İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

ROBOTLARIN BİLİNMEYEN CİSİMLERİN TUTULABİLİRLİĞİNİ İÇSEL MOTİVASYON DESTEĞİ İLE ÖĞRENMESİ

YÜKSEK LİSANS TEZİ

Erçin TEMEL

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

OCAK 2015

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

ROBOTLARIN BİLİNMEYEN CİSİMLERİN TUTULABİLİRLİĞİNİ İÇSEL MOTİVASYON DESTEĞİ İLE ÖĞRENMESİ

YÜKSEK LİSANS TEZİ

Erçin TEMEL
(504111514)

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Yrd. Doç. Dr. Sanem SARIEL

OCAK 2015

İTÜ, Fen Bilimleri Enstitüsü'nün 504111514 numaralı Yüksek Lisans Öğrencisi **Erçin TEMEL**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı **"ROBOTLARIN BİLİNMEYEN CİSİMLERİN TUTULABİLİRLİĞİNİ İÇSEL MOTİVASYON DESTEĞİ İLE ÖĞRENMESİ"** başlıklı tezini aşağıdaki imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı :	**Yrd. Doç. Dr. Sanem SARIEL** İstanbul Teknik Üniversitesi	.....
Jüri Üyeleri :	**Yrd. Doç. Dr. Ömer Sinan SARAÇ** İstanbul Teknik Üniversitesi	.....
	Prof. Dr. H. Levent AKIN Boğaziçi Üniversitesi	.....
		.....

Teslim Tarihi : **15 Aralık 2014**
Savunma Tarihi : **21 Ocak 2015**

ÖNSÖZ

Tez çalışması süresince gerçekleştirilen deneylerdeki önemli katkıları sebebiyle ITU Artificial Intelligence and Robotics (AIR) laboratuarında görevli tüm öğrencilere,

Tez çalışmamdaki çok önemli katkılarda ve yardımlarda bulunan tez danışmanım Sayın Sanem Sarıel'e teşekkürlerimi sunarım.

Tez çalışması süresince yorumları ve yönlendirmeleri için Sayın Beata Joanna Gryzb'e de teşekkürlerimi sunarım.

Bu tez çalışması TUBITAK tarafından desteklenen 111E-286 numaralı proje kapsamında gerçekleştirilmiştir.

Ocak 2015

Erçin TEMEL
(Bilgisayar Mühendisi)

İÇİNDEKİLER

Sayfa

KISALTMALAR

RL : Reinforcement Learning
RVIZ : ROS-Visualization
PCL : Point Cloud Library
V-REP : Virtual Robot Experimentation Platform
RANSAC : Random Sample Consensus
ROS : Robot Operating System

ÇİZELGE LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

ROBOTLARIN BİLİNMEYEN CİSİMLERİN TUTULABİLİRLİĞİNİ İÇSEL MOTİVASYON DESTEĞİ İLE ÖĞRENMESİ

ÖZET

İnsanlar ile etkileşim içerisinde olan bilişsel robotun tanımadığı bir nesneyle etkileşime geçebilmek için etkili tutma yöntemleri ile donatılmış olması gerekmektedir. Özellikle düzensiz olan ortamlarda, sensör ve kontrol mekanizmalarının yetersizliği, önceden edinilmiş bilgi yetersizliği ve nesnelerin farklı modelleri sebebi ile zorluklar katlanarak artmaktadır. Bu tür zorlukların aşılabilmesi için robotun verimli öğrenme metotlarına sahip olması gerekmektedir. Bu tez çalışmasında, tutma eylemi için belirlenen potansiyel başarılı yürütme deneyimlerinin öğrenilmesi, içsel motivasyon ve ivecenlik ile verimli hale getirilmiştir. Robotun tutma eyleminin başarılı veya başarısız olduğuna etkin şekilde karar verebilmesi için içsel motivasyon ve ivecenlik sistemi önerilmiştir. Önerilen sistemde, tutma görevinin başarı kararı için görme, planlama, yürütme, gözlemleme ve öğrenme süreçleri bir araya getirilmiştir.

Sistem ilk olarak, RGB-D kamera ile ortamda bulunan, robotun etkileşmesi gereken bilinmeyen nesnenin 3 boyutlu görüntüsünü alarak niteliklerini belirler. Bu görüntü, nesnenin kenarlarının ve köşelerinin 3 boyutlu olarak tespit edilmesinde kullanılır. Kenarların tesbiti için ise PCL (Point Cloud Library) kütüphanesindeki "Organized Point Cloud Segmentation with Connected Components" algoritması kullanılır. Bu algoritma, kameradan gelen noktalar arasından birbiri ile bağlantılı olanları işaretler. İşaretli noktalar ise bir başka sistem olan potansiyel tutma noktası bulma algoritmasına girdi olarak verilir.

Tutma noktası bulma algoritması, belirlenen kenar ve köşe noktaları üzerinden nokta ikililerini belirler. Bu noktaların birbiri ile karşılıklı ve 2 boyutlu düzlemde olması temel koşuldur. Algoritma öncelikle, cismin üzerinde referans noktası belirler ve referans noktasına en yakın noktayı bulur. Bir sonraki aşaması ise referans noktası ile ona en yakın noktadan geçen doğru üzerinde başka bir nokta bulmayı dener. Eğer başarılı olursa, referans noktanın haricindeki 2 nokta tutma noktası olarak işaretlenir. Tutma noktasından üretilen potansiyel nokta çiftleri, simülasyon ortamda denenerek ön elemeden geçirilirler. Böylece gerçek dünyada robot açısından başarılı olma ihtimali olmayan noktalar elenir ve zaman kaybı en aza indirgenmiş olur. Simülasyon sonucu üretilen nokta çiftleri başarılı olması beklenen çiftlerdir.

Öğrenme sürecinde ise pekiştirmeli öğrenme metodu kullanılmıştır. Böylece robot, eylemler, eylemlerin sonuçları, cismin durumu gibi etkenler konusunda deneme yanılma ile tecrübe kazanır ve öğrenir. Deneme yanılma yöntemi ile öğrenme sürecinde, robotun belirli sabit bir sayıda deneme yapmasından ziyade, tez çalışmasında içsel motivasyon, isteksizlik ve ivecenlik durumları da göz önünde bulundurularak metodlar geliştirilmiştir. Bu metodlar sayesinde robot tutulması imkansız olan bir nesnenin hızlı bir şekilde tutulamayacağına kanaat getirirken, tutulabilecek nesneler konusunda tutma noktalarının belirlenmesinde daha hızlı karar

vermesi mümkün olur. Böylece cisimler ile etkileşen robot, daha sonra cisimlerin aynısı ya da benzer durumlarda tutma noktaları ve nasıl tutulacağına dair hızlı eylemde bulunabilir. İnsani duygular olan motivasyon, isteksizlik ve ivecenlik kavramları ise robotun başarmaya çalıştığı göreve göre hızlı pes etmesine neden olurken, bazı durumlarda uzun süreli olarak vazgeçmeden denemeye devamını sağlamaktadır. Bu sayede, robotun insan ile etkileşimi esnasında hızlı karar verip, hızlı eylem yürütmesi sağlanırken, zaman veya mekan konusunda kısıtı olmadığı durumlarda da uzun soluklu denemeler yapabilmesini sağlamaktadır. Öğrenme sonucunda elde edilen tecrübeler, robotun gelecekte alınacak olan kararlar ve eylemler konusunda doğru yönlenmesini sağlayacaktır. İçsel motivasyon metotları ile karşılaştırıldığında, göreve göre hareket edebilecek olan öğrenme robotun karar mekanizmasını hızlandırırken, insan etkileşimini de artırmaktadır. Yöntem, V-REP simülasyon ortamı ve 7 hareket eksenli robot kolu kullanılarak analiz edilmiştir. Deneylerde farklı nesneler için, robotun karar mekanizmasındaki hız ve eylem planlarının değişkenliği gözlenmiştir. Deney setinde, küp, armut, üzüm, havuç göreceli olarak kolay tutulabilir nesnelerken, labut kolay devrilebilen ama tutulabilir, top ise tutulamaz cisim olarak seçilmiştir. Deneyler incelendiğinde tutulması mümkün olmayan top için robot çabucak tutulamayacağına karar verirken, diğer nesneler için farklı denemeleri farklı sayılarda yaparak tutulabildiğini, diğer yöntemlere göre çok daha hızlı karar vermiştir.

LEARNING GRASPABILITY OF UNKNOWN OBJECTS VIA INTRINSIC MOTIVATION

SUMMARY

Robots need effective grasp procedures to interact with and manipulate unknown objects. In unstructured environments, challenges arise mainly due to uncertainties in sensing and control, and lack of prior knowledge and model of objects. Effective learning methods are essential to deal with these challenges. One classic approach here is to use reinforcement learning (RL) where an agent actively interacts with an environment and learns from the consequences of its actions, rather than from being explicitly taught. An agent selects its actions on basis of its past experiences (exploitation) and also by new choices (exploration). The goal of an agent is to maximize the global reward; therefore the agent needs to rely on actions that led to high rewards in the past. However, if the agent is too greedy and neglects exploration, it might never find the optimal strategy for the task. Hence, to find the best ways to perform an action they need to find a balance between exploitation of current knowledge and exploration to discover new knowledge that might lead to better performance in the future.

In our work, we use reinforcement learning (RL) framework for learning and incorporate competence-based intrinsic motivation for guidance in search. The complexity of reinforcement learning is high in terms of the number of state action pairs and the computations needed to determine utility values. Approximate policy iteration methods can be used to alleviate this problem based on sampling. Imitation learning before reinforcement learning is one of the methods for decreasing the complexity in RL. Furthermore, it is also used for robots learn crucial parameters in movement to accomplish the task. Frustration level of the robot is also taken into account for learning mechanism. We further extend this approach by adopting an adaptive frustration level depending on a task. Intrinsic motivation is investigated in earlier works. System of "interestingness" was proposed and curiosity concept for reinforcement learning was introduced. Intrinsic motivation was also considered as learning objective. Different from curiosity and reward functions, ideal level of frustration is beneficial for exploration and faster learning. In addition, competence-based intrinsic motivation for learning was proposed in literature. In our work, main difference is that impulsiveness is adapted into the frustration rate in order to change the learning rate dynamically based on a task in real world environment for robots.

We propose an intrinsically motivated reinforcement learning system for robots to learn graspability of unknown objects. The system includes two main phases for determination of grasp points on objects and experimentation of them in the real world. The first phase includes the required methods to determine candidate grasp point pairs in simulation. Note that a robot arm with a two-fingered end effector is selected as the target platform. For this reason, grasp points are determined as point

pairs. In the second phase of the system, the grasp points determined in the first phase are experimented in the real world through reinforcement learning. The following subsections explain the details of these processes.

The first step in the framework is detecting objects in the scene by using an ASUS Xtion Pro Live RGB-D camera mounted on a linear platform for interpreting the scene for tabletop manipulation scenarios by a robotic arm. For object detection, Organized Point Cloud Segmentation with Connected Components algorithm from PCL is used. This algorithm finds and marks connected pixels coming from the RGB-D camera. Hence, the object's center of mass and its edges are detected to be used by the grasp point detection algorithm that finds candidate grasp point pairs for a two-fingered robotic hand.

Next step is detecting candidate grasp points in the simulator. Objects are represented by their center of masses (μ) and 3D edges (H). Then candidate grasp point pairs ($\rho = [p_1, p_2]$) are determined with Grasp Point Detection Algorithm. In the algorithm, initially the reference points are determined. The center of mass, the upside and the bottom side center points are chosen as references. Based on these points, cross section points coplanar with the reference points and parallel to the table surface are determined. In the next step, the algorithm detects the closest point to the reference points on the same planar and draws a line crossing with reference points and closest to it. The second step is determining the opposite point to the closest one on the same line. This procedure continues until all points are tested. The algorithm produces the candidate grasp pairs (two grasp points with x,y,z values) and orientation of each pair according to (0,0) point in 2D (x,y) plane. These grasp points are tested in the simulator for finding out only the feasible ones. These point pairs are tried in simulator environment in order to eliminate pairs which are impossible for grasping an object with robotic arm. This process is saving time, so robot can decide and learn faster.

In learning process, we propose a competence-based approach to reinforcement learning where exploration and exploitation is balanced while learning to grasp novel objects. In our approach, the dynamics of balancing between exploration and exploitation is tightly related to the level of frustration. The failures in obtaining a new goal may significantly increase the robot's level of frustration, and push it into searching new solutions in order to achieve its goal. However, a prolonged state of frustration, when no solution can been found, will lead to a state of learned helplessness, and the goal will be marked as unachievable at the current state (i.e., object not graspable). Simply speaking, an optimal level of frustration favours more explorative behaviour, whereas low or high level of frustration favours more exploitative behaviour. Additionally, we dynamically change the robot's impulsiveness that influences how fast the robot gets frustrated, and indirectly how much time it devotes to learning a particular task.

To demonstrate the advantages of our approach, we compare it with three other action selection methods: ε-greedy algorithm, Softmax function with constant temperature parameter, Softmax function with variable temperature depending on agent's overall frustration level. The results show that the robot equipped with frustration and impulsiveness learns faster than the robot with standard action selection strategies providing some evidence that the use of artificial emotions can improve the learning time. For example, when a robot plays a quick game with a human, it has to learn quickly. However, when the robot is alone, it can spend relatively more time on

exploring different states. By changing the impulsiveness, the robot may dynamically control its level of frustration and therefore the time devoted for learning a particular task. Hence, the robot could behave differently in different environments and for different tasks.

1. GİRİŞ

Planlama, makine öğrenmesi, yapay görme alanlarındaki gelişmeler ile sadece mekanik olarak robotlar tarafından yapılamaz olarak görülen birçok görev insanlardan daha başarılı bir şekilde robotlar tarafından yapılmaya başlandı. Durum bu şekilde olunca, robotların insani özellikler kazanmasının ihtiyacı doğmuştur. Özellikle, robotların standart mekanik hareketlerin dışına çıkması ile birlikte algı ve nesne etkileşimi kavramı robotlar için önemli kavramlar haline gelmiştir. Dolayısıyla robotların insanlarla etkileşime girebilmesi adına, robotların daha insani davranması üzerine çalışmalar devam etmektedir. Aynı zamanda robotların mekanik özellikleri sayesinde, insanların yapamayacağı ağır veya zor işlerin de gerçeklenmesi sağlanabilmektedir.

Bilişsel robotların, plan yapma, ortam ile etkileşime girme, yorum yapma ve öğrenme ile birlikte amaçlarını ve hedeflerini gerçekleştirmesi insan ile etkileşime girebilmesi adına en önemli özelliklerdir. Literatürde bulunan öğrenme yöntemleri incelendiğinde, robotlara cisimler veya insanlar ile etkileşime girebilmesi için bir tanım kümesi gerekmektedir. Bu tanım kümesi içerisinde başlangıç ve hedef durumları, sabit veya statik olarak ifade edilir ve robot bu tanım kümesi kapsamında çeşitli eylemler yürütür. Yürütülen eylem esnasında da ortamdan gelebilecek her türlü geri besleme robot için bir öğrenme algısı yaratacaktır. Böylece öğrenme algoritmaları ile görevin veya hedefin niteliğine göre robotun nasıl hareket edebileceği öğretilebilmektedir.

Bu öğrenme aşamasında, robotun kaç kere deneyeceği ya da ne kadar süre denemesi gerektiğine dair geliştirilmiş bir mantık veya formül bulunmamaktadır. Öğrenme süresi için genelde sabit bir sayı/süre verilirken, bazı durumlarda ise eylem uzayının tüm elemanlarının tüm kombinasyonlarının denenmesi sağlanmaktadır. Nihayetinde, gereksiz yorucu eylemlerin birden fazla uygulanması öğrenme hızında performans sorunları ortaya çıkarırken aynı zamanda öğrenme süreci çok uzun zaman almaktadır.

Bu tez çalışmasında çözülmeye çalışılan problem, robotların deneme yanılma ile öğrenirken, ne kadar süre veya sayıda deneyeceğinin belirlenmesi, ne zaman vazgeçmesi gerektiğinin hesaplanması veya robotun ne zaman öğrendiğine kanaat

getirmesi gerektiğinin çıkarımıdır. Problemin çözümü için kullanılan yaklaşım, insanlardaki içsel motivasyonun robotlara uygulanmasına dayanmaktadır. İsteksizlik, motivasyon ve ivecenlik duygularının insanlar üzerindeki modellerinin robotlara uyarlanması sağlanmıştır.

1.1 Hipotez

Hipotez, pekiştirmeli öğrenmeyle öğrenen robotun gereksiz yorucu eylemleri tekrar tekrar uygulamasını, yeterlilik temelli içsel motivasyon etkenlerini öğrenme sistemine entegre ederek engellemek ve başarısız olacağı durum-eylem ikililerinden hızlıca pes etmesini sağlarken, başarılı olacağı durum-eylem ikililerini farklı ortam koşullarına göre daha az veya daha çok deneyerek öğrenme hızını değiştirmektir.

1.2 Tezin Amacı

Bilişsel robotların öğrenme süreçlerinde kullanılan metotların en önemlilerinden biri robotun ortamdan geri besleme ile öğrendiği deneme yanılma adımlarını içeren pekiştirmeli öğrenme yöntemidir. Bu yöntemdeki temel sorun, robotun bir eylemi kaç kere deneyeceğinin genel kabul görmüş bir modelinin olmamasıdır. Bunun sonucunda, robotun öğrendiğinin ya da öğrenemediğinin belirlenmesi çok zaman almaktadır. Bu problemin çözümü için bu tez çalışmasında insanların öğrenme esnasındaki modellenmiş davranışları, robotlara uyarlanarak, onlara daha insani özellikler katabilmek, motivasyon duygusunun ve etkenlerinin fiziksel modellerinin pekiştirmeli öğrenme algoritmalarına entegre edebilmek hedeflenmiştir.

