

İçsel Motivasyon Sistemlerinde Sensör Motor Uzayının Tahmin Edilebilirlik Prensibiyle Bölünmesi

Melisa İdil Şener ve Emre Uğur
Bilgisayar Mühendisliği Bölümü
Boğaziçi Üniversitesi
İstanbul, Türkiye

Email: {melisa.sener, emre.ugur}@boun.edu.tr

Özetçe —İçsel motivasyon, öğrenen aracının (agent), öğrenilecek problemleri ve konseptleri, öğrenme esnasında kendi kendine seçmesini sağlayan ve çocuk öğrenmesinden ilham alınarak geliştirilmiş bir aktif öğrenme metodudur. Bu yaklaşım, bir robotun, bir çocuk gibi kendi kendine öğrenebilmesini sağladığı için, gelişimsel robotik alanında önemli bir yaklaşım olarak görülmektedir. Bu çalışmada, sensör motor uzayını adaptif bir şekilde bölünmesi için geliştirilmiş bir yöntem sunulmaktadır ve bu yöntem robotun bölünme sonrasında bir içsel motivasyon sistemi olan *Akıllı Adaptif Merak (Intelligent Adaptive Curiosity)* yöntemini kullanarak daha hızlı ve mantıklı bir şekilde bu uzayı öğrenmesini sağlar. AAM, sensör motor uzayını özniteliklerin benzerliğine göre böler ve her bölgenin öğrenme gelişimini içsel motivasyon kaynağı olarak kullanır. Sunduğumuz yöntem, sensör motor uzayını adaptif bir şekilde bölünmesi için, bölgelerin potansiyel hata oranının hesaplanması mantığıyla çalışmaktadır. Böylece oluşan bölgelerin her biri için bölünme sonrasında eğitilen öğrenme makinası daha yüksek bir performans sergiler ve semantik olarak daha mantıklıdır. Bu makalede, sunduğumuz yöntemin işleyişi açıklanmakta ve orijinal çalışmanın izlediği yöntemle kıyaslanması yapılmaktadır. Sunulan yöntemin, basit bir ortamla robotun etkileşimini simüle eden bir deney kümesinde, sensör motor uzayını bölmede, orijinal yöntemden daha iyi bir performans gösterdiği ve bu uzayı semantik olarak daha mantıklı bölgelere ayırdığı görülmüştür.

Anahtar Kelimeler—içsel motivasyon, otonom zihinsel gelişim, pekiştirmeli öğrenme, aktif öğrenme, gelişimsel robotik.

I. GİRİŞ

Geçmiş yıllardan beri, akıllı bir sistem üretme konusunda, bilim adamları genel olarak üç farklı ana akımı takip etmişlerdir. Bunların ilki, akıllı sistemin, verilen görevi önceden tanımlanmış bir prosedürü direkt olarak izlemesi üzerine kurulu bir yaklaşımdır. İkinci yaklaşım, göreve özel bir öğrenme programı üzerinden, akıllı sisteme sensor bilgisinin insan tarafından beslenmesi mantığı ile çalışmaktadır. Üçüncü yaklaşımda ise, akıllı sistemlerin evrimsel prensiplere benzer olarak, "en uygunun hayatta kalması" mantığı üzerine kurulu olduğunu görmekteyiz [1].

İçsel motivasyon, insanların öğrenme ve özümseme eğilimini yansıtan bir yapıdır. Bir canlının hayatta kalması, onun sürekli olarak öğrenmesine ve öğrendiklerini uygulamasına bağlıdır. Bu bağlamda, içsel motivasyon, sürekli ve kaliteli öğrenmenin önemli bir parçasıdır [2]. İçsel motivasyon, edinilmesi gereken yetenek ve bilginin düzeyini veya iyileşme oranını gözlemleyerek, öğrenme sinyalleri oluşturur [3]. Bu olgudan esinlenerek, robotik dünyasındaki bilim adamları, otonom zihin-

