen etmenin hemen hemen biz uzmanı izleyen etmen ile benzer ölçüde başarıya sahip olduğu görülmüştür. Sırasıyla zaman adım aralıklarına bakıldığında sekiz yol deneyinden sonra tecrübeli öğrenmenin daha başarılı olduğu hatta bir uzaman kadar olmasa bile tecrübelerden fazlaca yaralandığı aradaki zaman farkından anlaşılmıştır. Birbirini gözlemleyen etmen, bu üç deneyin gerisinde kaldığı görülmüştür. Bahsedilen dört deneyde sosyal öğrenme olmaksızın saf DRL modelinin en kötü durumundan önemli ölçüde daha iyi performansa sahip oldukları görülmüştür.




**Şekil 5.4.** Beş senaryo deneyinin t-test ölçümü ile karşılaştırılması. En iyi yol uzunluğu zamana göre değişimi

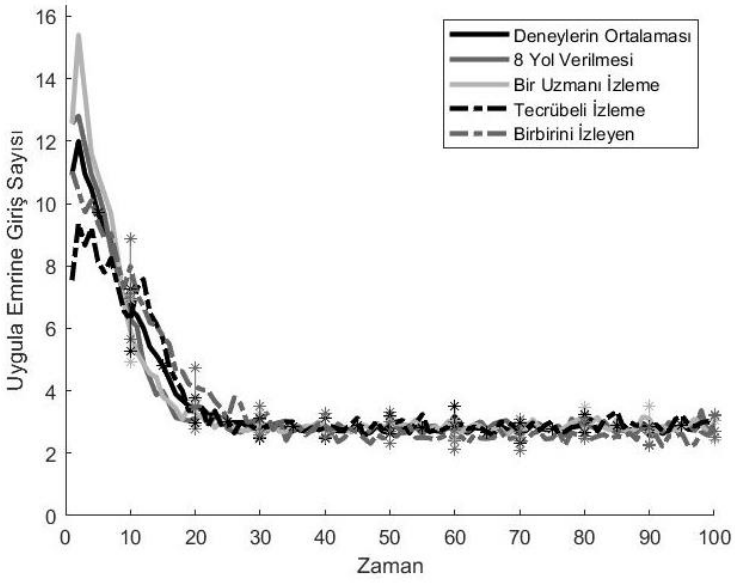
Kopyala emri kullanılan dört senaryo deneyinin deney ortalamaları ile ilişkisi Şekil 5.5` de gösterilmiştir. Kopyala emri aktif olan dört deney için alınan ortalama bilgisine en çok etkiyi birbirini gözlemleyen etmen yapmıştır. Çünkü birbirini gözlemleyerek öğrenen etmen, belirli bir zaman adımından (9000 zaman adımı) sonra bu emre girmeyi bıraktığından dolayı bu sürelerdeki verileri sıfırdır. Ortalamaya etkisi daha az olduğu için 9000 zaman adımından sonraki zaman adımları için bilgiler vermek doğru olmayacaktır. Birbirini gözlemleyen etmenin, 100 ile 600, 2900 ile 3100, 4000 ile 4700, 5000 ile 5300, 5500 ile 8300 ve 8500 ile 9200 zaman adımları arasında deneylerin ortalamasının altında kaldığı ve ortalamaya göre daha az kopyala emrine girdiği görülmüştür. Tecrübeli bir etmeni izleyen etmenin, 1300 ile 1600 zaman adımları arasında ortalamanın üstünde kaldığı ve ortalamaya göre daha çok kopyala emrine girdiği görülmüştür. 9100 ile 10000 zaman adımları arasında ise ortalamanın altında olduğu görülmüştür. Uzmanı izleyen bir etmenin, 9200 ile 10000 zaman adımları aralığında ortalamanın üstünde olduğu görülmüştür. Ancak dikkat edilmesi gereken nokta 9000 den sonraki adımlarından alınan bilgilerdir. Bu durum göz önünde bulundurulduğundan bir uzmanı izleyen etmenin neredeyse ortalamaya yakın kopyala emrine girdiği görülmüştür. 8 yol verilerek taklit eden bir etmenin, 100 ile 400 ve 9100 ile 10000

zaman adımları arasında deneylerin ortalamasının üstünde kaldığı ve ortalamaya göre daha fazla kopyala emrine girdiği görülmüştür. 1100 ile 1500 zaman adımları aralığında ise ortalamanın altında olduğu görülmüştür.