Tezin içerisinde Q öğrenmesi [1] pekiştirmeli öğrenme algoritması kullanılmıştır. Q öğrenmesi algoritması, programın eylem uzayından bir eylem seçmesi ile başlar ve bu eylem, hedefe ulaşılabilmek için denenir. Deneme ardından başarılı olup olmadığının kontrolü sağlanarak, başarı kriteri belirlenir. Başarısız olduğu sürece eğer bir kriter koyulmadıysa algoritma sonsuza kadar eylem uzayındaki eylemleri deneyecektir. Algoritmayı sonlandırıcı yöntemler geliştirilmiş olsa da insanın böyle bir deneme yanılma sürecinden vazgeçme durumları incelenip algoritmaya entegre edilmemiştir. Bu çalışmada robotun pes etmesi gereken zamanı ya da öğrendiğine kanaat getirip denemeleri bırakmasını belirleyecek yöntemler üzerine çalışılmıştır.

II. Bölümde robotlarda destekli öğrenme ve eylem seçim yöntemleri üzerine genel bilgiler verilmektedir. III. Bölümde içsel motivasyona psikologların bakış açısı açıklanarak, bilgisayar sistemlerinde uygulanma yöntemleri açıklanmaktadır. IV. Bölümde robotlar tarafından bilinmeyen nesnelerin nasıl tutulabileceği açıklanırken, içsel motivasyona dayalı yöntemler detaylı anlatılmaktadır. V. Bölümde deney ortamı bileşenleri ve deneylerin sonuçları belirtilmektedir. VI. Bölüm ise sonuç ve tartışma bölümüdür.

2. ROBOTLARDA PEKİŞTİRMELİ ÖĞRENME

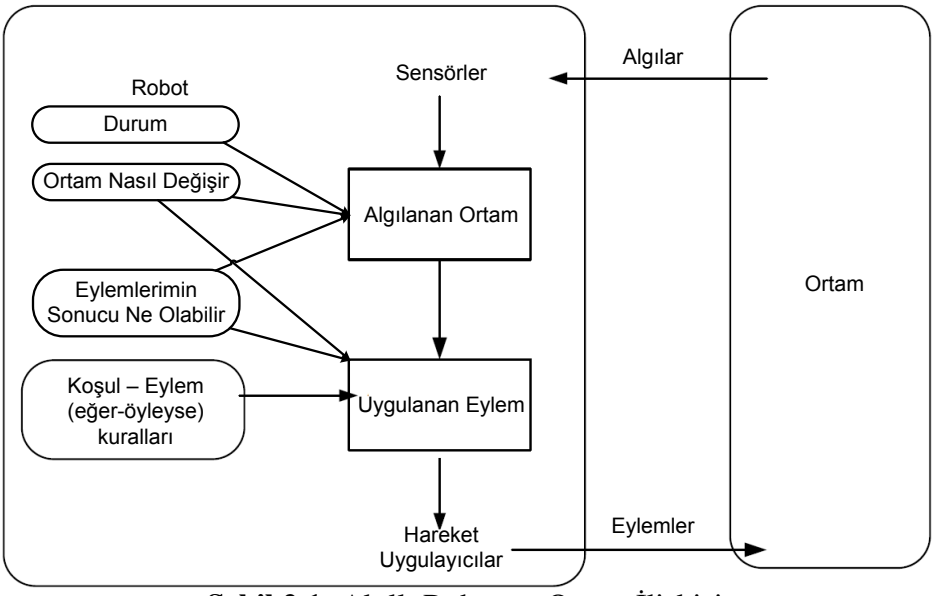
Pekiştirmeli öğrenmede (reinforcement learning), öğrenen robot/etmen sistemi kararlarını ve eylemlerini uygularken her zaman ortam ile etkileşim içerisindedir. Öğrenebilen bir robotun, içerisinde bulunduğu ortamdaki etkileri analiz ederek, hedeflediği noktaya gelmek için en iyi eylemi seçmeyi hedeflemesi ortamı analiz etmesi gerektiğine bir örnektir. Hedefe ulaşmak için ise robot bir seri eylem uygular ve her uyguladığı eylem sonrasında ortamdan ya ödül ya da ceza şeklinde bir geri dönüş alır. Bu eylem serilerine ise deneme-yanılma denir ki bu deneme-yanılmalar sonrasında robot en iyi olabilecek eylemler serisini ilgili problem çözümü için bulmuş veya seçmiş olur.

Pekiştirmeli öğrenmede, robotun içerisinde bulunduğu ortamın tanımı yapılmalıdır. Robotun bir problemi çözmeye çalışırken veya hedefe ulaşmaya çalışırken, uyguladığı her eylem sonucunda aldığı ödül/ceza performans ölçümlemek için kullanılır. Robotun çevresi ile etkileşimde olduğu, ödülün değişmesine neden olan her şey ise ortam olarak tanımlanabilir [2].

Öğrenebilen bir robot her zaman yeni olasılıkları denemek yerine daha önceden öğrendiği bilgilerini de kullanabilir. Bu bilgi ve tecrübeler çok basit olabileceği gibi çok karmaşık da olabilir. Akıllı bir robotun tecrübelerinden mi yoksa yeni olasılıkları deneyerek mi hedefe gideceğini belirten ve bunları hangi sıralama ile yapacağını belirten ise hedef için belirlenecek plandır. Robot ile ortam arasında standart olabilecek ilişkiler bütünü Şekil 2.1'de gösterilmiştir [3].

Öğrenebilen rasyonel bir robot, sürekli değişim içinde olan ortama uyum sağlayıp karmaşık problemleri çözebilmelidir. Rasyonel öğrenebilen bir robotun sahip olması gereken özellikler:

1. Periyodik ya da sürekli olmak üzere ortamdan verileri alabilmeli
2. Otonom hareket edebilmeli ve karar alabilmeli
3. Bilgileri birbiri ile birleştirerek geniş kapsamlı bilgi kümesi yaratabilmeli



Şekil 2.1: Akıllı Robot ve Ortam İlişkisi.

4. Sürekli öğrenme kabiliyetine sahip olmalı
5. Öğrendiklerini tecrübelere çevirebilmeli ve tecrübelerinden tekrar tekrar öğrenebilmeli

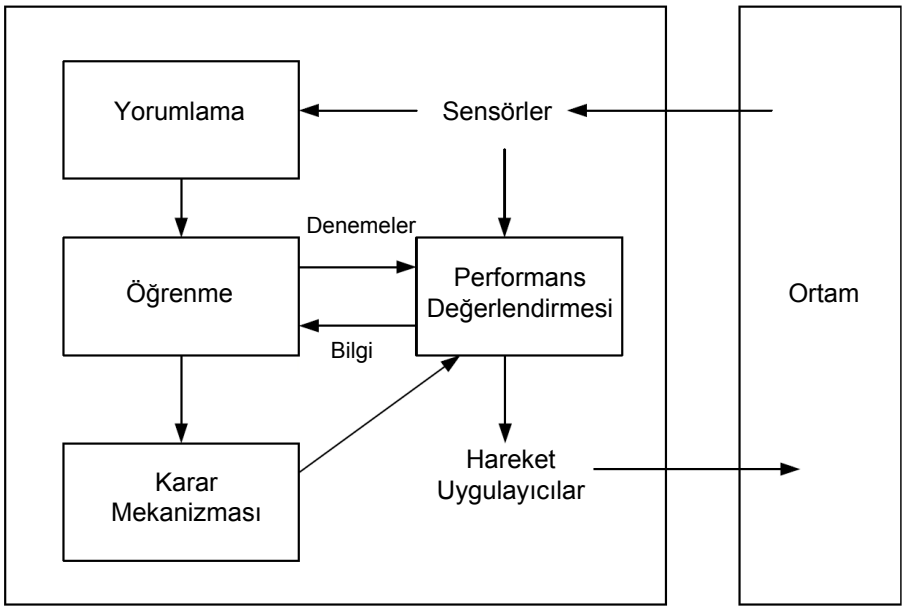
Akıllı robotların içerisinde bulunduğu gerçek dünyadaki ortamların sürekli değişmesi ve sensörlerin ortamdan aldığı verilerin genellikle eksik(partial information) ya da gürültülü (noisy) olabilmesi karmaşıklığı artırmaktadır. Bu durumlarda esnek olup, tecrübelerinden de faydalanarak gerekirse denenmemiş olası koşulları da deneyerek, anlamlı çıkarımlar ile eylem seçebilmeli ve kısmi keşfedilmiş ortamlarda hareket edebilmelidir. Uyguladığı eylemden elde ettiği ödül veya cezayı da tecrübelerine katabilmelidir. Robotun ortama göre esnek olabilmesi için sahip olması gereken özellikler:

1. Olaylara yanıt verebilen(responsive) olmalı. Ortamdan gelen bilgiler ışığında zamanında cevap verebilmeli.
2. Önetkin(proactive) olmalı. Çevresel etkenler karşısında fırsatları yakalayabilmeli, hedef odaklı hareket edip, uygun eylemi gerçekleştirebilmeli.
3. Sosyal olmalı. Ortam içerisinde bulunan insan veya diğer robotlar ile etkileşime geçebilmeli.

2.1 Öğrenen Robotlar

Açıklandığı gibi, akıllı robotların, kendi başlarına bilinmeyen veya sürekli değişen ortamlarda hareket edebilme veya karar verebilme yetisine sahip olması gerekmektedir. Robotun daha önceden öğrendiği bilgiler, bulunduğu ortam için çıkarım yapmasına veya doğru sonucunu bildiği kararlar almasına yeterli olmayabilir. Dolayısıyla robotların bu ortamlarda olabildiğince mantıklı kararlar alması gereklidir, her karar sonrası da kendini geliştirecek şekilde gelişme sağlamalıdır.

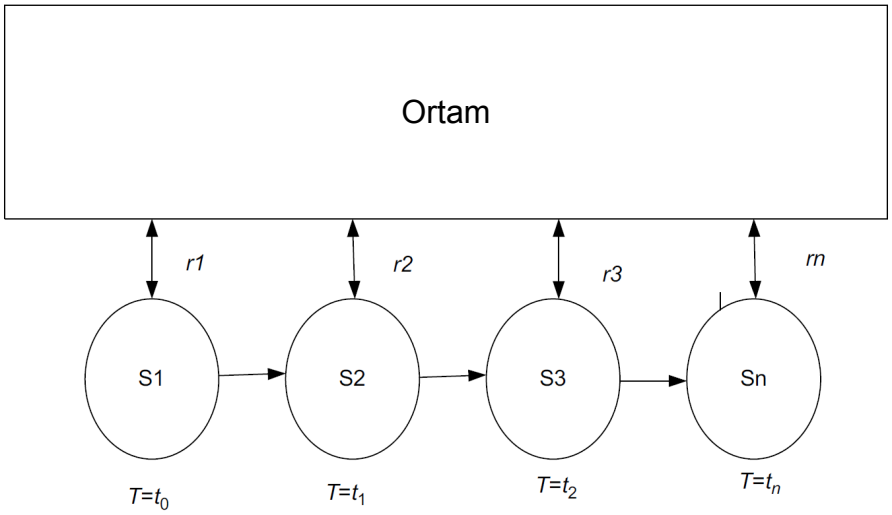
Gerçek hayattaki senaryolarda, bir öğretmen ile öğrenme, bu robotlar için gözetimli (supervised) öğrenmeye karşılık düşmektedir, her zaman mümkün olamamaktadır. Dolayısıyla robotların ortama göre tahmini modeller geliştirmesi gerekmektedir. Tahminlerinde kullanacağı veriler ise belli eylemler sonrasındaki çıktılar ne olabilir, bir rekabet ortamında karşıdaki rakip ne yapabilir gibi sorulara dayalıdır. Robotlar kararlarını verirken de bazen rastgele hareketler yapmak zorunda kalırlar. Bu eylemleri denerken, sonucuna göre tüm etkisini inceleyebilir olmalı yani ödül veya cezasını alabilmelidir (Şekil 2.2).



Şekil 2.2: Akıllı Robot ve Öğrenmesi.

Gerçek dünyada olduğu gibi robotların ortamlarında da, ödüller her zaman oyun veya görev sonrasında gelmektedir. Bazı durumlarda periyodik olarak da hedefe ulaşmadan küçük ödüller gelebilmektedir. Bu tür ödüller insanlarda olduğu gibi robotlarda da

hedefe ulaşmak için kararının ne kadar doğru olduğunu ölçümlemede önemli bir rol üstlenmektedir; çünkü robot uygulanan planın aynen kalması mı gerektiği ya da değiştirilmesi mi gerektiğini süreç içerisinde karar vererek önetkin bir şekilde hareket edebilmektedir. Netice itibari ile pekiştirmeli öğrenmede hedefe ulaşıldığında robotun en yüksek ödülü alması için ara ödülleri süreç içerisinde kullanabilmek çok faydalı olabilmektedir. Süreç içerisinde ödül gelmiyorsa da bahsedildiği gibi robot tahmin ederek ödülü maksimum şekilde elde etmeye çalışacaktır (Şekil 2.3).



Şekil 2.3: Ödülün Ortamdan Durum-Eylem İkilisi Sonrası Nasıl Geldiğinin İlişkisi.

2.2 Geri Bildirim ve Ödül Hesaplamaları

Akıllı robotun hedefe varmaktaki en önemli amacı maksimum ödülü elde etmektir. Bu ödüller belirli zaman aralıklarında ya da tüm hedef tamamlandığında elde edilir. Ortamdan elde edilen ödül denklem (2.1) ile belirtilebilir. Bu denklemde R_T T anındaki toplam ödül ve r_t t anındaki anlık ödül olarak değerlendirilmektedir.

$$R_T = r_{t-1} + r_{t-2} + r_{t-3} + \ldots + r_T \tag{2.1}$$

Robot-ortam ilişkisini basitçe durumlara ayırırsak, her durumda gelen ödül ve eylem sonrası ulaşılan durum farklı olacaktır. Sürekli devam eden görevlerde bölümlendirmek zor olsa da kesikli görevler de basitçe durumlara ayrılabilir. Sonuç olarak ise toplam ödül süreç içerisinde alınmış olan ödüllerin toplamı olmaktadır.

2.3 Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme sisteminde ortam etkenine ek olarak, ilke (policy), değer fonksiyonu (value function), ödül fonksiyonu (reward function) belirlenmelidir.

İlke, belirli bir zaman dilimi içerisinde robotun davranışını veya eylemini tanımlamak için kullanılır. İlke, basit bir başvuru tablosu olabileceği gibi geniş bir örnek uzayını arayan karmaşık bir fonksiyon da olabilir. Genellikle ilkeler stokastik yapıdadır.

Ödül fonksiyonu ise robotun içerisinde bulunduğu durumu ve ilgili davranış seçeneklerinin seçilebilirlik seviyesini tek bir sayı ile belirtir. Bu sayı sayesinde nicelik seviyesinde, eylemlerin hangisinin faydalı veya faydasız olduğu belirlenir. Dolayısıyla anlıktır ve anlık olarak bir problemin çözümü için değer üretir.

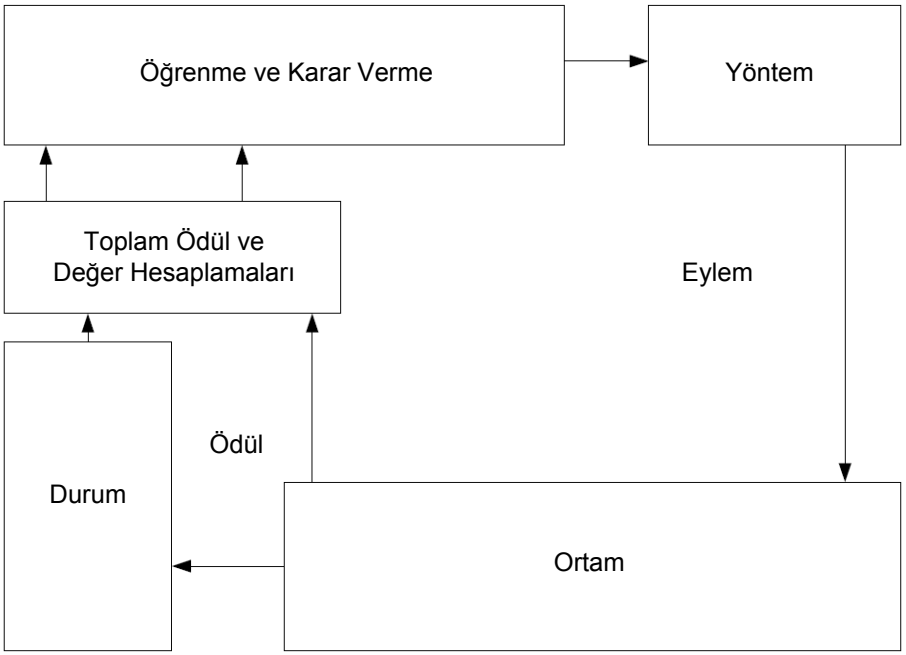
Ödül fonksiyonundan farklı olarak, değer fonksiyonu ileriye dönük olarak iyi veya kötüyü belirler. Bu fonksiyon ile robot bir eylemi seçtiğinde uzun vadede toplanabilecek ödül miktarını tahmin edebilir. Bu noktadan yola çıkarak, ödüller anlık ortam koşullarını değerlendirirken; değerler durum-eylem ikililerinin ve ilişkili ödüllerin uzun vadeli istenebilirliğini belirler. İleriye dönük olarak yapılacak seçimler göz önünde bulundurularak değer mekanizması işleyecektir. Buna rağmen, ödüller olmazsa değerler olmaz; ama değerler sayesinde maksimum ödül tahmin edilebilmektedir. Davranışlar ise daha ziyade değerlere göre seçilir. Değerleri belirlemek ise ödülleri belirlemekten daha zordur; çünkü ödüller ortamdan direk alınan bilgi ile belirlenirken, değerler çeşitli hesaplar doğrultusunda tahmin edilmelidir. Bu yüzden başarılı bir ilke ile değer tahminlemesi yapılmalıdır.