Türkiye Robotbilim Konferansı, 2018

sel gelişim konseptini makinelere uyarlamaya çalışmışlardır. Manuel gelişimin aksine, otonom gelişim iki aşamadan oluşmaktadır. İlk aşamada (yapım ve programlama evresi), öğrenen aracının çalışacağı ortama uygun olarak, herhangi bir göreve özgü olmayan ve öğrenen aracının otonom gelişimini kontrol etmeye yarayan gelişimsel bir program oluşturulur. Bu program çalışmaya ilk başladığı an, öğrenen aracının "doğduğu" an olarak kabul edilir ve ikinci aşama başlamış olur. İkinci aşamada (otonom gelişim evresi), öğrenen aracı bulunduğu ortamla etkileşime geçerek, bünyesindeki gelişimsel program sayesinde, ortamı öğrenir ve ortamda kullanabileceği yetenekleri geliştirir. Gelişim evresinin başlarında öğrenilmiş olan yetenekler, gelişimin sonraki evrelerinde öğrenilecek olan yeteneklerin temelini oluşturur [4]. Bu bildirinin özünü oluşturan ve önerdiğimiz metodun temeli olan nokta, her bölgenin sensör-motor uzayının nasıl bölüneceğine karar verilmesidir. Ortamın dinamiklerinin öğrenme uzayına yansması ve bölgelerin belirlenmesi, bölünme anına kadar bu uzaydaki deneyimlerin başarımına bağlı olarak belirlenmelidir. Bu nedenle, olası sensör-motor gruplarının belirlenmesinde, sunmakta olduğumuz yöntemde, bölünme sonrası oluşacak olan sensör-motor gruplardan (bölgelerden), elde edilecek potansiyel başarıma bakmaktayız. Yöntemimiz, bir robotun basit bir ortamla etkileşimini simüle eden bir deney ile test edilmiş ve orijinal çalışmadan [5] daha iyi bir performans göstermiştir.

II. BENZER ÇALIŞMALAR

İçsel motivasyon üzerine yapılan çalışmalarda genel yapı, iki ana modülden oluşmaktadır. Oudeyer ve arkadaşlarının [5] çalışmasındaki sınıflandırmaya göre, bir modül verilen sensör-motor durumdaki aksiyonun sonuçlarını tahmin etmeye yarayan bir öğrenme makinası **M** iken, diğer modül ise **M** modülünün yaptığı hataları tahmin etmeye yarayan bir meta-öğrenme makinası **metaM** şeklinde tanımlanmıştır. Oudeyer, varolan çalışmaları aksiyon seçimi yöntemlerine göre üç farklı gruba ayırmıştır.

- **Hata Maksimizasyonu** : Bu türdeki sistemlerde (ör., [6], [7], [8]), **metaM** modülünün **M** modülü için yürüttüğü hata tahmininin en yüksek olduğu aksiyon seçilir. Bu yöntem öğrenilebilir ve deterministik ortamlarda işe yararmaktayken, gerçek dünyada uygulanabilirliği kısıtlıdır. Bunun nedenlerine örnek olarak, sensörlere gelen bilginin gürültülü olması, öğrenilecek uzayın öğrenme sisteminin yeteneklerine göre

çok karmaşık olması veya öğrenmede kullanılacak parametrelerin gözlemlenemez olması verilebilir.

- **Gelişim Maksimizasyonu** : Bu tür sistemlerde (ör., [9], [10]) **M** ve **metaM**'ye ek olarak üçüncü bir modül olan bilgi kazancı denetçisi (**knowledge gain assessor - KGA**) bulunmaktadır. **KGA**, **M**'nin yakın gelecekteki ortalama hatasını tahmin eder ve en yakın zamanda yapılmış olan ortalama hata oranını kaydeder. Bir sonraki aksiyonun seçimi için, yakın gelecekteki ortalama hata tahmininden, en yakın zamanda yapılmış olan ortalama hata oranı çıkarılır ve ortalama hatada en çok düşüşe neden olacağı tahmin edilen aksiyon seçilir. Bu yöntem öğrenme uzayındaki görünümü benzer olan aktivitelerle çalışan sistemlerde işe yaramaktayken, öğrenme uzayındaki görünümü birbirinden çok farklı olan aktiviteleri kapsayan sistemlerde hatalı sonuçlar verebilmektedir. Bunun nedeni, farklı türden aktivitelerin yarattığı hataların, aktivitelerin türüne bakılmaksızın değerlendirilmesidir.
- **Benzerlik-tabanlı Gelişim Maksimizasyonu** : Gelişim Maksimizasyonu yaklaşımının gerçek dünyada başarılı olabilmesi için, farklı türden aktivitelerin tanınması ve ayırt edilmesi gerekmektedir. Bu tür sistemler (ör., [11], [5]), farklı türden aktivitelerdeki gelişimlerini ayrı ayrı denetleyip, hangisi üzerine ilerleyeceğine daha doğru bir şekilde karar verebilmektedir. Bu yaklaşımı benimsemiş olan herhangi bir akıllı sistem, farklı türden aktivitelerin ayrımını kendi başına yapabilmeli ve aktiviteleri belirli bağlamlarda ilişkilendirebiliyor olmalıdır.