**Şekil 5.5.** Kopyala emri kullanılan dört senaryo deneyinin deney ortalamaları ile ilişkisi

Uygula emri kullanılan dört senaryo deneyinin deney ortalamaları ile ilişkisi Şekil 5.6` da gösterilmiştir. Birbirini gözlemleyen etmenin, 8300 ile 8500 ve 9500 ile 9800 zaman adımları arasında deneylerin ortalamasının altında kaldığı ve ortalamaya göre daha az uygula emrine girdiği görülmüştür. Tecrübeli bir etmeni izleyen etmenin, 1200 ile 1400 zaman adımları arasında ortalamanın üstünde kaldığı ve ortalamaya göre daha çok uygula emrine girdiği görülmüştür. 100 ile 300 zaman adımları arasında ise ortalamanın altında olduğu görülmüştür. Uzmanı izleyen bir etmenin, 100 ile 300 ve 1500 ile 1800 zaman adımları aralığında ortalamanın altında olduğu ve 1100 ile 1300 zaman adımları aralığından ise ortalamanın üstünde olduğu görülmüştür. 8 yol verilerek taklit eden bir etmenin, 1200 ile 2000 zaman adımları arasında deneylerin ortalamasının altında kaldığı ve ortalamaya göre daha az uygula emrine girdiği görülmüştür.




**Şekil 5.6.** Uygula emri kullanılan dört senaryo deneyinin deney ortalamaları ile ilişkisi

# 6. SONUÇ VE ÖNERİLER

Bu tez, sosyal öğrenme destekli, modelden bağımsız, bir grup içerisinde öğrenen etmenlerin öğrenme hız ve performanslarını artıran, özgün bir Derin Pekiştirmeli Öğrenme algoritmasını sunmaktadır. Sunulan özgün yöntem ise doğada kopyalamanın kullanımından esinlenerek içsel geri bildirim ile kopyalanan davranışların olumlu/olumsuz etkilerini tespit etmeyi amaçlamaktadır. Bu sayede bir topluluk içerisinde öğrenen etmenlerin, bir uzmanın varlığına yokluğuna bakmaksızın farklı etmenlerden kopyalama yoluyla öğrenmeyi mümkün kılmıştır. Öğrenen etmenler iki katmanlı bir kontrol sistemi tarafından yönetilmiştir. Literatürde çok katmanlı bir yapı olmadığı gibi farklı hafıza türlerinin incelenmesi söz konusu değildir. Tez kapsamında yapılan çalışmalar ile SÖ katmanındaki farklı hafıza tipleri ve bunların DPÖ algoritmasının performansına etkileri ilk kez incelenmiştir. Kontrol sistemi tarafından, kopyalama davranışının seçilebilmesi mümkün kılınmıştır. Böylece aksiyon olarak kontrol edilmesi etmenlerin öğrenmesine olumsuz bir etkisi olmadığı, aksine olumlu etkiler yaratarak öğrenmeyi desteklediği gösterilmiştir. Geliştirilmiş olan kontrol sisteminde kopyalama davranışının, kopyalamanın doğada kullanımına uygun şekilde eğitilmekte olan derin sinirsel ağın gerçekleştirebileceği aksiyonlar arasında tanımlanması, literatürde bulunan kopyalama ile DRL kullanımına yönelik diğer araştırmaların içerdiği kopyalama verisi toplama yöntemleri ile farklılık gösteren ve kopyalama davranışının dinamik olarak akıllı etmen tarafından kontrol edildiği özgün bir yaklaşımdır. Taklit ile güçlendirilmiş öğrenmeyi kullanan diğer araştırmalarla karşılaştırıldığında, bu araştırmalardan farklı olarak yöntemimiz kopyalama kaynaklı maliyetlerin bir etmen grubunun kümülatif öğrenme performansına etkileri belirlenmiştir. Taklit yoluyla elde edilen bilgilerin DQN algoritması içinde kullanılabileceği gösterilmiştir. Tez kapsamında kullanılan olasılık hesabı ile hafıza yapısında bulunan yolların faydalı olup olmadığını belirlemek mümkündür. Algoritma, belirlenmiş bir simülasyonda test edilmiştir ve SÖ destekli DQN algoritmasının, taklit olmadan aynı DQN algoritmasına kıyasla daha hızlı öğrendiği gösterilmiştir. Etmenler arasında transfer edilen tek bilgi, taklit edilen etmenin gerçekleştirdiği aksiyonlardır. Bu aksiyonların hangi durumda veya bağlamda anlamlı olduğunu belirleme süreci, taklit eden etmenin öğrenme süreci tarafından gerçekleştirilir. Bir uzmanı veya tecrübeli etmeni taklit eden etmen, taklit