Ortam, ilke, değer ve ödül fonksiyonları göz önüne alındığında Şekil 2.4'daki pekiştirmeli öğrenme modeli ortaya çıkmaktadır.

2.4 İdeal Yöntemin Öğrenilmesi

Model aslında durumlar arasındaki geçiş hakkında matematiksel olarak bilgiye sahip olmaktır. Modele sahip olmak, durumlar arasındaki geçişi daha verimli hale getirebilmektedir. İki çeşit yöntem tipi vardır:

- Modelsiz Yöntemler: Robotların model bilgisi olmadan öğrenmesi
- Model Tabanlı Yöntemler : Robotların modele bağlı olarak yaptığı öğrenme



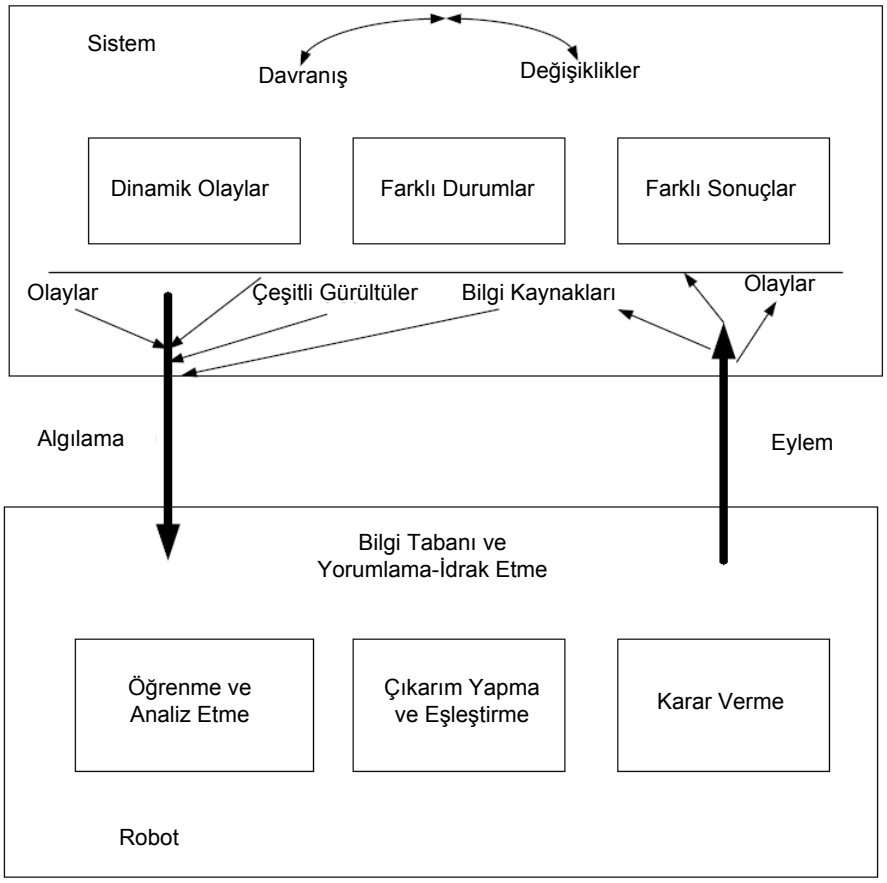
Şekil 2.4: Pekiştirmeli Öğrenme Modeli.

Pekiştirmeli öğrenmenin temel problemi olan seçilen eylemin iyi veya kötü olmasının kararı şimdiye kadar anlatılan yöntemlerde eylem sonucuna göre karar veriliyordu. Eğer ödül alındıysa iyi, ceza alındıysa kötü olarak algılanıyordu. Oysaki Sutton'un önerdiği [4] zamansal fark (temporal-difference) metoduna göre herhangi bir t anında bir eylemin sonucu olarak ödül/ceza elde edildiğinde, daha önceden gerçekleştirilmiş ama ödül/ceza alınamamış eylemlerin de ödül/ceza değerlemesi güncellenmektedir. Bunun sonucu olarak ise anlık ve bir sonraki durumun ödülünü tahmin etmek mümkün olmaktadır. Bu tür metodu gerçeklemek için ise uyarlanabilir dinamik programlama kavramı konusunda bilgi sahibi olunmalıdır.

2.5 Uyarlanabilir Dinamik Programlama

Dinamik programlamanın odaklandığı nokta, hesaplama konusunda verimliliktir. Bu tür programlama, karmaşık hesaplama problemlerini daha basit küçük problemlere böler ve hesaplar. Elde edilen sonuçlar bir diziye saklanır ve ilgili problemin sonucu gerektiğinde dizideki hazır sonucu kullanır. Uyarlanabilir dinamik programlama ise dinamik programlama ile pekiştirmeli öğrenmeyi kapsar [3]. Uyarlanabilir dinamik programlama, her eylem veya her adımda alınan ödül veya cezaya göre tekrar

programlama yapar ve en verimli yöntemi bulmaya çalışır. Şekil 2.5'de sistemin genel çalışma prensibi incelenebilir.



Şekil 2.5: Uyarlanabilir Dinamik Programlama Modeli.

2.6 Q Öğrenmesi

Q öğrenmesi durum ve eylem ikililerinin gerçek değerini tahmin etmeye dayalı öğrenmedir. $Q(s,a)$ s durumunda a eylemi gerçekleştirilirse elde edilebilecek ödülü ifade eder ve en verimli ikili olarak belirlendiyse robotun s durumda a eylemi yapması beklenir. Q öğrenmesinin temel özellikleri aşağıdaki gibidir:

- Pekiştirmeli öğrenme türü olan Q öğrenmesinde, her eylemden sonra eyleme bağlı olan durumların Q değerleri değişecektir.
- Sınırlı sayıda durum,eylem ikilisi olması şartıyla ve bu ikililerin çok kez denenmesi durumunda Q değerlerinin yakınsayacaktır [4].

- Herhangi bir sıra ile yapılan durum-eylem ikililerinden öğrenme sağlanabilir. Yani belirli bir sıra ile eylemlerin yapılması gerekli değildir.

Yukarıdaki özelliklerden de çıkarılabileceği gibi, Q öğrenmesinde eylemler ve eylemler sonucunda elde edilen ödüller kaydedilmez, bunlardan ziyade Q değerleri hesaplanır ve saklanır. Bir s durumunda ve a eyleminde optimum Q değeri $Q^*(s,a)$ şeklinde ifade edilir. Bu ifade s durumunda iken a eylemi uygulanırsa elde edilebilecek Q değerini simgeler. Eğer en optimum durum eylem ikilisi bu ise robot maksimum Q değerine ulaşacaktır ve bu durum için yöntem bu olarak seçilecektir.

Q öğrenmesinde durum-eylem ikilisinden elde edilecek ödül ise 2.2 eşitliğindeki gibi gösterilir.

$$Q : SXA \rightarrow R \tag{2.2}$$

Q öğrenmesinde bir durumun s ilgili eylem a için Q değer güncellemesi ise 2.3 denklemi ile hesaplanır.

$$Q'(s,a) = Q(s,a) + \alpha[R + (\gamma maxQ(s',a)) - Q(s,a)] \tag{2.3}$$

2.3 denkleminde; $Q'(s,a)$, durum-eylem ikilisinin bir sonraki adımdaki Q değerini, $Q(s,a)$, durum-eylem ikilisinin şu andaki değerini, α öğrenme katsayısını, R ise anlık ödülü temsil ederken, γ ise gelecek ödüllerin önemini ifade eder. Formüldeki $maxQ(s',a)$ ise tahmin edilen maksimum Q değerini ifade etmektedir.

Anlaşılabileceği üzere, Q öğrenmesi tahmini Q değerleri üzerinden hareket eder. Bu şekilde durum geçişleri üzerinde olasılıksal karmaşık hesaplar yapmadan geleceğe yönelik olarak durum-eylem ikililerinin tahmini Q değerleri üzerinden öğrenme sağlar. Bu öğrenme şekli ile işlem olarak verimlilik elde edildiğinden tercih sebebidir.

2.6.1 Eylem seçim metotları

Daha önceden de belirtildiği gibi pekiştirmeli öğrenmenin diğer öğrenme türlerinden en önemli farkı, her eylem sonrası elde edilen ödülün/cezanın bir talimat gibi alınmasından ziyade, yorumlanarak durum-eylem ikilisinin gelecek ve geçmişteki durumlara göre değerlendirilmesidir. Bu sayede aktif bir şekilde deneme-yanılma

araştırması yapabilmektedir [4]. Değerlendirme süreci sonunda, elde edilen bilgi en iyi veya en kötü davranış olmamaktadır; çünkü değerlendirmede kullanılan fonksiyonlar optimizasyon temelli metotlardır. Diğer taraftan, ödülleri talimat gibi algılayan öğrenme türlerinin başında gözetimli (supervised öğrenme) gelmektedir. Ödülleri değerlendirmekten ziyade, her yapılan eylem ve sonuç kaydedilir ardından sonuç ile ilişkilendirilmeye çalışır.

Pekiştirmeli öğrenmede durum-eylem ikilileri için araştırma-faydalanma dengesinin kurulması büyük önem taşımaktadır. Bu denge için ise eylem seçim metodunun iyi belirlenmesi ve parametrelerinin doğru seçilmesi gereklidir. Robotun sürekli faydalanma yönüne gittiği, yani mevcut bilgilerle eylem seçmesi ve uygulaması devamlı aynı yolu kullanarak hedefe gitmesine yol açar, dolayısıyla hedefe giden daha iyi bir yol varsa, robot tarafından sürekli göz ardı edilecektir veya hedef çok düşük ödüllere sahip bir yolda olabilir ve ödüller düşük olduğu için robot bu yolu denemez ve yüksek ödüllerden faydalanma yoluna giderse hedefe hiçbir zaman ulaşamayabilir. Diğer yandan, robot sürekli araştırırsa, elindeki bilgilerden yararlanarak belirli bir yol seçmezse ilerleme kaydedemez ve dolayısıyla bir öğrenme sağlanamaz. Bu yüzden araştırma ve faydalanma dengesinin iyi kurulması gereklidir.

Ödülü değerlendiren metotlardan ε-greedy ve SoftMax eylem seçim metotları ağırlıklı olarak incelenecektir. Tez kapsamında da SoftMax eylem seçim algoritması motivasyon teorisi ile genişletilmiş ve ε-greedy, SoftMax ve geliştirilen hali karşılaştırılmıştır.

2.6.1.1 Greedy ve ε-greedy metotları

Pekiştirmeli öğrenmede en temel eylem seçim metodu tahmini değeri en yüksek olan ödüle sahip olan eylemi seçmektir, bu terim, öğrenme esnasında t, açgözlü olarak bir eylem a^* seçimi yapmaktır ve $Q_t(a^*) = max_a Q_t(a)$ şeklinde tanımlanabilir. Bu metot her zaman anlık olarak ödülün daha fazla olduğu eylemi seçer ve dolayısıyla diğer eylemlerden örnekleme yapıp daha iyisinin olup olmadığını kontrol etmez [4]. Bu çözüme yakın ama basitçe bir çözüm olarak ε-greedy metodu gösterilebilir. Bu metotta, genel olarak açgözlü algoritma şeklinde çalışır ama çalışma esnasında ε olasılık ile rastgele başka bir eylem seçimi yapar ve bu eylem üzerine hareket eder. Bu metodun avantajı, çok fazla deneme yanılma yapıldığında her eylemin denendiğinin

garantisinin verilebilmesidir. Dolayısıyla $Q_t(a)$, $Q_t(a^*)$'a yakınsar diyebiliriz. Tabii ki bu garanti sadece asimptotik olarak doğrudur ve pratikte çok uygulanabilir değildir [4].

2.6.1.2 Softmax eylem seçim metodu

ε-greedy metodu verimli ve araştırma-faydalanma dengesi açısından çok tercih edilen bir metot olsa da en kötü eylem ile en iyi eylemin seçilme olasılığı aynı olmaktadır. Bu da algoritmada istenmeyen sonuçlar doğurmaktadır. Bu noktadaki çözüm ise eylemlerin seçilme olasılığının tahmini ödüle göre değerlendirilmesidir. Eylemlerin, tahmini değerlerine göre sıralandırılması ve ağırlık verilmesi yöntemini kullanan metot ise *softmax* eylem seçim metodudur. *Softmax* eylem seçiminde genellikle, Gibbs veya Boltzman dağılımı kullanılır. Bu metot aşağıdaki formüle göre belirli bir zamanda t, belirli bir olasılık ile eylem a seçer,

$$P(a)_t = \frac{e^{Q_t(a)/\tau}}{\sum_{b=1}^{n} e^{Q_t(b)/\tau}} \tag{2.4}$$

Bu denklemde, $P(a)_t$ eylem a'nın seçilme olasılığını adım t'de belirtmektedir. $Q_t(a)$ ise a eyleminin değer fonksiyonunu temsil etmektedir. τ ise rastgeleliliği kontrol eden pozitif parametreyi temsil etmektedir ve sıcaklık değeri olarak isimlendirilmektedir.

2.4 denkleminde τ sıcaklık ifade eden bir parametre görevi görmektedir. Yüksek sıcaklık değerlerinde eylemlerin seçilme olasılıkları neredeyse eşit olurken, düşük sıcaklık değerleri ise eylemlerin beklenilen ödül değeri olasılıklarını ciddi oranda etkileyecektir. Dolayısıyla τ değerinin 0'a yakınsadığı durumda bir eylem için beklenilen ödül değerinin olasılığı 1'e yakınsayacaktır.

2.6.1.3 Robotların pekiştirmeli öğrenme ile tutmayı öğrenmesi çalışmaları

Tez çalışmasındaki odak nokta, robot tarafından nesnelerin tutulabilirliğinin daha hızlı öğrenilmesi ya da duruma göre robotun pes etmesinin gerekliliğinin hesaplanmasıdır. Bu doğrultuda daha önce yapılan tutulabilirlik çalışmalarında [5], [6], cisimlerin şekilleri belirli bir nesneye benzetilmeye çalışılmıştır. Cisimlerin benzetilebilenleri üzerinde tutulabilirlik noktaları belirlenmiş ve bu noktalara uygulanabilecek olan kuvvet hesaplanmıştır ve robotun kuvvet uygulaması sağlanmıştır. Uygulanan kuvvet neticesinde ise cismin tutulabilirliği teorik olarak hesaplanmıştır. Bir başka çalışmada ise tutma başarısının dış etkenlere ve kuvvetlere de bağlı olarak en verimli şekilde nasıl

tutulabileceğine dair bir araştırma yapılmıştır. Uygulanacak olan kuvvet ile sürtünme kuvvetinin birlikteliğinin üzerinde durarak optimizasyon problemi şeklinde çözüm geliştirmiştir ve minimum enerji ile tutma başarısı ölçülmüştür [7]. Diğer bir yandan, çok az ön bilgiye sahip nesnelerin tutulabilinirliği hakkında, kuvvet geri beslemesi bilgisine dayanarak çözüm hedeflenmiştir. Çalışmada görüntü işleme yöntemleri kullanılmış olsa da nesnenin çeşitli noktalarının denenmesi ve kuvvet sensörlerine geri besleme sağlanması ile de tutulabilirlik hesabı yapılması sağlanmıştır. Sadece robotun kendisinden gelen kuvvet sensörü bilgileri sebebiyle ortam kısmen keşfedilebilir olarak tanımlanmıştır [8]. Detry ve diğerlerinin çalışmasında ise tutulabilirlik başarısı, çeşitli kaynaklardan gelen tutma konfigürasyonlarının tekrar denenmesi ve deneme esnasındaki tüm verilerin kaydedilerek ölçümlenmesi ile sağlanmıştır [9]. Tutma konfigürasyonları, taklit yoluyla öğrenme ya da doğrudan tutma noktaları olarak çeşitlendirilmiştir. 3 boyutlu düzlemde, tutma açısı ve pozisyonu nesnenin özellikleri, nesnenin bulunduğu düzleme göre birleştirilerek tutulabilirlik başarılı ölçütü hesaplanmıştır. Bu çalışmada, önceden bilinmeyen cisimlerin tutulmasının karmaşıklığı zorlu bir problem olarak belirtilmiş ve sistemin karmaşıklığı ile birlikte çok değişkenlik gösterebildiği vurgulanmıştır. Bu karmaşıklık, seçilen sensörler, çevre bilgisi ve sahne konfigürasyonu sebebiyle çok artabilmektedir. Kroemer ve diğerlerinin çalışmasında ise karmaşıklığı çözmek için, pekiştirmeli öğrenme ile gözetimli öğrenme yöntemlerini birlikte kullanılmaktadır [10]. Gözetimli öğrenme ile bilinen ya da ne olduğu tahmin edilebilen cisimler için başarılı tutma şekillerini seçmektedir ve pekiştirmeli öğrenme ile bu seçimlerin ilgili nesne için ne kadar doğru olduğu öğrenilmektedir. Başka bir çalışmada robotun hareketinin, her bir anındaki parametreleri öğrenilmektedir [11]. Aynı zamanda bu paremetreler ile birlikte, nesnenin şekli ve robotun hedefi ile ilgili parametreler de öğrenilmektedir. Bir diğer çalışmada [12], 3 boyutlu görüntü veren kameradan gelen bilgiler ile cismi çevreleyebilecek sanal bir kutu oluşturulmuştur. Bu kutu ile cismin büyüklüğü ve şekli tahmini olarak hesaplanarak tutulabilecek noktalar saptanmaya çalışılmıştır. Çalışmada vurgulanan nokta ise cismi tutabilmek için en önemli özelliğin cismin tanınması veya öğrenilmesi değil, cismin şeklinin üzerinden yorum yaparak yola çıkmak ve tutma noktası belirlemek olduğu belirtilmiştir.