Schmidhuber'in çalışmasında [11] meraklı model oluşturan sistem (curious model building system), öğrenen aracıyı ortamda öğrenebileceği şeyler olduğu beklentisini kazanmış olduğu durumlara doğru iten sistem olarak tanımlanmıştır. Çalışmada sunulan model, pekiştirmeli öğrenme algoritmalarından biri olan, Watkins'in Q-Learning algoritmasına [12] göre geliştirilmiştir. Bu çalışmada merak (curiosity), ortamın reaksiyonlarını tahmin eden sistemin iyileştirilmesi isteği olarak tanımlanmış ve gerçekleşmesi için de pekiştirmeli öğrenme kullanılmıştır. Bu mekanizmada, model adaptif olduğu için, zamanla tahminler iyileşecek ve öğrenen aracı artık iyi tahmin edebildiği alanları keşfetmekten vazgeçip, ortamın henüz tahmin edilmemiş kısımlarına yönelecektir. Bu çalışma, aktif öğrenme prensibi üzerine yapılmış bir çalışmadır ve Benzerlik-tabanlı Gelişim Maksimizasyonu grubuna düşmektedir.

Gelişimsel robotik disiplini içerisine düşen bir diğer çalışmada [13] ise pekiştirmeli öğrenme sistemi iki evreye ayrılmış; bunlardan ilki olan *evrimsel evrede* öğrenmeye yardımcı olacak bileşenlerin içsel motivasyonla geliştirilmesi, ikincisi olan *öğrenme evresinde* ise evrimsel evrede geliştirilmiş bileşenlerin diğer bileşenlerin öğrenmesine rehberlik etmek adına dışsal motivasyondan yararlanmasını düşünülmüştür. Bu çalışmada pekiştirmeli öğrenmeli hiyerarşik yapay sinir ağları kullanılmıştır. İçsel motivasyonun temel alındığı bir diğer çalışma [3], Bilişsel Bilimler ve Bilgisayarlı Görü alanlarıyla bağlantılı olarak gerçekleştirilmiştir. Bu çalışmada içsel motivasyon temelli açık dikkat (overt attention) yeteneklerinin gelişimi

hedeflenmiş ve öğrenen aracının bakışını sahnedeki ilgili yerlere çevirmeyi öğrenmesi üzerine deneyler yapılmış ve yapay sinir ağları kullanılmıştır. Çalışmanın, robotik kolları kontrol eden bir sistemle entegre olması düşünülmektedir. İçsel motivasyonun kullanıldığı çalışmalarda, pekiştirmeli öğrenmenin temel yapıtaşlarından biri olan ödül fonksiyonu seçiminin, öğrenme oranında kritik bir rolü olduğu gözlenmiştir. Bu amaçla, Niekum [14] ödül fonksiyonunun genetik programlama yöntemiyle belirlendiği bir çalışma yapmıştır. Uğur'un [15] çalışmasında ise, objelerin farklı zorluk düzeylerindeki sağlamlık öğrenme görevlerini hiyerarşik bir yapıya oturtmak hedeflenmiştir. Çalışmada, objelerin sağlamlıklarını öğrenmek adına, obje seçim aksiyonu, obje çeşitliliğini arttıracak bir bulguları (heuristic) kullanılarak gerçekleştirilmiştir. Böylece robot, her adımda aday aksiyonlar içerisinde en "ilginç" aksiyonu seçerek keşfine devam etmekte ve öğrenme gelişimini arttırmaktadır. Chan [16] çalışmasında, **AAM**'de dağıtık hesaplama yaklaşımı izleyerek, her sensöre özel öğrenme alanları oluşturmuştur.

Benzer çalışmaların bir kısmı öğrenme uzayını adaptif bölmek yerine el ile önceden bölünmüş uzayın öğrenilmesinde **AAM**'yi kullanmaktadır. Diğer bir kısım [15] ise gelişimsel robotiğin ruhuna daha uygun olarak, robotun deneyimleri sayesinde otonom bölünme mekanizmasını kullanarak **AAM** metodunu uygulamaktadır. İkinci kategoride olan bizim çalışmamızın farkı, sensör-motor uzayın, öğrenme sisteminin başarımına bağlı olarak bölünmesi ve potansiyel hataların hesaplanmasına dayalı olmasıdır.

Oudeyer'in [5] yöntemi öğrenme gelişiminin benzer durumlarla karşılaştırılması üzerine kuruludur. Önerdiğimiz sistem de benzer durumların karşılaştırılması üzerine kurulu bir sistemdir fakat, benzer durumları ayırt etme mekanizması, potansiyel hataların hesaplanması üzerinden yapılır.