edilen etmenin davranış gösterimlerini kullanarak bireysel öğrenme hızlarını arttırabildiği görülmüştür.

## 6.1 Öneriler ve Gelecek Çalışmalar

1. Benzetim ortamında geliştirilen ve test edilen yöntemin, gerçek otonom robotlar üzerinde sınanması/geliştirilmesini içerecek yeni araştırma projelerinde kullanılabilir.
2. Bu tez çalışmasında kullanılan kopyalama türü, hafıza tipi, vaka uzunluğu, hafıza boyutları ve simülasyon deney süreleri değiştirilebilir ve farklı benzetim ortamlarında denenebilir.
3. Entropi hesabı için kullanılan normalizasyon fonksiyonu yerine farklı fonksiyon kullanılabilir.

Daha büyük etmen toplulukları ile çalışılabilir fakat üzerinde durulması gereken önemli bir konu hangi etmenin taklit edileceği sorununu çözme ihtiyacı ortaya çıkmaktadır. Taklit edilecek etmen rastgele belirlenebilir. Ancak doğadaki etmenler, hangi bireyi taklit edeceklerine karar vermek için çevresel ipuçlarını kullanabilirler. Öğrenen etmenler tarafından algılanabilen böyle çevresel ipuçlarının eklenmesi, etmenlerin öğrenme hızını daha da artırabilir.

# 7. KAYNAKLAR

Bu tez çalışmasında APA atıf sistemi kullanılmıştır.


Adam, S., Busoniu, L., & Babuska, R. (2012). Experience Replay for Real-Time Reinforcement Learning Control. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*(2), 201–212. https://doi.org/10.1109/TSMCC.2011.2106494

Akins, C. K., & Zentall, T. R. (1996). Imitative learning in male Japanese quail (Coturnix japonica) using the two-action method. *Journal of Comparative Psychology*, *110*(3), 316–320. https://doi.org/10.1037/0735-7036.110.3.316

Ali, R., Qadri, Y. A., Bin Zikria, Y., Umer, T., Kim, B.-S., & Kim, S. W. (2019). Q-learning-enabled channel access in next-generation dense wireless networks for IoT-based eHealth systems. *EURASIP Journal on Wireless Communications and Networking*, *2019*(1), 178. https://doi.org/10.1186/s13638-019-1498-x

Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Processing Magazine*, *34*(6), 26–38. https://doi.org/10.1109/MSP.2017.2743240

Arwa, E. O., & Folly, K. A. (2020). Reinforcement Learning Techniques for Optimal Power Control in Grid-Connected Microgrids: A Comprehensive Review. *IEEE Access*, *8*, 208992–209007. https://doi.org/10.1109/ACCESS.2020.3038735

Babaeizadeh, M., Frosio, I., Tyree, S., Clemons, J., & Kautz, J. (2016). *Reinforcement Learning through Asynchronous Advantage Actor-Critic on a GPU*. http://arxiv.org/abs/1611.06256

Bandura, A. (1962). *Social learning through imitation* (In M. R. Jones (Ed.) (ed.)). Nebraska Symposium on Motivation, Univer. Nebraska Press.