3. İÇSEL MOTİVASYON

Yaşayan organizmalar için çok farklı ölçeklerde ve çeşitte motivasyon sistemi mevcuttur. En basit hali ile bir çeşit kimyasal enerjiyi sürdürmek veya vücut sıcaklığını koruyabilmek ya da bir ortama uyum sağlayabilmek/yaşayabilmek için motivasyon etkenleri önemlidir. Bu tür motivasyon kriterlerinden yola çıkarak, akıllı robotlarda benzer sistemler geliştirilmiştir. Otonom bir yaşam ve yaşam boyu öğrenme üzerine çalışmalar [13] yapılmıştır. Diğer örnekler, tespih böceğinden esinlenen robotlar [14], dua edebilen robot [15] ve köpek benzeri robotlar [16] çalışmalarında değerlendirilmiş ve geliştirilmiştir.

Bazı hayvanlarda ve özellikle insanlarda bulunan motivasyon tipleri ise keşfetme, değiştirme veya ortama uyum sağlama, merak etme veya yeni etkinlikler ile eğlenme şeklinde sıralanabilir. İçsel motivasyon kavramı ise algısal gelişim açısından insanlarda oldukça önemlidir ve psikoloji biliminde bu kavram üzerine bir çok çalışma yapılmıştır [17], [18], [19]. Bu sebeple, bu kavram son yıllarda robot gelişiminde ve insan robot etkileşiminde ön plana çıkmaya başlamıştır [20], [21].

3.1 Psikologların Bakış Açısı ile İçsel Motivasyon

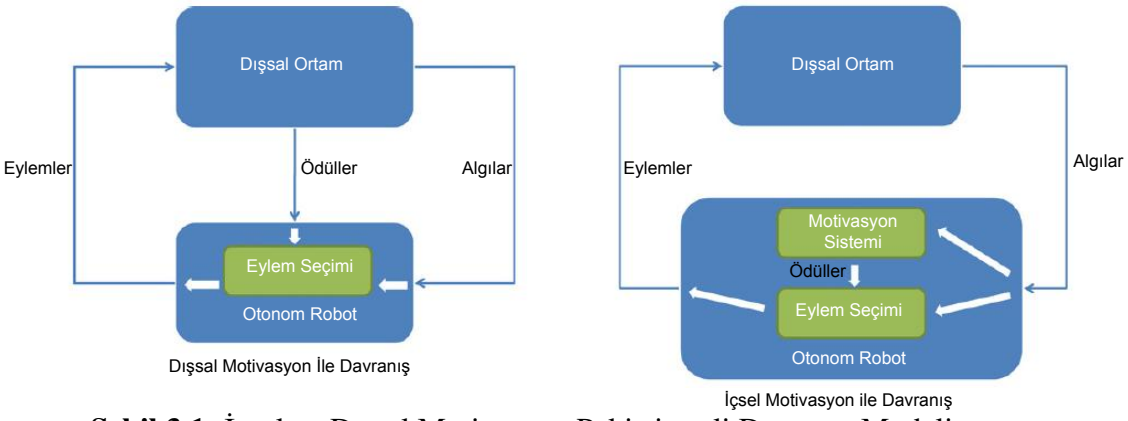
İçsel motivasyon özellikle çocukluk yaşlarında etkili olsa da ilerleyen yaşlarda da etkinliklerimizi yerine getirmek için önemli bir etkendir. Çocuklarda yeni bir cisim ile karşılaştıklarında göze çarpan davranışlar, ısırma, fırlatma, tutma, yakalama veya bağırmadır. Daha büyük yaşlarda ise bulmaca çözmek, resim yapmak, kitap okumak gibi davranışlarda içsel motivasyonun etkisi yadsınamaz derecede önemlidir [22].

İçsel motivasyon ile dışsal motivasyonun anlamlarının da incelenmesi gereklidir. İçsel motivasyon Ryan ve diğerlerinin çalışmasına göre, "İçsel tatminkarlığı artırıcı ama dışsal hiçbir ürün ya da faktör ile ilgili olmayan şekilde yapılan eylemlerin tümüdür.". Dışsal motivasyon ise "Yapılan herhangi bir etkinlik dışsal bir ürün ya da faktörden yararlanmaya yönelik ise bu tür eylemlere dışsal motivasyon denir." [19].

Bu tanımlara bakıldığında bu iki kavram birbirinin tersi olsa dahi insanın her davranışı içerisinde biri varken diğeri olamaz gibi bir durum yoktur. Bir örnek vermek gerekirse, bir öğrenci ödevini ailesinin tutumu yüzünden yapıyorsa bu dışsal bir motivasyondur; ama aynı durumdaki başka bir öğrenci ileriki hayatı için iyi olacağını ve ona fayda sağlayacağını düşünüyorsa bu içsel motivasyon kapsamına girmektedir. Kaldı ki eğer hem ailesi zorluyor hem de kendisi yeni bilgiler öğreniyorum şeklinde düşünceye sahipse iki motivasyon da bir arada olabilir [19].

3.2 Bilgisayar Sistemlerinde İçsel ve Dışsal Motivasyon

Motivasyon etkeni robotların mimarisinde dolaylı yoldan gerçeklenirken, son yıllarda mimarilerde açık bir etken olarak ya da belirli parametreler ile kurgulanabilen bir noktaya gelmiştir. Örnek olarak, enerji ile ilgili bir motivasyon değerine robot sahipse ve ilgili parametre için belirli bir aralık tanımlandıysa, bu aralık dışına çıkıldığında parametre adına sistemin geneline sinyal gönderilebilmekte ve robotun enerji istasyonu aramasına sebep olabilmektedir [23]. Enerji seviyesi için 1 ile 0 arasındaki değerler göz önüne alınmış ve sürekli şekilde enerji seviyesi eylem seçim metoduna gönderilerek her zaman 1 değerine yakın tutulmaya çalışılmıştır. Motivasyonu sadece enerji seviyesine indirgersek, bir nevi motivasyon sürekli yüksek tutulmaya çalışır. Şekil 3.1'de içsel ve dışsal motivasyon sonucu davranış modelleri incelenebilir.



Şekil 3.1: İçsel ve Dışsal Motivasyon Pekiştirmeli Davranış Modeli.

3.2.1 Yeterlilik temelli içsel motivasyon ile pekiştirmeli öğrenme

Bu tür motivasyonda hedef bir tür meydan okumaya yöneliktir. Hedefin ne kadar zor olduğu ve hedefe ulaşırken gerçeklenen performans incelenir. Meydan okumanın gereği olarak robot burada hedefi başarmanın beklenilen performansını kendisi belirler ve kendisi ulaşmaya çalışır. Neticede hedefe ne kadar iyi ve verimli ulaşıldığının önemli olması kadar hedefe nasıl ve ne kadar verimle ulaşıldığı da önemlidir [22]. Bir diğer çalışmada vurgulanan konu, insanın ve örneklenen robotun bir hedef için kendi kendine koyduğu daha basit hedeflere ulaşarak asıl hedefe gitmesidir [24]. Bu şekilde motivasyonun sürekli dinamik olarak değiştiği formülize edilmiş ve bu şekilde öz uyarlanan hedeflerin belirlenebildiği belirtilmiştir. Yeterlilik temelli içsel motivasyon ile kurgulanan pekiştirmeli öğrenmede de olasılık modelleri ve tahmin mekanizmaları uygulanabilir ama zorunlu değildir.

Biraz daha detaylı bakıldığında, öğrenilen durum-eylem planları için tecrübeyi yorumlayan bir yapının bulunması gerekir ve bu yapı eylemleri planlarken ve hedefleri belirlerken görev alır. Aynı zaman öğrenilen her yeni bilgi bu modül ile bağlantılı olmalıdır. "know-how" yapısının yanı sıra motivasyon modülü de bulunmalıdır. Bu modül ise performans kriterlerine göre ödülü değerlendirecektir.

Sonuç kadar sürecin de önemli olması ve bahsedilen iki ayrı modül sayesinde, pekiştirmeli öğrenme ile öğrenen bir çocuğun hatalarının da faydalı olabileceği, yeterliliklerinin geliştirilebileceği ortaya konulmuştur [24]. Pekiştirmeli öğrenmede, yeterlilik temelli içsel motivasyon metodu sayesinde olumsuz geri bildirimlerin çocuğun yeni yetenekleri kendine katabileceği ve geliştirebileceği gösterilmiştir. Çocuklardaki bu gelişim sürecinin robotlarda da uygulanabilir olduğu başka bir çalışmada belirtilmiştir [25]. Çalışmada içsel motivasyon pekiştirmeli öğrenmenin, robotun içerisinde bulunduğu ortam ile sürekli etkileşim içerisinde olarak, araştırarak, kendisini geliştirebileceği üzerine yoğunlaşılmıştır.

Pekiştirmeli öğrenme sistemlerindeki en önemli karar aşamalarından biri araştırma-faydalanma noktasındaki karar mekanizmalarıdır. Öğrenme algoritmalarının doğası gereği, robot mümkün olduğunca yüksek ödül alabileceği eylemleri seçer; ancak robot çok fazla açgözlü olursa, en verimli stratejiyi bulamayabilir. Aynı ikilem

küçük yaşlardaki çocuklarda da görülmektedir. Dolayısıyla hem insan doğası gereği hem de robot algoritmalarında dikkat edilmesi gereken önemli unsurlardan bir tanesi mevcut bilgiye dayalı çözüm ile yeni seçenekleri keşfederek çözüm üretme arasında dengeyi sağlamaktır. Yeterlilik temelli içsel motivasyon metodu ile göreve ve yeterlilik durumuna göre dinamik olarak denge kurulabilmektedir. Bu aşamada insanlarda bulunan isteksizlik [25] kavramı devreye girmektedir ve isteksizlik motivasyonun bir etkeni olarak ifade edilmiştir. Çalışmada belirtildiği gibi, isteksizliğin genelde olumsuz bir duygu olduğunu ve bu sebeple de genelde öğrenme konusunda göz ardı edildiği vurgulanmıştır. Çalışmada durumun aksine aslında isteksizlik duygusu araştırma ile faydalanma durumları arasında dinamik bir denge kurabilmekte olduğu vurgulanmıştır. Sonuç olarak ise isteksizlik ile birlikte oluşan öğrenim hızındaki düşüş, duruma göre yeni öğrenilmeye başlanmış bir davranışta gereksiz denemeleri ortadan kaldırabilen bir faktör olabileceği belirtilmiştir. Ayrıca isteksizlik kavramından ziyade motivasyon konusunda ilginçlik ya da merak kavramı deneme yanılma algoritmaları için kullanılmıştır [26], [27]. Merak kavramını bilgiyi artırıcı bir arzu gibi tanımlarken, sıkılmışlık kavramını ise tam tersi olarak ifade etmektedir. İçsel motivasyon ile ilgili diğer bir çalışmada [28], otonom robotların bir görevi yerine getirirken tekrar öğrenme esnasında sadece hedefe ulaşıp ulaşılamamasına göre ödül verilmemesi gerektiği vurgulanmaktadır. Geçişli (hedefe ulaşılırken ortamdan alından geri bildirimlerin de değerlendirilmesi) bir ödül mekanizması gerektiği, dışsal ödüllere ulaşılırken bunun da içsel motivasyonu artırıcı ve geliştirici bir özellik olabileceği vurgulanmıştır. Merak ve ödül fonksiyonları üzerine yapılan içsel motivasyon çalışmaları haricinde, insanlardaki ideal seviyedeki isteksizliğin hızlı öğrenme ve faydalanma aşamasında çok etkili olduğunu ortaya konmuştur [29].

Pekiştirmeli öğrenmedeki en önemli problemlerden biri olan karmaşıklık sorunsalı, yeterlilik temelli içsel motivasyon metotları ile çözüm için kapı aralamıştır; ancak içsel motivasyon ile desteklenmeyen öğrenme sistemlerinde bu sorunsal başka yollarla aşılmaktadır. Karşılaştırmak adına, yapılan bir çalışmasında odaklanılan nokta tekrarlı metotların sınıflandırma teknikleri ve öğretici eşliğinde örnekleme metotları ile plan bulmasıdır [30]. Bu şekilde önemli ölçüde performans kazancı sağlanmıştır. Bu çözüm bilinmeyen cisimlerde ve öğretici eşliğinde örnekleme yapılmadığı durumlarda ihtiyaç olan verimliliği sağlayamamaktadır. Bahsedilen

çalışma gibi, başka bir çalışmada da vurgulanan, deneme yanılma ile öğrenme öncesinde taklide dayalı öğrenme karmaşıklığı belirgin ölçüde düşürmektedir [31]. Çalışmada odaklanılan nokta cismin veya görevin özelliklerinden ziyade robotun motorların birincil parametrelerinin öğrenilmesidir. Göreve göre benzer öğrenilmiş bir görev varsa önceden öğrenilmiş motor birincillerinin uyarlanması sayesinde görevi tamamlamaya ve yeni parametrelerin öğrenilmesine devam edilmesine odaklanılmıştır. [32] çalışmasında ise dokunma sensörleri kullanılarak öğrenme, cisimleri manipüle etme ve alet kullanımı üzerinde durulmuştur. İlk öğrenme adımında yine taklit ile öğrenme göz önünde bulundurulmuştur ve bu da çalışmadaki karmaşıklığı düşürmüştür. Ek olarak taklit yolu ile öğrenme, robotun olmazsa olmaz parametrelerini hızlıca öğrenmesi açısından da kullanılabilmektedir.

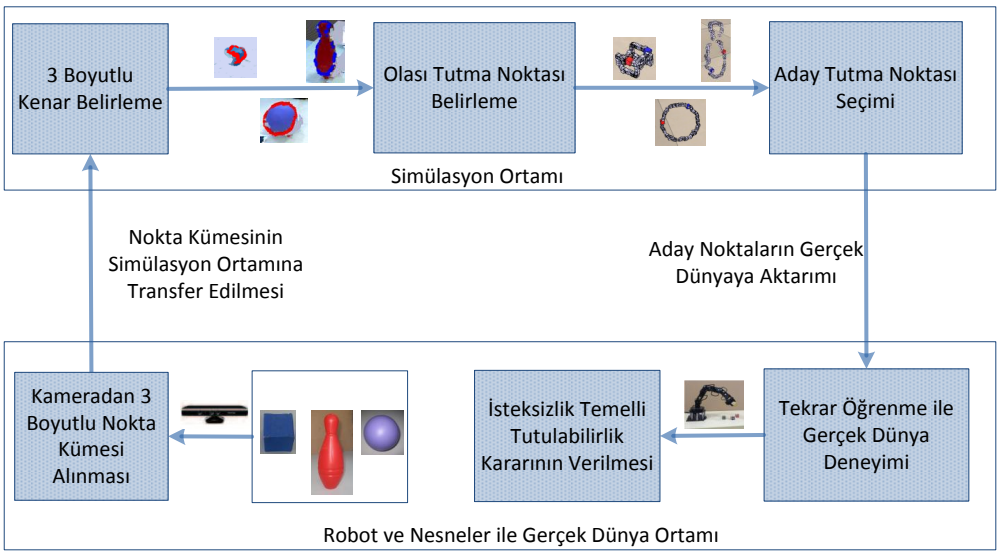
Önceki çalışmalara göre tez çalışmasının temel farklılığı insanlarda bulunan ve insanlar için modellenen ivecenlik kavramını [33] isteksizlik oranına ve yeterlilik kavramına entegre ederek insanlardaki dinamik olarak değişen motivasyon duygusunun robotlarda modellenmesidir. Bu model sayesinde, robotun deneme yanılma algoritmaları sonucu öğrenme için gereksiz efor sarf etmemesi hedeflenmiştir. Eğer görev başarılı sonuçlanacaksa buna hızlı karar vermesi sağlanmıştır. Eğer görev başarılı sonuçlanamayacak özelliklere sahipse robotun durumu fark etmesi sağlanmış, doğru zamanda pes etmesi sağlanmıştır. Sonuç olarak ise görevin tipine ve robotun bulunduğu ortama göre karar mekanizmasında dinamik bir yapı kurulmuştur.

4. BİLİNMEYEN NESNELERİN TUTULMASI

Bu bölümde bilinmeyen cisimleri tutmak için geliştirilen ve kullanılan yöntemler anlatılacaktır. Daha sonra bu yöntemlerin öğrenme ile ilişkisi aktarılacaktır.