III. YÖNTEM

A. Genel Bakış

Akıllı Adaptif Merak (AAM) [5], öğrenme ve problem uzayını adaptif olarak parçalara ayırır ve bu öğrenme uzayını adım adım öğrenirken hangi kısımlarını öncelikle keşfedeceğini ve öğreneceğini belirlemek için öğrenme gelişim hızını kullanır. Öğrenme gelişim hızının yüksek olduğu kısımları öncelikle tarar ve öğrenir. Öğrenilmesi çok kolay ve öğrenilmesi çok zor/imkansız olan problemlerde öğrenme hızı daha düşük olduğundan, çok basit ve çok zor öğrenme problemlerinden, orta basitlikte problemler otomatik olarak daha önce çalışılır ve daha kolay öğrenilir. **AAM**, robotu kendi öğrenme gelişimini arttıracak durumlara yönlendiren bir sistemdir. Bu dürtü, robotun çok kolay tahmin edilebilen ve öngörülemeyen durumlardan kaçınmasını sağlar ve böylelikle otonom zihinsel gelişimini destekler. Robotun aktiviteleri otonom bir şekilde karmaşıklıkla ve kompleks gelişimsel zincir, kendi kendine organize olur.

AAM modelinde akış şu şekildedir: Robotun karşılaştığı her bir deneyim, bir vektör halinde sistem hafızasında tutulur. Sensorimotor uzay robotun karşılaştığı örnek sayısına bağlı olarak sürekli olarak bölgelere ayrılır. Her bölge, kendi alanına düşen örneklerden sorumludur ve kendine ait bir öğrenme makinasına sahiptir. Her öğrenme makinası, bağlı olduğu bölgeye ait olan örneklerle eğitilir ve daha sonra bu bölgeye düşen

bir örnek ile karşılaştığında, sonuç bu öğrenme makinası tarafından tahmin edilir. Örneğin işlenmesinden sonra, gerçek sonuç ile tahmin arasındaki hata hesaplanır ve ilgili bölgenin hata listesine kaydedilir. Daha sonra bu liste, ilgili bölgenin potansiyel öğrenme gelişimini hesaplamada kullanılır. Bölgelerin öğrenme gelişimleri olası aksiyonlar arasından, bir sonraki adımda öğrenmeye en çok katkıda bulunacak aksiyonların belirlenmesinde kullanılır. Bu bildiride önerdiğimiz yöntemin ana fikri, öğrenme uzayının akıllı bir şekilde bölünmesi ile ilgili olup detaylar III-C'de verilmiştir.

B. Format

Her öğrenme makinasının girdisi, robotun o anki sensör $S(t)$ ve motor $M(t)$ parametrelerinin birleşimi olan tek bir vektördür $SM(t)$. Bu girdiye bağlı olarak, öğrenme makinası bir sonraki sensör durumunu $S(t+1)$ tahmin eder, yani çıktı olarak verir.

C. Sensorimotor (SM) Uzayın Bölgelere Ayrılması

Önerdiğimiz yaklaşımda, **AAM** yönteminin öğrenme uzayının tahmin edilebilirlik ölçütü kullanılarak parçalanması yoluyla, öğrenme performansının artırılması amaçlanmıştır. **AAM**'de öğrenme uzayı, özyinelemeli bir biçimde alt bölgelere ayrılmaktadır. Bu bölgelerin ayrılma mantığı, bu çalışmanın temelini oluşturmaktadır. Yöntemimizde, öğrenme uzayı alt bölgelere ayrılmadan önce, bu uzayın her bir özneliği üzerinden, belirli sayılarda farklı yerlerden ikiye bölünmesiyle hesaplanmış olan potansiyel öğrenme yetileri, bölgelerin hangi öznelikten ve bu özneliğin hangi değerinden ayrılacağına yanıtını verir.

AAM'de bölünme prosedürü kısaca şu şekilde özetlenmektedir: Sistemde ilk başta yalnızca bir bölge vardır ve her bölge belli bir sayıda örneği barındırdığında (C kriteri) iki ayrı bölgeye ayrılır. Her bölge özyinelemeli olarak aynı evreden geçer. Her bölge, bölündüğü boyut ve bölünme değerini bünyesinde tutar ve yeni bir tahmin yapılması gerektiğinde, bu sayede ilgili bölge bulunur. SM uzayı bölünürken bölünme SM içerisindeki seçilen bir öznelik (j) kullanılarak yapılır. Örneğin, bölünme için seçilen öznelik robotun motor komutuysa, ve bölünme noktası 0.5 değeryse, yeni oluşan bölgelerden ilkinin örneklerinin hepsinde motor komutunun değeri 0.5'ten küçük, ikincisinin örneklerinin hepsinde aynı komutun değeri 0.5'ten büyük olacaktır.