Bandura, A. (1977). *Social learning theory: The process of modeling.* (R. L. Harris (ed.)). Readings in the theory of action.Wiley.

Bellman, R. (1957). Dynamic programming. In *Princeton University Press, Princeton*.

Blackmore, S. (1999). *The Meme Machine*. Oxford University Press.

Brown, N., Bakhtin, A., Lerer, A., & Gong, Q. (2020). Combining deep reinforcement learning and search for imperfect-information games. *Advances in Neural Information Processing Systems*, *33*, 17057–17069.

Campbell, F. M., Heyes, C. M., & Goldsmith, A. R. (1999). Stimulus learning and response learning by observation in the European starling, in a two-object/two-action test. *Animal Behaviour*, *58*(1), 151–158. https://doi.org/10.1006/anbe.1999.1121

Campo, A., Gutiérrez, Á., Nouyan, S., Pinciroli, C., Longchamp, V., Garnier, S., & Dorigo, M. (2010). Artificial pheromone for path selection by a foraging swarm of robots. *Biological Cybernetics*, *103*(5), 339–352. https://doi.org/10.1007/s00422-010-0402-x

Charpentier, A., Elie, R., & Remlinger, C. (2020). Reinforcement Learning in Economics and Finance. *Computational Economics*, 1–38. http://arxiv.org/abs/2003.10014

Chen, Q., Heydari, B., & Moghaddam, M. (2021). Leveraging Task Modularity in Reinforcement Learning for Adaptable Industry 4.0 Automation. *Journal of Mechanical Design*, *143*(7). https://doi.org/10.1115/1.4049531

Custance, D., Whiten, A., & Fredman, T. (1999). Social learning of an artificial fruit task in capuchin monkeys (Cebus apella). *Journal of Comparative Psychology*, *113*(1), 13–23.

https://doi.org/10.1037/0735-7036.113.1.13

Darwin, C. (1859). *On the Origin of Species*. John Murray, London.

Dautenhahn, K., & Nehaniv, C. L. (Eds.). (2002). *Imitation in Animals and Artifacts*. The MIT Press. https://doi.org/10.7551/mitpress/3676.001.0001

Dawkins, R. (1976). *The Selfish Gene*. Oxford University Press.

Dawson, B. V., & Foss, B. M. (1965). Observational learning in budgerigars. *Animal Behaviour*, *13*(4), 470–474. https://doi.org/10.1016/0003-3472(65)90108-9

Demiris, Y. (2002). Mirror neurons, imitation and the learning of movement sequences. *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP '02.*, *1*, 111–115. https://doi.org/10.1109/ICONIP.2002.1202141

Eiben, A., Griffioen, A., & Haasdijk, E. (2007). Population-based adaptive systems: concepts, issues, and the platform NEW TIES. *... Conference on Complex Systems, ...*, 1–15. https://doi.org/10.1.1.143.8841

El Gourari, A., Raoufi, M., Skouri, M., & Ouatik, F. (2021). The Implementation of Deep Reinforcement Learning in E-Learning and Distance Learning: Remote Practical Work. *Mobile Information Systems*, *2021*, 1–11. https://doi.org/10.1155/2021/9959954

Erbaş, M. D. (2012). *Embodied Imitation and Behavioural Adaptation in Robot Collectives*. West of England.

Erbas, M. D., Winfield, A. F., & Bull, L. (2014). Embodied imitation-enhanced reinforcement learning in multi-agent systems. *Adaptive Behavior*, *22*(1), 31–50. https://doi.org/10.1177/1059712313500503

Erdődi, L., Sommervoll, Å. Å., & Zennaro, F. M. (2021). Simulating SQL injection vulnerability exploitation using Q-learning reinforcement learning agents. *Journal of Information Security and Applications*, *61*, 102903. https://doi.org/10.1016/j.jisa.2021.102903

Feldman, J. A., Fanty, M. A., & Goodard, N. H. (1988). Computing with structured neural networks. *Computer*, *21*(3), 91–103. https://doi.org/10.1109/2.34

Finn, C., Levine, S., & Abbeel, P. (2016). Guided cost learning: deep inverse optimal control via policy optimization. *ICML'16: Proceedings of the 33rd International Conference on International Conference on Machine Learning*, *48*, 49–58.