4.1 Amaç

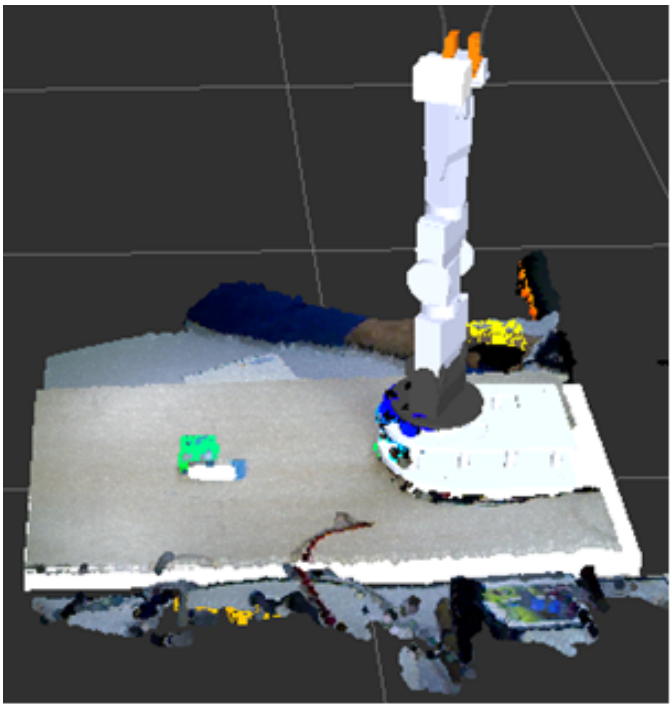
Tezde önerilen sistem içsel motivasyona dayalı pekiştirmeli bir öğrenme sistemi içermektedir. Bu öğrenme sistemi nesnelerin tutma noktalarının belirlenmesi ve bu noktaların gerçek dünyada denenmesi adımlarını içermektedir (Şekil 4.1). İlk adım, potansiyel tutma noktası çiftleri bulmak için gerekli olan metotlar ve işlemlerden oluşmaktadır. Nokta çiftleri bulunmasının sebebi robot kolunun uç elemanının 2 parmaklı yapıda olmasından kaynaklanmaktadır. İkinci adım ise ilk adımda bulunan ve simülasyon ortamında test edildikten sonra potansiyel olarak belirlenen nokta çiftlerinin gerçek dünyada test edilmesini sağlayan metotlardan ve pekiştirmeli öğrenme metodu sayesinde öğrenme sisteminden oluşmaktadır.



Şekil 4.1: İçsel Motivasyon Temelli Öğrenme Sistemi Genel Bakış.

4.2 Nesnelerin Görsel Temsili

Çalışmada, sahne içerisindeki nesneler, masa üzeri nesne etkileşimini sağlamak amacıyla doğrusal bir platform üzerine yerleştirilmiş ASUS Xtion Pro Live RGB-D kamera ile belirlenmektedir. Çalışmada kullanılan sahne yorumlayıcı sistem, bilinen cisimlerin tanınması ile birlikte tanınmayan cisimlerin belirlenmesi ve özelliklerinin tanımlanması işlemini sağlayan algoritmaları içermektedir [34]. Bilinmeyen cisimlerin belirlenmesi ve özelliklerinin tanımlanması için PCL kütüphanesinden "Organized Point Cloud Segmentation with Connected Components" algoritmasının kullanılması tercih edilmiştir [35]. 3 Boyutlu kameradan gelen RGB verileri haricinde bu noktaların 3 boyutlu ortamda XYZ koordinat verileri de kamera tarafından bilgisayara aktarılır. Sürekli aktarılan görüntü verisinden algoritma RGB verisini ve XYZ koordinat verisi özelliklerini kullanarak birbirine bağlı olan pikselleri bulur ve işaretler. Böylece genel olarak bir cisim bulunmuş olur. Bulunan cisimdeki her bir piksel işaretli olduğundan, cismin sınırları veya kenarları hızlı bir şekilde belirlenebilir. Nesne hakkında bulunan bu iki veri ile nesnenin konumu, nesne üzerinde manipülasyon yapılacak ise noktalar kümesi ve nesnenin sınırları elde edilir. Çalışmada kullanılacak olan bu veriler kullanılarak robot kolunda kullanılmak üzere öğrenme algoritması için cismin tutma noktaları belirlenmiştir. Bir sahneye ilişkin örnek bir nokta bulutu ve bu bulut üzerinde görüntü işleme algoritması ile tanınan nesne Rviz görselleştirme aracında Şekil 4.2'deki gibi görülmektedir.



Şekil 4.2: Nokta Kümesi ile Temsil Edilen Sahne.

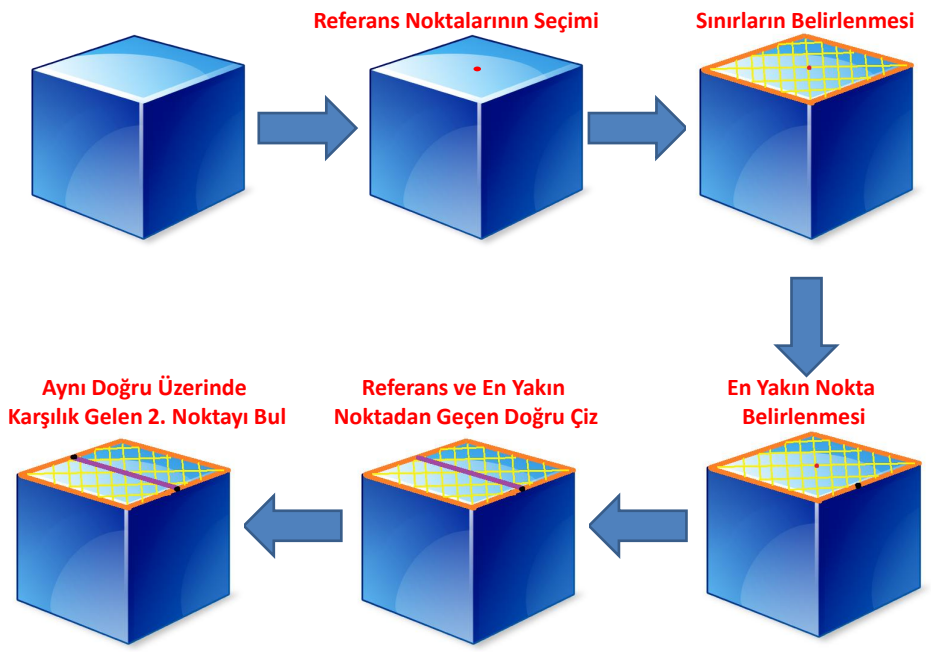
4.3 Tutma İçin Potansiyel Nokta Çiftlerinin Bulunması

Nesneler ağırlık merkezleri (μ) ve 3 boyutlu kenarları (H) ile temsil edilmektedir. Tutma noktaları çiftleri ise ($\rho = [p_1, p_2]$) şeklinde ifade edilirken Algoritma 1 kullanılarak bulunmaktadır.

Algoritma 1 Tutma Noktası Belirleme (μ, H)

Girdi: Nesne Ağırlık Merkezi μ, Nesne Kenarları Nokta Kümesi H
Çıktı: Tutma İkilileri P
Referans noktası olarak ref belirle $maxZ$, $minZ$ ve C.
for each referans noktası **do**
  $cNoktalar$ = `AynıDüzlemdekiNoktalarıBul()`
  $mNokta$ = `ReferansNoktasınaEnYakınNoktayıBul(`$cNoktalar$`)`
  $egim$ =`eğimiBul(`$mNoktas$`,`ref`)`
  for each $p \in cNoktalar$ **do**
    $Pegim$ =`EğimBul(`$mNokta$`,`p`)`
    if `AynıDoğruÜzerinde`($Pegim$,$egim$) **then**
      $P \leftarrow \{ p, mNokta \}$
    end if
  end for
end for

Şekil 4.3'de algoritmanın örnek bir küp nesnesi için çalışma adımları görülmektedir.



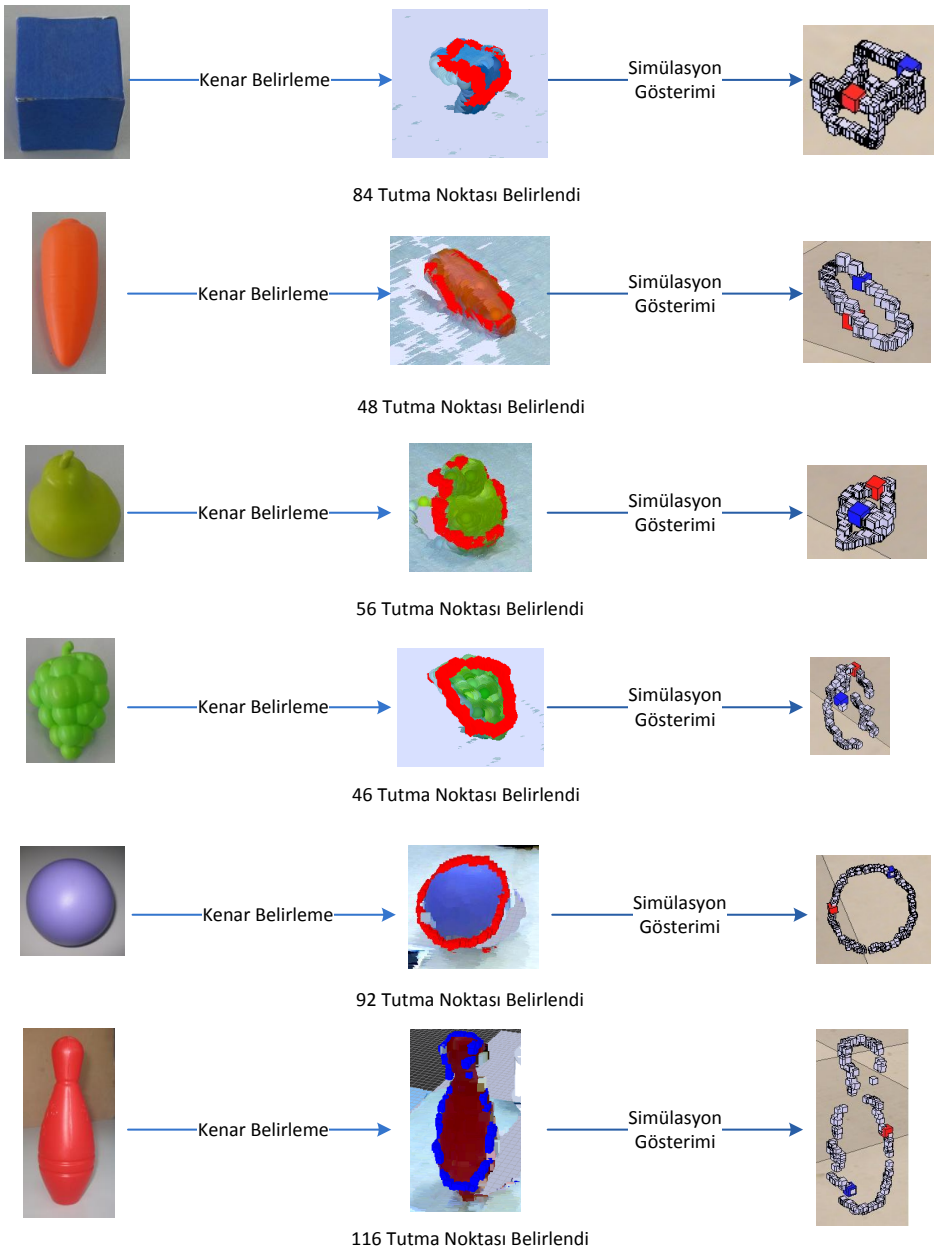
Şekil 4.3: Tutma Noktası Belirleme Algoritması Örnekleme.

Şekil 4.4'de ise 6 farklı nesnenin kenarları ve örnek olarak işaretlenmiş tutma nokta ikilileri görülmektedir.

4.4 Cismin Tutulabilirliğinin Öğrenilmesi ve Öğrenmeden Vazgeçilmesi

Önerilen sistemde, simülasyon ortamından alınan potansiyel tutma nokta ikilileri, test edilmesi ve başarılı olanların robot tarafından öğrenilmesi için gerçek dünya ortamına aktarılır. Bu denemeler esnasında kullanılan öğrenme metodu, isteksizlik temelli içsel motivasyonun ivecenlik kavramı ile harmanlanarak pekiştirmeli öğrenme sistemine entegre edilmesi ile geliştirilmiştir. Bu sistem sayesinde eğer cisim tutulamaz bir yapıdaysa robot çabucak pes edebilmektedir.

İvecenlik değeri, isteksizlik temelli motivasyon kavramının ilişkilendirileceği ve çalışmanın hedef noktası olan kavramı temsil etmektedir. İsteksizlik-Tepki-Sabır Üçlemesi açısından bakıldığında, yüksek ivecenliğe sahip bir kişi sabırsız olarak değerlendirilmekte, dolayısıyla hızlı isteksizliğe kapılmaktadır, bu da öğrenme yetisini direk olarak etkilemektedir. Bu değişkenlik göz önüne alınarak, ivecenlik değerindeki değişkenlikler ile robotun isteksizlik değerine direk etki edilebilir ve duruma göre farklı isteksizlik değerleri üretilebilir.



Şekil 4.4: Bilinmeyen cisimler için potansiyel tutma noktaları tutma noktası belirleme algoritması ile bulunmaktadır. 6 nesne bu işlemleri resmetmek amacıyla örnek olarak seçilmiştir. İlk adımda 3 boyutlu nokta kümesi kullanılarak 3 Boyutlu kenar belirlenmesi gerçeklenir. İkinci adımda ise nesnelerin nokta kümesinden kırmızı olarak işaretlenmiş tutma noktası ikilileri bulmaktır. Her cisim için toplamda kaç adet tutma noktası ikilisi üretildiği belirtilmiştir.

Robotun tutamayacağı cisim için hızlıca pes etmesinin yanı sıra asıl hedef, tutulabilecek cisim için robotun hızlıca tutulabilecek bir nesne şeklinde karar verebilmesi ve öğrenebilmesidir. Öğrenme sisteminin temelinde Q-Learning [1] algoritması ve SoftMax [4] eylem seçim stratejisi kullanılmaktadır. Bu öğrenme

sisteminde durum uzayı, simülasyonda üretilen potansiyel tutma nokta ikilileridir. Durum uzayının temsili:

$$S = [\mu, \rho, \phi, \omega, \vec{O_v}] \tag{4.1}$$

Denklemde, μ nesnenin ağırlık merkezi, ρ ise seçilen tutma noktası ikilisi $\rho = [p_1, p_2]$, ϕ tutma açısını temsil ederken, ω ise yaklaşma açısını temsil etmektedir. $\vec{O_v}$ vektörü 3 boyutlu öteleme vektörüdür. Bu vektörün sıfırdan farklı olduğu durum, robotun cismi itmesi veya cisme vurması sonucu nesnenin hareket ettiği durumdur.

Sistemde eylemlerin temsili:

$$A = [||\vec{R_v}||, \omega] \tag{4.2}$$

Denklemde, $||\vec{R_v}||$ vektörü x eksenindeki kayma miktarı ω ise ilgili cisme yaklaşma açısını temsil etmektedir.

Sistemde, robot başarılı bir eylem yürütürse ödül değeri 10 (R_{max}) olarak belirlenirken başarısız bir eylem yürütürse 0.1 (R_{min}) değerini [25] almaktadır. Bu değerler doğrultusunda kullanılan Q öğrenmesi algoritması denklem 2.3'de gösterilmiştir.

Sistemimizde 4 eylem seçim algoritması incelenmiştir. Birincisi ve en basit olanı ε-greedy eylem seçim metodudur(M_1). Bu metot büyük oranda en yüksek değere sahip eylemi seçer ama çok nadir olarak (ε olasılıkla) rastgele eylem de seçer. Seçim için dağılımlardan biri eşit öncelikli olarak kullanılır.

İkinci eylem seçim metodu ise sabit sıcaklık değeri [4] ile konfigüre edilmiş olan Softmax (Denklem 2.4) eylem seçim metodudur (M_2).

Üçüncü strateji ise (M_3) Softmax eylem seçim metodunu kullanır. Bu yaklaşımda, sıcaklık değeri olan τ parametresi değişkendir ve robotun isteksizlik seviyesine göre değişmektedir [25]. Optimal seviyedeki isteksizlik değeri daha araştırmacı bir davranışı desteklerken, yüksek değerlerde daha faydaya yönelik, hızlı sonuca yönelik davranışları destekler. Bu modeli robota uygulayabilmek adına denklem 4.3 geliştirilmiştir.

$$\frac{df}{dt} = -Lf + A_0 \tag{4.3}$$

denklemde, f anlık isteksizlik değerini, A_0 ise eylemin sonuç değerini (başarılı ya da başarısızlık) ve L ise kaçak değerini vurgulamaktadır. Denklem anlık isteksizlik değişimlerini fark edebilmek için kullanılır. L değeri ile integrali alınan isteksizlik değerinin yakınsaması sağlanmaktadır. Eğer isteksizlik değerinde ani ve büyük bir değişim olursa diferansiyel denklemin ürettiği değer bir anda yakınsanan değerden sapacağı için anlık isteksizlik değerindeki değişim yakalanabilecektir.