Gallese, V., Fadiga, L., Fogassi, L., & Rizzolatti, G. (1996). Action recognition in the premotor cortex. *Brain*, *119*(2), 593–609. https://doi.org/10.1093/brain/119.2.593

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Heyes, C. M. (1996). *Introduction: Identifying and defining imitation*. In C. M. Heyes & B. G. Galef (Eds.),Social learning in animals: The roots of culture (pp. 211-220). New York: Academic Press.

Heyes, C. M., & Dawson, G. R. (1990). A demonstration of observational learning in rats using a bidirectional control. *The Quarterly Journal of Experimental Psychology B: Comparative and Physiological Psychology*, *42*((B-1)), 59–71.

Heyes, C. M., Jaldow, E., & Dawson, G. R. (1994). Imitation in Rats: Conditions of Occurrence in a Bidirectional Control Procedure. *Learning and Motivation*, *25*(3), 276–287. https://doi.org/10.1006/lmot.1994.1015

Ho, J., & Ermon, S. (2016). Generative Adversarial Imitation Learning. *Advances in Neural Information Processing Systems*. http://arxiv.org/abs/1606.03476

Hull, C. (1943). *Principles of behavior: an introduction to behavior theory*. Appleton-Century.

Ibarz, J., Tan, J., Finn, C., Kalakrishnan, M., Pastor, P., & Levine, S. (2021). How to train your robot with deep reinforcement learning: lessons we have learned. *The International Journal of Robotics Research*, *40*(4–5), 698–721. https://doi.org/10.1177/0278364920987859

Jeremiah, J. J., Abbey, S. J., Booth, C. A., & Kashyap, A. (2021). Results of Application of Artificial Neural Networks in Predicting Geo-Mechanical Properties of Stabilised Clays—A Review. *Geotechnics*, *1*(1), 147–171. https://doi.org/10.3390/geotechnics1010008

Jiang, Z., Xu, D., & Liang, J. (2017). *A Deep Reinforcement Learning Framework for the Financial Portfolio Management Problem*. http://arxiv.org/abs/1706.10059

Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, *4*, 237–285. https://doi.org/10.1613/jair.301

Katz, D., & Kahn, R. (1966). *The Social Psychology of Organizations*. John Wiley and Sons, New York.

Kheradmandian, G., & Rahmati, M. (2009). Automatic abstraction in reinforcement learning using data mining techniques. *Robotics and Autonomous Systems*, *57*(11), 1119–1128. https://doi.org/10.1016/j.robot.2009.07.002

Krieger, M. J. B., & Billeter, J.-B. (2000). The call of duty: Self-organised task allocation in a population of up to twelve mobile robots. *Robotics and Autonomous Systems*, *30*(1–2), 65–84. https://doi.org/10.1016/S0921-8890(99)00065-2

Kube, C. R., & Bonabeau, E. (1998). Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, *30*(1–2), 85–101. https://doi.org/10.1016/S0921-8890(99)00066-4

Lave, J., & Wenger, E. (1991). *Situated Learning: Legitimate Peripheral Participation*. Cambridge University Press.

Lazarus, R. S. (1966). *Psychological Stress and the Coping Process*. McGraw-Hill, New York.

Levin, K. (1951). *Field theory in social science: selected theoretical papers (Edited by Dorwin Cartwright.)*. Harpers.

Locke, J. (1960). *An Essay Concerning Human Understanding*. London: Printed by Eliz. Holt.