Denklem 4.3'de kaçak değeri (L) sabittir ve tüm simülasyonda değer 1'dir [25]. Yüksek değerlerdeki L isteksizlik değerinin yavaş yükselmesine sebep olacaktır. Bu da robotun daha uzun süre araştırmaya yönelik davranış göstereceğine işaret etmektedir. Bu değişkenlik sebebiyle bu çalışmada 4. metot önerilmiştir (M_4). Bu metotta L değeri değişkenlik göstermektedir. Dinamik L ise beklenilen faydaya dayalı motivasyon formülü 4.3 ile sağlanmaktadır [33].

$$L = \frac{beklenti * deger}{Z + \Gamma(T - t)} \tag{4.4}$$

Denklemde, beklenti maksimum beklenilen ödülün olasılığını temsil ederken, değer ise beklenilen eylem değerini (R_{max}) belirtmektedir. Z ise ödülün anlık olması durumunda kullanılan bir sabittir. Γ ise robotun gecikmeye duyarlılığını yani ivecenliğini belirtmektedir.

Çalışmada önemli olan bir diğer konu ise robotun bir cismi nasıl tutacağını öğrenmesi ile birlikte tutulabildiğini veya tutulamadığını öğrenmesi gerektiğidir. Tutma noktası çiftleri ρ ve eylem a denemesi, toplam isteksizlik değeri belirli bir seviyeye ulaştığında bitmektedir. Böylece tek nokta çiftinde ve eylem planında saatlerce deneme yapılması engellenmektedir. Üst seviyeyi belirleyen denklem:

$$Tolerans = e^{-||\vec{O}_v|| * \varphi} \tag{4.5}$$

Denklemde, $||\vec{O}_v||$ nesnenin ötelenme vektörünü temsil etmektedir. φ ise robotun göreve başladığından beri toplam deneme sayısını temsil eder. Bu hesaba bakıldığında her seferinde sabit değerler üretilmediği kolayca anlaşılabilir. Bu sayede eğer bir nokta çifti ve eylem planı çok kötü sonuç üretirse, robot hızlıca o çifti eleyebilmektedir.

Neticede ise robotun görevi tamamlamak adına ve öğrenmeyi bitirmek amacıyla bir eşik değerine sahip olması gerekmektedir. Bu değer ise nesnenin yapısına göre değişkenlik göstermektedir ve yine robotun isteksizlik katsayısı ile ilişkilidir.

$$isteksizlikLimiti = e^{1/\sqrt{n}} \tag{4.6}$$

Denklemde, n simülasyondan gelen, potansiyel tutma noktası sayısını ifade etmektedir.

4.5 İvecenlik Değeri ve Öğrenme Oranı

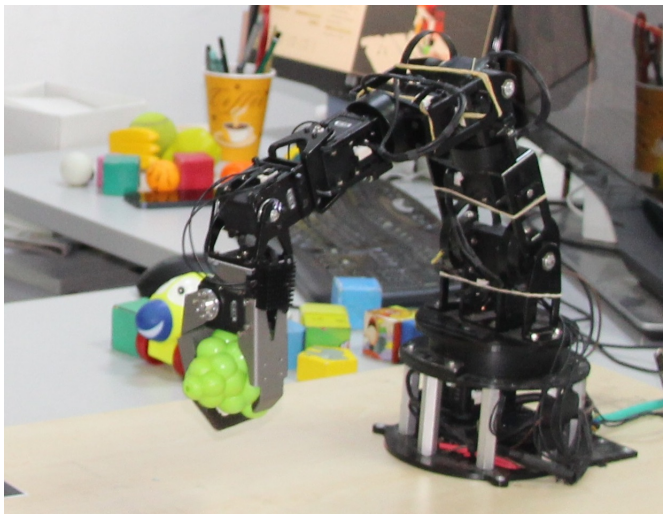
Çalışmadaki temel nokta tekrar vurgulanması gerekirse, ivecenlik değerinin öğrenme esnasında isteksizlik değerine olan etkisi incelenmektedir. Özellikle insan robot etkileşiminde, öğrenme hızı ve karar mekanizmasının hızı önemli bir konu olmaktadır [36]. Örnek vermek gerekirse, eğer bir robot insan ile hızlı bir oyun oynuyorsa hızlı karar vermelidir ve hızlı öğrenmeli, fakat diğer bir açıdan, eğer robot evde yalnızsa ve çok uzun zamanı varsa görevi tamamlamak için çok uzun zaman görev için harcayabilir. İvecenlik katsayısını değiştirerek, robotun isteksizlik değerinin belirli görevler için dinamik olarak değiştirilmesi sağlanabilir. Böylece farklı ortamlarda, farklı görevler için robot farklı öğrenme ve karar hızlarına sahip olur.

5. DENEYLER

Çalışmanın bu bölümünde, geliştirilen sistemin deneylerinden ve testlerinden bahsedilecektir. Öncelikle deney ortamı açıklanıp, ardından sonuçlar ve yorumlar verilecektir.

5.1 Deney Ortamı Bileşenleri

Cyton Veta robot kolu[1] (Şekil 5.1) tasarımı gereği insan kolu örnek alınarak geliştirilmiştir. Bu yüzden yedi serbestlik derecesine sahiptir. Çok serbestlik derecesine sahip olduğu için engeller arasında manevra kabiliyeti yüksektir ve nesneler ile başarılı etkileşim sağlayabilir. Çok serbestlik derecesine sahip olmasının bir diğer avantajı ise kolun ucunda bulunan tutucunun hedef pozisyonu için sınırsız sayıda konfigürasyon ile ulaşması mümkündür. Çizelge 5.1'de kolun eksen aralığı, elektriksel ve mekanik özellikleri verilmiştir.



Şekil 5.1: Cyton Veta robot kolu.

Cyton Veta robot kolu, fabrika çıkışında 2 paralel parmak tutucu ile gelmektedir; ancak Dynamixel FR07 – G101 tutucu seti ile değiştirilmiştir. Yeni setin en önemli özelliği maksimum açıklığının 15cm olmasıdır. Bu sayede daha geniş cisimler başarılı

[1] http://robai.com/robots/cyton-gamma-300/

Çizelge 5.1: Cyton Veta Robot kolu özellikleri.

Eksen Aralığı (Derece)	Mekanik Özellikler
7 Serbeslik Derecesi	Toplam Ağırlık: 1.2kg
Shoulder Roll – 300	Maksimum açıklıkta taşıma yükü: 300g
Shoulder Pitch – 210	Kol Uzunluğu : 53.4cm
Elbow Roll – 300	Erişebilme Mesafesi: 48cm
Elbow Pitch – 210	Maksimum lineer hız: 20 cm/s
Wrist Roll – 300	Maksimum eklem hızı: 30 rpm
Wrist Pitch – 210	Yinelenebilirlik: +/-0.5mm
Wrist Yaw – 210	Tutucu Maksimum açıklık: 3.5cm

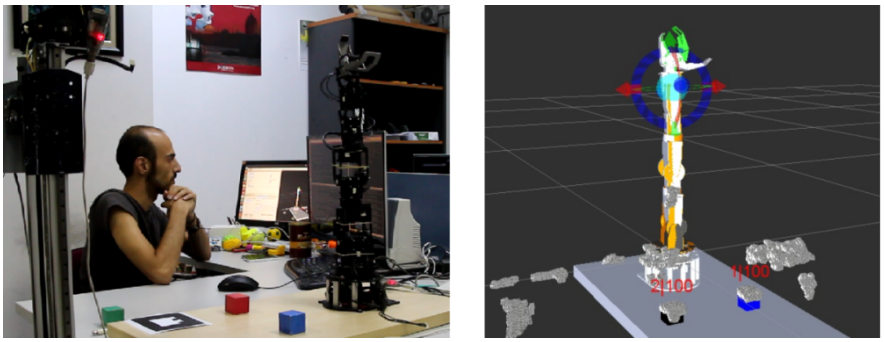
bir şekilde tutulabilmektedir. Ayrıca, robot kolu üzerinde bulunan hareketi sağlayan motorların bazıları daha yüksek torklara sahip servo motorlar ile değiştirilmiştir. Bu sayede daha uzun süre çalışabilen ve biraz daha ağır yükleri kaldırabilen bir robot kolu elde edilmiştir.

Cyton Veta robot kolu üzerindeki motorlar Papatya Zinciri modeli ile birbirine bağlanmıştır ve motorlar birbiri ile arasında RS 485 haberleşme protokolünü kullanmaktadırlar. Motorların kontrolünde bilgisayardaki USB arabirimi kullanılmaktadır. Bilgisayar ie motorlar arasındaki iletişim için ise USB2Dynamixel modülü kullanılmaktadır.

Robot kolu Linux işletim sistemi kullanılarak kontrol edilmesi tarafımızca tasarlandığı için ROS(Robot Operating System) açık kaynak kodlu altyapı kullanılmıştır. Bu altyapı içerisinde Dynamixel motorlar için hazırlanmış olan, Dynamixel kütüphanesinden faydalanılmıştır. Bu kütüphane sayesinde, her bir motorun konum, konum ile ilişkili hatası, hız, yük ve sıcaklık bilgileri anlık olarak alınabilmektedir. Motorlar pozisyon kontrolü ile sürülebildiği için kullanıcıdan sadece 3 boyutlu düzlemde konum tabanlı hedef alınabilmesi yeterlidir, böylece robot kolu ilgili noktaya tutucusunu götürebilecektir.

Çalışmada ROS kullanılması sayesinde, ROS sisteminde sağlanan rviz[2] görselleştirme aracıyla, robotların bilgi tabanını ve bilgisayar ortamında robot ortamının nasıl değerlendirildiğinin görselleştirilmesi mümkün hale gelmiştir.

[2]http://wiki.ros.org/rviz



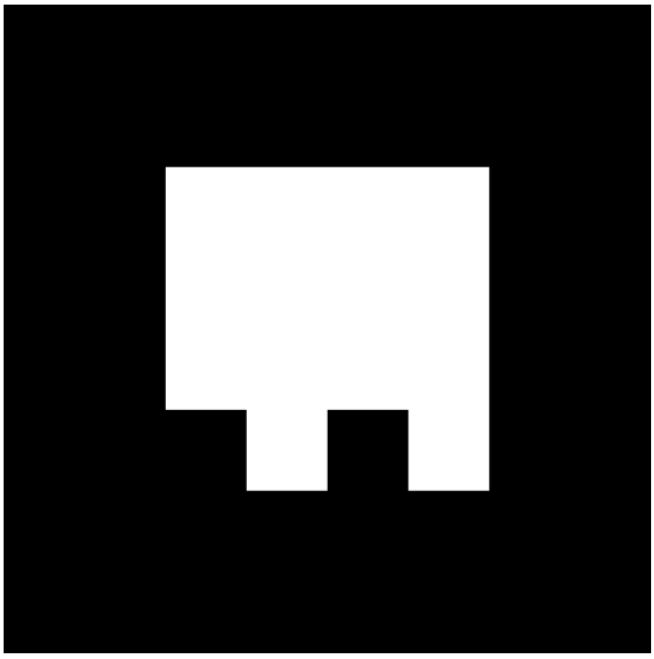
Şekil 5.2: (sol) Cyton Veta robot kolu (sağ) Cyton Veta robot kolunun rviz görseli.

Çalışmada, robot kolunun bir nesne ile etkileşimi için temel davranış modelleri tanımlanmıştır:

- Bir konuma gitme (tutucu için)
- Nesneye gitme (tutucu için)
- Nesneyi tutma
- Nesneyi bırakma

Bu davranış modellerinin ardı ardına belirli bir sıra ile kullanılması, daha karmaşık nesne etkileşim davranışlarının yaratılmasını sağlamaktadır. Nesne ile etkileşimin başlayabilmesi için ise bir sahne yorumlayıcısı ile nesnelerin tanınması ya da en azından saptanması gereklidir. Böylece sahne yorumlayıcı, robot kolunun içerisinde bulunduğu ortama nesneyi entegre edebilir. Aynı zamanda nesnenin sahneye entegrasyonu için sahnenin daha önceden tüm özelliklerinin yorumlayıcıya belirtilmiş olması gereklidir ya da sahnenin belirli bir noktasına yerleştirilecek bir işaret ile sahnenin özelliklerinin tanınması sağlanabilir. Bu amaçla robot kolunun bulunduğu zeminin köşesine Şekil 5.3'de görüldüğü gibi ARTag etiketi(Gerçek kamera pozisyonu ve oryantasyonu bilindiğinde köşeli yapısı sayesinde 3 boyutlu düzlemde nesnelerin konumlarını gerçek dünyadaki doğru yerlere yerleştirmeye yarayan etiketlerdir.) eklenmiştir. Bu etiket, RGB-D kamera aracılığı ile tanınarak sahnenin açısı, gerçek dünyadaki yeri, yüksekliği veya derinliği gibi özellikler sisteme aktarılabilmektedir. Böylece robot kolu bir cisim ile etkileşimi esnasında, sahnenin temel özellikleri değişse bile görevi olması gerektiği gibi tamamlayabilmektedir. Bunların yanı sıra, robot kolunun cisim ile etkileşimi açısından bir diğer önemli konu ise robot kolunun

cisimle ilgili görevi tamamlayabilmek için doğru konumlandırma ile hareket ediyor olması gerektiğidir. Konumlama işlemi esnasında 7 eksen hareket sağlayan motorların uygun açı ve tork parametreleri ile aktive edilmesi gereklidir. Bu aktivasyon için gerekli olan ters kinematik hesaplarının yapılması ve gerektiğinde kolun engellerden sakınması için kullanılan yol planlayıcısı ROS sisteminde bulunan MoveIt[3] aracıdır. Bu araç kullanılarak, robotun çizim ile gelen parametreleri ve motorların kendilerine has özellikleri göz önüne alınarak ters kinematik hesaplar yapılmakta ve her motor için ilgili tork ve açı değerleri motorlara gönderilmektedir, aynı zamanda sistem sürekli geri besleme alarak, hesaplama dışında bir durum oluşması durumunu gözlemlemektedir.



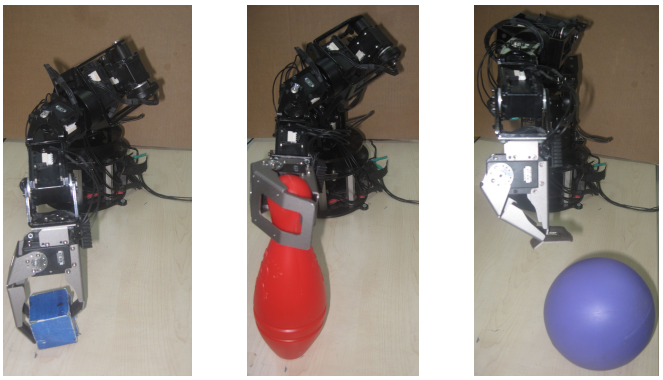
Şekil 5.3: Kullanılan ARTag.

5.2 Deney ve Sonuçları

Sistemi genel yapısını tekrar vurgulamak adına, potansiyel tutma noktaları öncelikle simülasyonda belirlenir ve gerçek dünya ortamındaki robot koluna aktarılır. Deney ortamında V-REP benzetim ortamı kullanılmıştır ve benzetim ortamından gelen sonuçlar robot koluna aktarılır. Aktarılan verilere göre robot kolunun üç farklı hali Şekil 5.4'de görülmektedir.

Tez çalışmasında önerilen sistemin ve öğrenme sürecinin başarım oranını ölçmek amacıyla dört farklı eylem seçim stratejisi incelenmiştir. Yüksek ivecenlik değeri

[3]http://moveit.ros.org



(a) Diklemesine başarılı tutulmuş örnek.

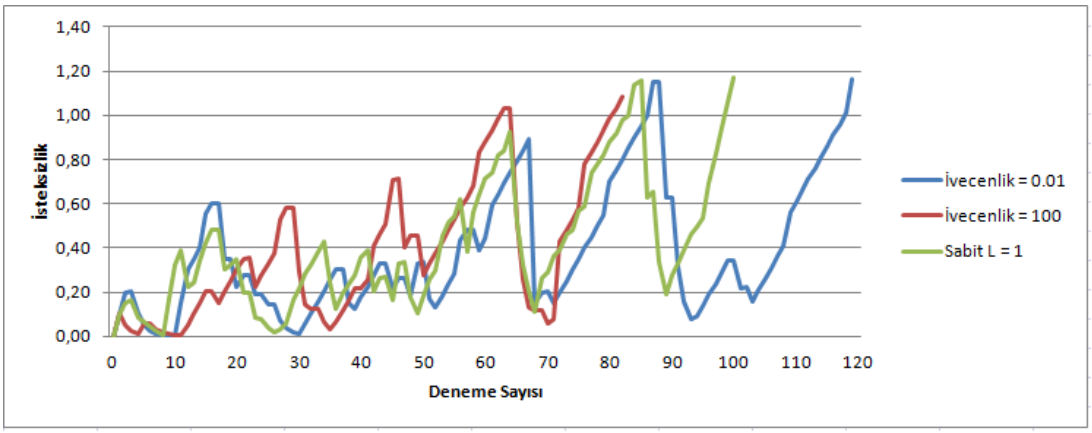
(b) Yanlamasına başarılı tutulmuş örnek.

(c) Diklemesine tutulamamış örnek.

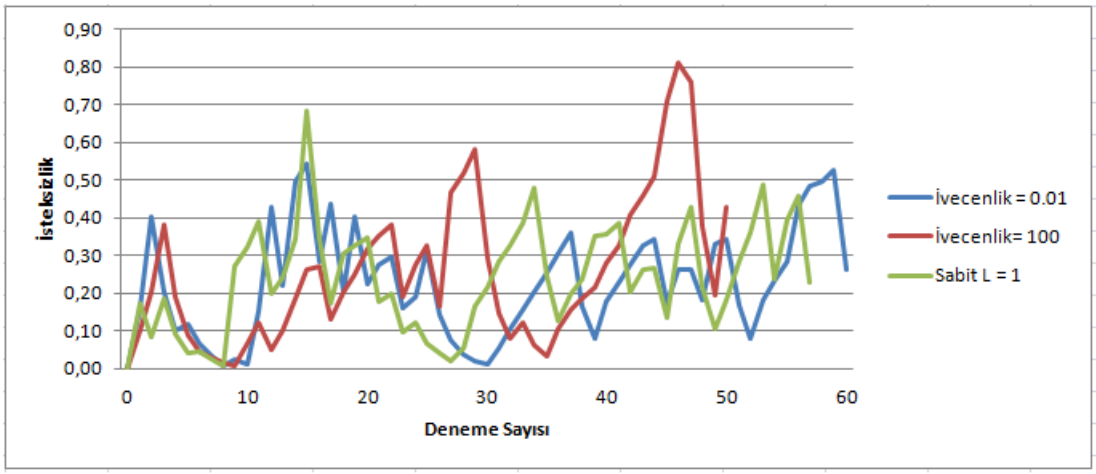
Şekil 5.4: 3 farklı nesne için robot kolunun hareketinin gösterimi. (a)Mavi küp göreceli olarak tutulması çok daha kolay (b) Plastik bowling labutu ise sadece üstten tutulabilir, birçok noktadan tutulması pek mümkün olmayan bir cisim (c) Plastik top ise tutulması mümkün olmayan bir cisim; çünkü sert ve büyük.

isteksizlik değerinin çok daha hızlı yükselmesine neden olacaktır. Bir diğer deyişle hızlı öfkelenen bir robota sebep olacaktır. Bu bilgiler ışığında, karşılaştırma adına ivecenlik değeri olarak 100 ve 0.01 değerleri verilmiştir. Deneylerin sonuçları, önerilen sistemi ve metodu desteklemektedir. Dolayısıyla, düşük ivecenlik değeri uzun süreli araştırmaya sebep olurken, bizim sistemimizde ise çok daha fazla tutma noktası ikilisi denemeyi sağlayacaktır. Deney sonuçlarından 6 tane nesne (küp, armut, havuç, üzüm, labut ve top) ele alınarak, karar ve öğrenme hızı karşılaştırılacaktır. (M_4) bizim önerdiğimiz dinamik isteksizlik kavramının ivecenlik ile sağlandığı sistemi temsil etmektedir. (M_3) ise sabit değer ile olan isteksizlik kavramını temsil etmektedir. Şekil 5.5, Şekil 5.6, 5.7 ve 5.8'de robotun küp, armut, havuç ve üzüm için isteksizlik katsayısını her deneme esnasında göstermektedir. Küp için benzetimde toplamda 84, armut için 56, havuç için 48, üzüm için ise 46 tane potansiyel tutma noktası ikilisi üretilmiştir. Robotun kolayca bu cisimleri tutalbildiği düşünüldüğünde, isteksizlik değeri genelde alt seviyelerde kalmıştır; ancak yeteri kadar deneme yapıldıktan sonra robotun isteksizlik değeri öğrenme sürecinin bitmesini sağlayan eşik değerine (4.6 denklemine göre) ulaşmıştır ve öğrenme süreci tamamlanmıştır.

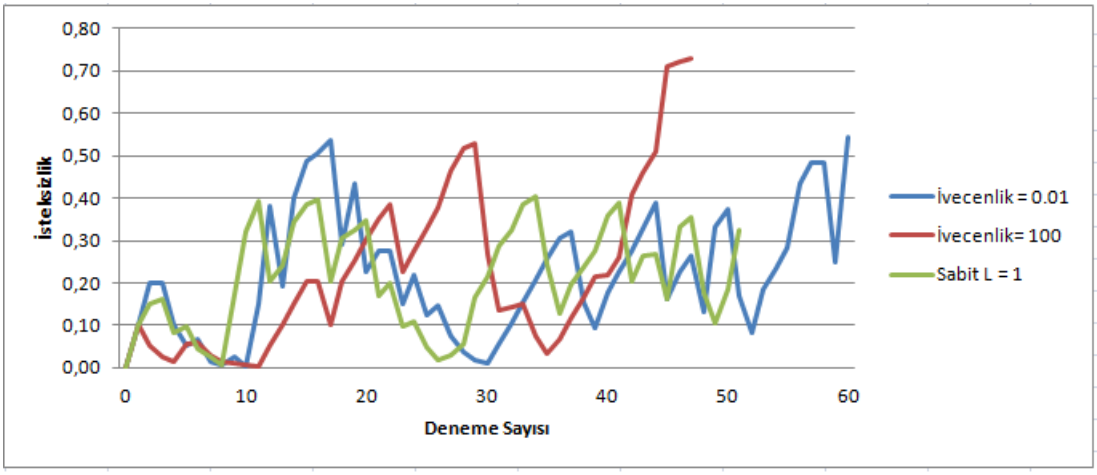
Labutun durumu incelendiğinde ise sonuçlar Şekil 5.9'de görülmektedir. Benzetim 116 tane potansiyel tutma noktası üretmiştir. Labutun çok hafif ve dengesiz olması durumunu göz önüne aldığımızda, kolayca devrilebildiği ve dolayısıyla göreceli olarak sahneden hızlıca yok olduğu anlaşılabilmektedir. Eğer labut düşerse, tolerans



Şekil 5.5: Küp için M_3 ve M_4 Metotları ile İsteksizlik Değeri Değişimi.

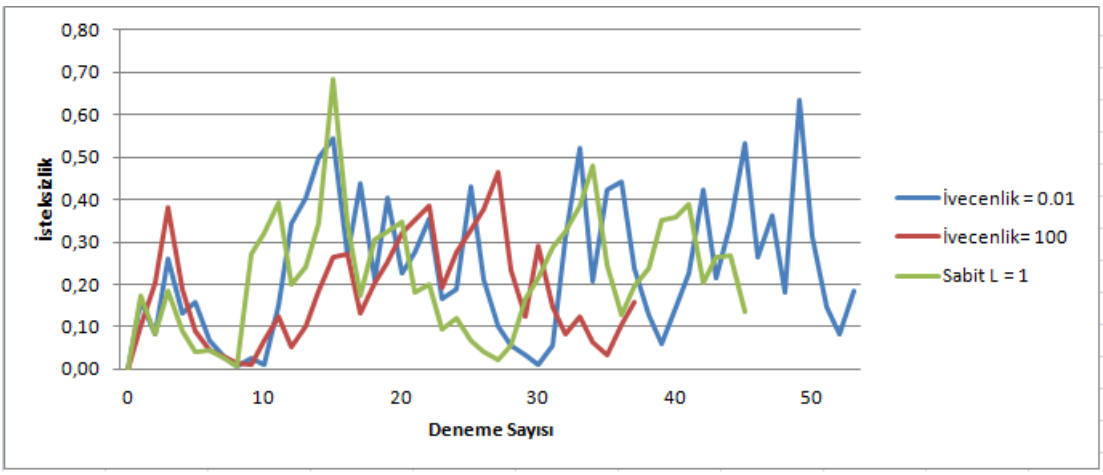


Şekil 5.6: Armut için M_3 ve M_4 Metotları ile İsteksizlik Değeri Değişimi.

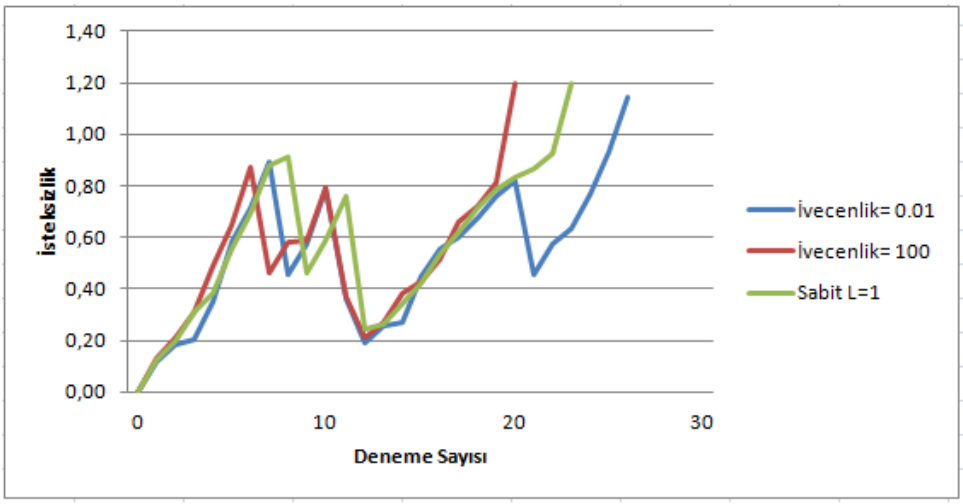


Şekil 5.7: Havuc için M_3 ve M_4 Metotları ile İsteksizlik Değeri Değişimi.

hızlıca düşmekte (Denklem 4.5'e göre) ve tutma ikilisi hemen başarısız olarak nitelendirilmektedir ve elenmektedir, dolayısıyla eğer hala limite ulaşılmadıysa robot hemen başka bir tutma ikilisi denemeye başlamaktadır



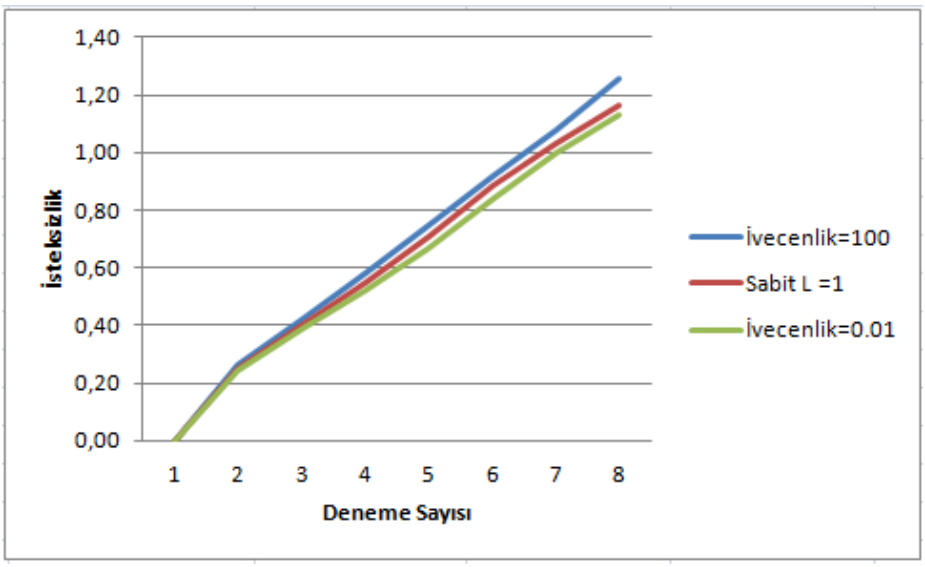
Şekil 5.8: Üzüm için M_3 ve M_4 Metotları ile İsteksizlik Değeri Değişimi.



Şekil 5.9: Labut için M_3 ve M_4 Metotları ile İsteksizlik Değeri Değişimi.

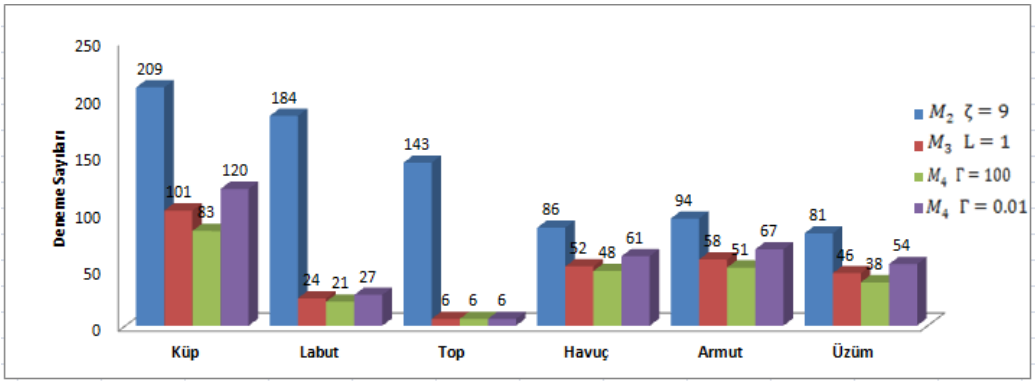
Topun durumu ise biraz daha farklıdır. Top sert plastikten yapılmıştır ve robotun tutabileceğinden daha büyüktür. Dolayısıyla tutulması imkansızdır ve her tutma girişiminde sahneden yok olmaktadır (Şekil 5.4(c)). Her denemede dolayısıyla robotun isteksizlik için tolerans değeri hızlıca düşmektedir ve hemen başka ikiliye geçilmektedir. Bunun yanı sıra cisim sahneden kaybolduğu için isteksizlik değeri de çok hızlı artmaktadır ve limite çok hızlı ulaşmaktadır. Toplamda 92 tutma ikilisi gerçek dünyaya transfer edilmiş olsa da birkaç denemeden sonra robot cismin tutulamaz olduğunu görüp vazgeçmektedir (Şekil 5.10).

Şekil 5.11'de ise en basit eylem seçim metodu olan ε-greedy metodu hariç seçim stratejilerinin öğrenme hızına olan etkileri karşılaştırılmıştır. Metodların karşılaştırılmasında denemelerin tamamı gerçek dünya ortamında yapılmıştır. ε-greedy metodu ile deneme sayısı çok fazla olduğu için, bu metodun deneme analizi ayrı bir grafikte



Şekil 5.10: Top için M_3 ve M_4 Metotları ile İsteksizlik Değeri Değişimi.

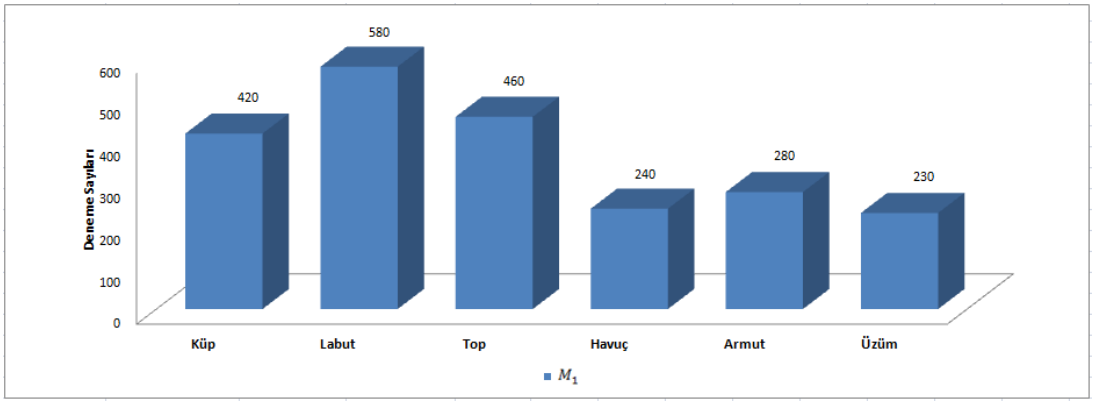
simülasyon ortamında denenerek verilmiştir. İsteksizlik kavramına dayalı eylem seçim metotları, standart SoftMax metoduna göre çok daha az deneme sayısına ihtiyaç duyarak öğrenmeyi sağlamaktadır.



Şekil 5.11: 4 Farklı Eylem Seçim Stratejisi İçin Deneme Sayıları.

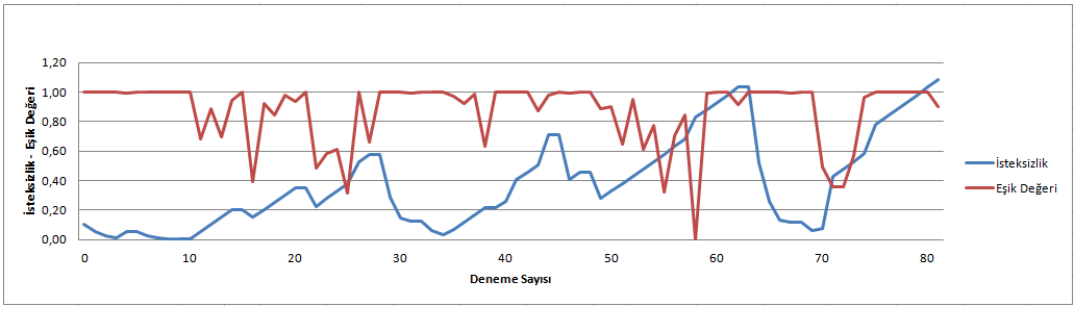
Şekil 5.12'de ilk metod olan ε-greedy metodu ile öğrenmenin tamamlanması için gerekli olan deneme sayısı gösterilmiştir. Şekil 5.11'deki metotlar ile karşılaştırıldığında çok büyük bir fark göze çarpmaktadır. Dolayısıyla öğrenme hızı düşük seviyelerdedir. ε-greedy metodu için yapılan denemeler ise çok fazla sayıda olduğu için simülasyonda tamamlanmıştır. Diğer metotlardaki gibi gerçek robot üzerinde analiz edilmemiştir.

Nesnelere göre belirlenen isteksizlik değerlerine göre robot tarafından tutulabilme veya robotun vazgeçme durumlarının yanı sıra, hangi durumlarda eşik değeri ile isteksizlik değerinin kesiştiği ve tutma ikililerinin ne zaman değiştiğinin analizi de yapılmıştır.