Long, Y., & He, H. (2020). Robot path planning based on deep reinforcement learning. *2020 IEEE Conference on Telecommunications, Optics and Computer Science (TOCS)*, 151–154. https://doi.org/10.1109/TOCS50858.2020.9339752

Luketina, J., Nardelli, N., Farquhar, G., Foerster, J., Andreas, J., Grefenstette, E., Whiteson, S., & Rocktäschel, T. (2019). *A Survey of Reinforcement Learning Informed by Natural Language*. http://arxiv.org/abs/1906.03926

Mahmud, M., Kaiser, M. S., Hussain, A., & Vassanelli, S. (2018). Applications of Deep Learning and Reinforcement Learning to Biological Data. *IEEE Transactions on Neural Networks and Learning Systems*, *29*(6), 2063–2079. https://doi.org/10.1109/TNNLS.2018.2790388

Mao, Y., Gao, F., Zhang, Q., & Yang, Z. (2022). An AUV Target-Tracking Method Combining Imitation Learning and Deep Reinforcement Learning. *Journal of Marine Science and Engineering*, *10*(3), 383. https://doi.org/10.3390/jmse10030383

Mataric, M. J. (1994). *Interaction and Intelligent Behavior*. https://doi.org/10.21236/ADA290049

Mataric, M. J., Nilsson, M., & Simsarin, K. T. (1995). Cooperative multi-robot box-pushing. *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, *3*, 556–561. https://doi.org/10.1109/IROS.1995.525940

Meltzoff, A. N., & Moore, M. K. (1977). Imitation of Facial and Manual Gestures by Human Neonates. *Science*, *198*(4312), 75–78. https://doi.org/10.1126/science.198.4312.75

Mirhoseini, A., Goldie, A., Yazgan, M., Jiang, J., Songhori, E., Wang, S., Lee, Y.-J., Johnson, E., Pathak, O., Bae, S., Nazi, A., Pak, J., Tong, A., Srinivasa, K., Hang, W., Tuncer, E., Babu, A., Le, Q. V., Laudon, J., … Dean, J. (2020). *Chip Placement with Deep Reinforcement Learning*. http://arxiv.org/abs/2004.10746

Mitchell, R. W. (1987). A Comparative-Developmental Approach to Understanding Imitation. In P. Bateson & P. Klopfer (Eds.), *Perspectives in Ethology* (pp. 183–215). Springer US. https://doi.org/10.1007/978-1-4613-1815-6_7

Ng, A. Y., & Russell, S. J. (2000). Algorithms for Inverse Reinforcement Learning. In *ICML '00: Proceedings of the Seventeenth International Conference on Machine Learnin* (pp. 663–670).

Nguyen, T. T., & Reddi, V. J. (2021). Deep Reinforcement Learning for Cyber Security. *IEEE Transactions on Neural Networks and Learning Systems*, 1–17. https://doi.org/10.1109/TNNLS.2021.3121870

Pomerleau, D. A. (1991). Efficient Training of Artificial Neural Networks for Autonomous Navigation. *Neural Computation*, *3*(1), 88–97. https://doi.org/10.1162/neco.1991.3.1.88

Popova, M., Isayev, O., & Tropsha, A. (2018). Deep reinforcement learning for de novo drug design. *Science Advances*, *4*(7). https://doi.org/10.1126/sciadv.aap7885

Ratliff, N. D., Silver, D., & Bagnell, J. A. (2009). Learning to search: Functional gradient techniques for imitation learning. *Autonomous Robots*, *27*(1), 25–53. https://doi.org/10.1007/s10514-009-9121-3

Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, *21*(4), 25–34. https://doi.org/10.1145/37402.37406

Richard S. Sutton, & Andrew G. Barto. (2018). Reinforcement Learning: An Introduction (2nd edition). In *Syria Studies* (Vol. 7, Issue 1).

Ross, S., Gordon, G. J., & Bagnell, J. A. (2011). A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. *In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics.JMLR Workshop and Conference Proceedings.*, 627–635. http://arxiv.org/abs/1011.0686

Rotter, J. B. (1947). *Social learning and clinical psychology.* Prentice-Hall, Inc. https://doi.org/10.1037/10788-000

Rummery, G. A., & Niranjan, M. (1994). On-line q-learning using connectionist systems cued/f-infeng/tr 166. *Update*, *September*.

Şahin, E. (2005). *Swarm Robotics: From Sources of Inspiration to Domains of Application* (pp. 10–20). https://doi.org/10.1007/978-3-540-30552-1_2

Schulman, J., Levine, S., Moritz, P., Jordan, M. I., & Abbeel, P. (2015). Trust Region Policy Optimization. *In International Conference on Machine Learning*, 1889–1897. http://arxiv.org/abs/1502.05477

Stirling, T., & Floreano, D. (2010). *Energy Efficient Swarm Deployment for Search in Unknown Environments* (pp. 562–563). https://doi.org/10.1007/978-3-642-15461-4_61

Suay, H. B., & Chernova, S. (2011). Effect of human guidance and state space size on Interactive Reinforcement Learning. *2011 RO-MAN*, 1–6. https://doi.org/10.1109/ROMAN.2011.6005223

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Thorndike, E. L. (1898). Animal intelligence: An experimental study of the associative processes in animals. *The Psychological Review: Monograph Supplements*, *2*(4), i–109. https://doi.org/10.1037/h0092987

Tomasello, M., Kruger, A. C., & Ratner, H. H. (1993). Cultural learning. *Behavioral and Brain Sciences*, *16*(3), 495–511. https://doi.org/10.1017/S0140525X0003123X

Trianni, V., Groß, R., Labella, T. H., Şahin, E., & Dorigo, M. (2003). *Evolving Aggregation Behaviors in a Swarm of Robots* (pp. 865–874). https://doi.org/10.1007/978-3-540-39432-7_93

Vicsek, T., Czirók, A., Ben-Jacob, E., Cohen, I., & Shochet, O. (1995). Novel Type of Phase Transition in a System of Self-Driven Particles. *Physical Review Letters*, *75*(6), 1226–1229. https://doi.org/10.1103/PhysRevLett.75.1226

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.

Wang, S., Jia, D., & Weng, X. (2018). *Deep Reinforcement Learning for Autonomous Driving*. http://arxiv.org/abs/1811.11329

Wang, W. Y., Li, J., & He, X. (2018). Deep Reinforcement Learning for NLP. *Proceedings of ACL 2018, Tutorial Abstracts*, 19–21. https://doi.org/10.18653/v1/P18-5007

Watkins, C. J. C. H. (1989). *Learning from Delayed Rewards*. Cambridge University.

Yi, M., Xu, X., Zeng, Y., & Jung, S. (2018). Deep imitation reinforcement learning with expert demonstration data. *The Journal of Engineering*, *2018*(16), 1567–1573. https://doi.org/10.1049/joe.2018.8314

Yu, C., Liu, J., Nemati, S., & Yin, G. (2021). Reinforcement Learning in Healthcare: A Survey. *ACM Computing Surveys*, *55*(1), 1–36. https://doi.org/10.1145/3477600

Zeng, J., Hu, J., & Zhang, Y. (2018). Adaptive Traffic Signal Control with Deep Recurrent Q-learning. *2018 IEEE Intelligent Vehicles Symposium (IV)*, 1215–1220. https://doi.org/10.1109/IVS.2018.8500414

Zentall, T. R. (1996). An Analysis of Imitative Learning in Animals. In *Social Learning in Animals* (pp. 221–243). Elsevier. https://doi.org/10.1016/B978-012273965-1/50012-1

Zentall, T. R. (2001). Imitation in Animals: Evidence, Function, and Mechanisms. *Cybernetics and Systems*, *32*(1–2), 53–96. https://doi.org/10.1080/019697201300001812

Zentall, T. R., Sutton, J. E., & Sherburne, L. M. (1996). True Imitative Learning in Pigeons. *Psychological Science*, *7*(6), 343–346. https://doi.org/10.1111/j.1467-9280.1996.tb00386.x

Zhao, D., Chen, Y., & Lv, L. (2017). Deep Reinforcement Learning With Visual Attention for Vehicle Classification. *IEEE Transactions on Cognitive and Developmental Systems*, *9*(4), 356–367. https://doi.org/10.1109/TCDS.2016.2614675

Ziebart, B. D., Maas, A. L., Bagnell, J. A., & Dey, A. K. (2008). Maximum entropy inverse reinforcement learning. In I. D. Fox & C. P. Gomes (Eds.), *AAAI* (Vol. 8).