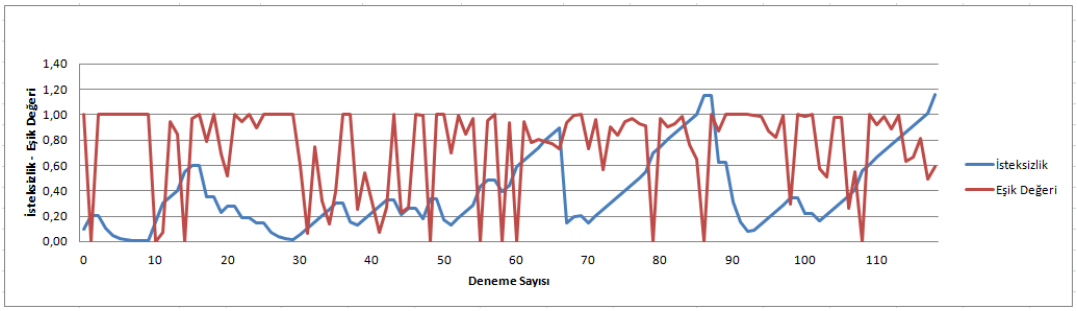
Şekil 5.12: ε-greedy Metodu ile Deneme Sayıları.

Şekil 5.13'e göre küp için ivecenlik değeri 100 iken robotun ivecenlik değeri 0.01 değerine göre daha sabırsız olduğu deneme sayısının azlığından anlaşılabilmektedir. Şekildeki isteksizlik ile eşik değerinin kesiştiği noktalarda robotun tutma ikilisi için beklentisinin kalmadığını ve başka bir tutma ikilisi denemek için adım attığını görebilmekteyiz. Başka bir ikili denerken, isteksizlik değeri bir öncekinin yarısına düşmektedir; ancak sıfıra yakınsamadığı için eğer robot bıkma evresine geldiyse eşik değerine çok hızlı yakınsayacaktır, bu sebeple de hızlıca pes etme ve öğrenip öğrenmeme kararına varabilecektir. İsteksizlik değeri 0.01 iken çok daha fazla denemede bulunduğu bıkma evresinin çok daha uzun sürdüğünü Şekil 5.14'den inceleyebiliriz. İsteksizlik değeri 0.01'ken robot çok fazla deneme yaptığı için isteksizlik değerinde çok dalgalanmakta olmaktadır. Daha önce denenmemiş bir çok tutma ikilisi bu süreçte farklı konfigürasyon ile denendiği için, eşik değerinde de büyük ölçüde dalgalanma olmaktadır. Eşik değerindeki en önemli faktör Denklem 4.5'e göre cismin kayma miktarı olduğu için bu grafikten aynı zamanda cismin tutulmaya çalışıldığı evrede çok fazla yer değiştirdiğini yorumlayabiliriz.

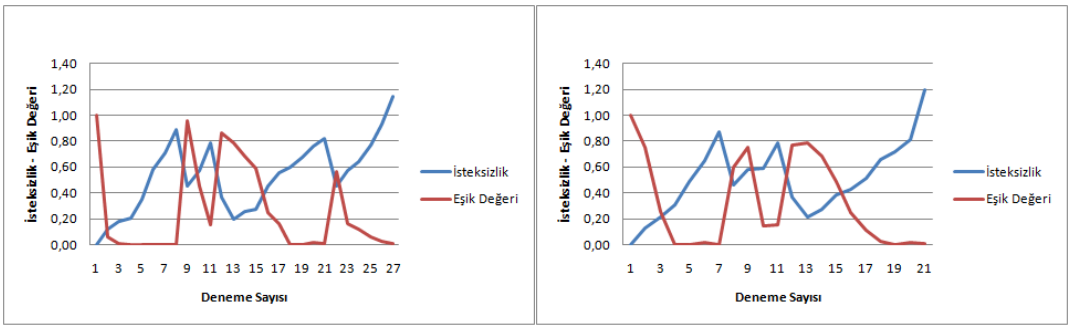


Şekil 5.13: Küp İçin İvecenlik Değeri 100 iken, İsteksizlik ve Eşik Değeri Değişimi.

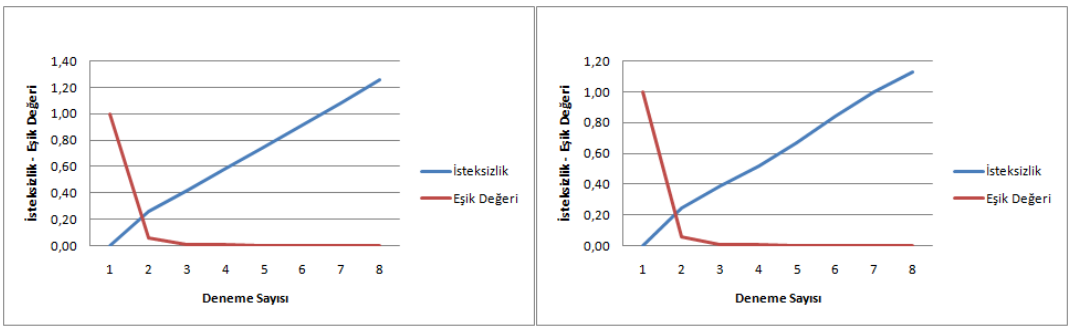
Şekil 5.15 ve 5.16 incelendiğinde cisimlerin tutma denemeleri esnasında, sahne içerisinde robot kolunun ittirmesi ya da tutamaması sebebiyle çok fazla yer



Şekil 5.14: Küp İçin İvecenlik Değeri 0.01 iken, İsteksizlik ve Eşik Değeri Değişimi.



Şekil 5.15: Labut İçin İvecenlik Değeri 100 ve 0.01 iken, İsteksizlik ve Eşik Değeri Değişimi.



Şekil 5.16: Top İçin İvecenlik Değeri 100 ve 0.01 iken, İsteksizlik ve Eşik Değeri Değişimi.

değiştirdikleri ya da sahneden yok oldukları çıkarılabilir. Cisimlerin daha önce bahsedilen özellikleri de göz önüne alındığında özellikle topun ortamdan çıkma gibi durumları çok olmaktadır, topun yanı sıra, labutun da düşmesi ve bulunduğu yerden çok farklı bir noktaya gitmesi durumları oluşabilmektedir. Cisimlerin tutma denemeleri esnasındaki fazla hareketleri sebebiyle, robotun her iki ivecenlik değerinde de iki cisim için de çoğu zaman eşik değeri çok düşük çıkmaktadır . Bu da sürekli olarak isteksizlik değeri ile çakışmaya sebep olmakta ve bu sayede başarısız olacağı aşikar olan tutma ikilileri için robot çok sayıda deneme yapamamaktadır. Topu hiçbir şekilde tutamadığı için isteksizlik değeri sürekli artmakta ve robot kısa zamanda denemeyi bitirmektedir. İsteksizlik değeri daha ilk denemelerden itibaren

eşik değerinin üstünde olduğu için robot her denemesinde farklı bir tutma ikilisi konfigürasyonu için eğitime devam etmektedir.

6. SONUÇ VE ÖNERİLER

Bu tez çalışmasında, robotların nesneler ile etkileşimi için önemli bir önem taşıyan tutma eylemi üzerine durulmuştur. Çalışmada içsel motivasyon kavramı ile entegre olmuş destekli öğrenme alt yapısı geliştirilmiştir. Geliştirilen altyapı robotların öğrenme ve karar verme hızını dinamik hale getirirken, eylemlerini zaman ve ortam şartlarına göre değiştirebilmelerini sağlamaktadır.

Konu olan robotlardaki karar verme dinamizmi için insan davranışları ve öğrenme süreçleri örnek alınmıştır. Daha önce yapılan çalışmalarda, içsel motivasyonun, insanların öğrenmesi üzerindeki etkileri incelenmiş ve robotlara çeşitli formüller ile uyarlanmıştır. Bu metotlar içsel motivasyon değerini genellikle sabit bir değere sabitleyerek, görev süresince robotun aynı değer ile hareket etmesini ön görmüşlerdir.

Daha önceki çalışmalara ek olarak, tez kapsamında içsel motivasyonun hesaplama metotlarında olmayan isteksizlik ve ivecenlik değerleri entegre edilmiştir. Bu değerler sayesinde robotun öğrenme eylemi için robotun hedeflediği göreve, içerisinde bulunduğu ortama ve zamana göre belirli tolerans değerleri dinamik bir şekilde süreç içerisinde hesaplanmaktadır. İnsani duygulardan örnek alınarak, eğer robot sınırlı olmayan bir zaman dilimi içinde bir şeyin nasıl yapılması gerektiğini kesinlikle öğrenmesi gerekiyorsa, eylem-durum uzayının tamamını çok uzun sürede deneyebilir; ancak robot, bir oyun esnasında hızlı karar vermesi gerekiyorsa ya da hızlı bir hamle yapması gerekiyorsa, tolerans değerleri düşük olacaktır ve öğrenme için gerekli olan süreci mümkün olduğunca kısa kesmek isteyecektir. Böylece öğrenme hızı ve seçeceği eylemler değişebilecek ve robot duruma göre farklı davranabilecektir.

Tezde geliştirilen metotlar, insan kolundan örnek alınarak geliştirilen yedi eksenli bir robot kolunda gerçeklenmiş ve denenmiştir. Deneyler esnasında farklı cisimler ile tutma eylemi üzerine testler yapılmıştır. İvecenlik değerinin değiştiği koşulda cisim farklılaştığında, her bir cisim için deneme sayısı farklı olmuştur ve bazı cisimlerin tutulamaz olduğuna, bazılarının ise tutulabilir olduğuna hızlı karar vermiştir. Bu sonucun yanı sıra, robot cisimlerin tutulabilir olanlarının tüm eylem-durum

uzayındaki olasılıkları denememiş ve belirli bir noktadan sonra tutulabildiğine karar kılmıştır. Deney sonuçları göstermektedir ki içsel motivasyon metotlarına entegre edilmiş olan ivecenlik değeri, ortama ve koşullara bağlı olarak öğrenme ve karar verme hızını dinamik olarak değiştirebilmektedir. İlerleyen çalışmalarda deney setinin genişletilmesi ve ivecenlik katsayısının farklı ortamlar ve zaman durumlarında değişkenliğinin ölçülmesi ve etkisinin analiz edilmesi düşünülmektedir.

KAYNAKLAR

[1] **Watkins, C.J. ve Dayan, P.** (1992). Q-learning, *Machine learning*, ***8***(3-4), 279–292.

[2] **Russell, S.**, **Norvig, P. ve Intelligence, A.** (1995). A modern approach, *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs*, ***25***.

[3] **Kulkarni, P.** (2012). *Reinforcement and systemic machine learning for decision making*, cilt 1, John Wiley & Sons.

[4] **Barto, A.G.** (1998). *Reinforcement learning: An introduction*, MIT press.

[5] **Bicchi, A.** (1995). On the closure properties of robotic grasping, *The International Journal of Robotics Research*, ***14***(4), 319–334.

[6] **Ding, D.**, **Liu, Y.H. ve Wang, S.** (2000). The synthesis of 3-D form-closure grasps, *Robotica*, ***18***(01), 51–58.

[7] **Buss, M.**, **Hashimoto, H. ve Moore, J.B.** (1996). Dextrous hand grasping force optimization, *Robotics and Automation, IEEE Transactions on*, ***12***(3), 406–418.

[8] **Platt, R.** (2007). Learning grasp strategies composed of contact relative motions, *Humanoid Robots, 2007 7th IEEE-RAS International Conference on*, IEEE, s.49–56.

[9] **Detry, R.**, **Baseski, E.**, **Popovic, M.**, **Touati, Y.**, **Kruger, N.**, **Kroemer, O.**, **Peters, J. ve Piater, J.** (2009). Learning object-specific grasp affordance densities, *Development and Learning, 2009. ICDL 2009. IEEE 8th International Conference on*, IEEE, s.1–7.

[10] **Kroemer, O.**, **Detry, R.**, **Piater, J. ve Peters, J.** (2010). Combining active learning and reactive control for robot grasping, *Robotics and Autonomous Systems*, ***58***(9), 1105–1116.

[11] **Stulp, F.**, **Theodorou, E.**, **Kalakrishnan, M.**, **Pastor, P.**, **Righetti, L. ve Schaal, S.** (2011). Learning motion primitive goals for robust manipulation, *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, IEEE, s.325–331.

[12] **Huebner, K.**, **Ruthotto, S. ve Kragic, D.** (2008). Minimum volume bounding box decomposition for shape approximation in robot grasping, *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, IEEE, s.1628–1633.

[13] **Arkin, R.C.** (2003). Moving up the food chain: Motivation and Emotion in behavior-based robots.

[14] **Endo, Y. ve Arkin, R.C.** (2001). Implementing tolman's schematic sowbug: behavior-based robotics in the 1930's, *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, cilt 1, IEEE, s.477–484.

[15] **Arkin, R.C.** (1998). *Behavior-based robotics*, MIT press.

[16] **Fujita, M.**, **Costa, G.**, **Takagi, T.**, **Hasegawa, R.**, **Yokono, J. ve Shimomura, H.** (2001). Experimental results of emotionally grounded symbol acquisition by four-legged robot, *Proceedings of Autonomous Agents*, cilt2001.

[17] **Berlyne, D.E.** (1960). Conflict, arousal, and curiosity.

[18] **Csikszentmihalyi, M. ve Csikzentmihaly, M.** (1991). *Flow: The psychology of optimal experience*, cilt 41, HarperPerennial New York.

[19] **Ryan, R.M. ve Deci, E.L.** (2000). Intrinsic and extrinsic motivations: Classic definitions and new directions, *Contemporary educational psychology*, ***25***(1), 54–67.

[20] **Barto, A.G.**, **Singh, S. ve Chentanez, N.** (2004). Intrinsically motivated learning of hierarchical collections of skills, *Proc. 3rd Int. Conf. Development Learn*, s.112–119.

[21] **Oudeyer, P.Y.**, **Kaplan, F. ve Hafner, V.V.** (2007). Intrinsic motivation systems for autonomous mental development, *Evolutionary Computation, IEEE Transactions on*, ***11***(2), 265–286.

[22] **Oudeyer, P.Y. ve Kaplan, F.** (2007). What is intrinsic motivation? a typology of computational approaches, *Frontiers in neurorobotics*, ***1***.

[23] **Konidaris, G. ve Barto, A.**, (2006). An adaptive robot motivational system, From Animals to Animats 9, Springer, s.346–356.

[24] **Baranes, A. ve Oudeyer, P.Y.** (2010). Maturationally-constrained competence-based intrinsically motivated learning, *Development and Learning (ICDL), 2010 IEEE 9th International Conference on*, IEEE, s.197–203.

[25] **Grzyb, B.**, **Boedecker, J.**, **Asada, M.**, **Del Pobil, A.P. ve Smith, L.B.** (2011). Between Frustration and Elation: Sense of Control Regulates the lntrinsic Motivation for Motor Learning., *AAAI Workshop on Lifelong learning*.

[26] **Schmidhuber, J.** (1990). A Possibility for Implementing Curiosity and Boredom in Model-building Neural Controllers, *Proceedings of the First International Conference on Simulation of Adaptive Behavior on From Animals to Animats*, MIT Press, Cambridge, MA, USA, s.222–227.

[27] **Lenat, D.B.** (1976). AM: An artificial intelligence approach to discovery in mathematics as heuristic search, **Teknik Rapor**, DTIC Document.

[28] **Uchibe, E. ve Doya, K.** (2008). Finding intrinsic rewards by embodied evolution and constrained reinforcement learning, *Neural Networks*, ***21***(10), 1447–1455.

[29] **Wong, P.T.** (1979). Frustration, exploration, and learning., *Canadian Psychological Review/Psychologie canadienne*, ***20***(3), 133.

[30] **Dimitrakakis, C. ve Lagoudakis, M.G.** (2008). Rollout sampling approximate policy iteration, *Machine Learning*, ***72***(3), 157–171.

[31] **Kober, J. ve Peters, J.** (2009). Learning motor primitives for robotics, *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, IEEE, s.2112–2118.

[32] **Chebotar, Y.**, **Kroemer, O. ve Peters, J.** Learning Robot Tactile Sensing for Object Manipulation.

[33] **Steel, P. ve König, C.J.** (2006). Integrating theories of motivation, *Academy of Management Review*, ***31***(4), 889–913.

[34] **Ersen, M.**, **Ozturk, M.D.**, **Biberci, M.**, **Sariel, S. ve Yalcin, H.** (2014). Scene Interpretation for Lifelong Robot Learning, *The 9th International Workshop on Cognitive Robotics (CogRob 2014) held in conjunction with ECAI-2014*, Prague, Czech Republic.

[35] **Trevor, A.**, **Gedikli, S.**, **Rusu, R. ve Christensen, H.** (2013). Efficient organized point cloud segmentation with connected components, *Semantic Perception Mapping and Exploration (SPME)*.

[36] **Sauser, E.L. ve Billard, A.G.** (2006). Biologically inspired multimodal integration: Interferences in a human-robot interaction game, *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, IEEE, s.5619–5624.

ÖZGEÇMİŞ

Ad Soyad: Erçin Temel

Doğum Yeri ve Tarihi: Eskişehir 1988

Adres: 19 Mayıs Mh., Turapoğlu Sk, Kozyatağı/İstanbul

E-Posta: ercintemel@itu.edu.tr

Lisans: Sabancı Üniversitesi Bilgisayar Mühendisliği

TEZDEN TÜRETİLEN YAYINLAR

- **Temel, E.**, Grzyb, B. J., Sariel, S., 2014: Learning Graspability of Unknown Objects via Intrinsic Motivation. *International Workshop - Artificial Intelligence and Cognition*, November 26-27, 2014 Torino, Italy.