Makalenin geri kalanında, potansiyel hata oranına bağlı olan yöntemimize **PH-AAM** olarak, kıyaslanacak olan [5] varyansa bağlı olan yönteme ise **V-AAM** olarak değinilecektir.

V-AAM'de, seçilen bir öznelik için, bölünen bölgelerin içindeki örneklerin kendi içerisindeki varyansın minimum, örnekler arasındaki uzaklığın (seçili öznelik ortalamalarının) maksimum olacak şekilde bölünmesi amaçlanmaktadır. Örnek kümesi, ikiye ayrılmış örnek kümeleri içerisindeki her örneğin $S(t+1)$ bileşeninin varyansları toplamının, her bir örnek kümesinin boyutuyla ağırlıklanarak hesaplanması sonucu çıkan değerin minimal tutulması mantığıyla işlemektedir. Basitçe, bölge R_n , SM vektörü içerisindeki her bir öznelik için, j bölünme ölçütü (öznelik indeksi) ve v_j bölünme değeri olmak üzere,

- R_{n+1} 'in her bir örneğinin $(SM(t), S(t+1))_i$, $SM(t)$ bileşeninin j 'inci boyutundaki değerler v_j 'den küçük,

- R_{n+2} 'in her bir örneğinin $(SM(t), S(t+1))_i$, $SM(t)$ bileşeninin j 'inci boyutundaki değerler v_j 'den büyük,
- $|R_{n+1}|var(\{S(t+1)\})|(SM(t), S(t+1)) \in R_{n+1}| + |R_{n+2}|var(\{S(t+1)\})|(SM(t), S(t+1)) \in R_{n+2}|$ değerinin minimal,

olduğu yerden bölünür.

Önermekte olduğumuz **PH-AAM** yönteminde ise, SM vektöründeki her öznelik için farklı bölünme aralıklarıyla oluşturulmuş bölgeler bulunmakta ve her alt bölge çiftinden, potansiyel hata toplamı en düşük olan seçilmektedir. Böylece örneklerdeki öznelik dağılımına göre bölme işlemi yapan **V-AAM** yerine bölme işlemi, oluşması muhtemel olan alt bölgelerdeki öğrenme makinalarının potansiyel başarımları göz önüne alınarak yapılmaktadır. **PH-AAM**, potansiyel bölgelerin hata oranının hesaplanması esasına bağlı olarak çalışmaktadır. Her örnek $SM(t)$, l boyunda bir vektör olmak üzere, bölgenin nereden bölüneceğine karar verilmesi şu akışta gerçekleşir :

- Her ebeveyn (parent) bölgeye ait örnek kümesi, her bir boyut j için , yalnızca j boyutundaki değerleri göz önüne alınarak küçükten büyüğe doğru sıralanır.
- Sıralanmış örnekler, farklı noktalardan ikiye bölünür. Bulunan her iki çocuk (child) bölge, kendi alanına düşen her bir örnek için eğitilir ve yaptığı hata hesaplanır. Bu işlem artmalı (incremental) bir şekilde gerçekleştirilir. Bu iki çocuk bölge hataları toplamı, örnek kümelerinin boyutlarına bölünür ve hesaplanan bu iki değer minimum olanı alınır. j boyutu için farklı bölme noktalarından aynı şekilde hesaplanmış olan değerlerin en küçüğü seçilir ve potansiyel bölme noktası olarak kaydedilir.
- Bütün boyutlar arasından, potansiyel hata değerinin en küçük olduğu boyut ve ilgili değeri seçilir ve ebeveyn bölge, bu boyuttan ve karşılık gelen değerden bölünür.

Yöntemimiz sayesinde, olası bölme pozisyonları arasından en düşük hataya neden olacak olan boyut ve değer seçilmiş olur.

D. Öğrenme Makinaları

Her bölge, kendine ait örnekler ile eğitilmiş birer öğrenme makinasına sahiptir. Öğrenme makinaları, bağlı olduğu bölgenin kapsadığı bir örnek ile karşılaştığında, bu örneğin sonucunu tahmin etme sorumluluğuna sahiptir. Bu öğrenme makinası için herhangi bir makina öğrenmesi algoritması kullanılabilir. Ancak, sistemin bütünlüğü açısından, tüm bölgeler aynı algoritmayı kullanmalıdır. Bu bildiride kullanılan algoritmanın ne olduğu, öğrenme probleminin ihtiyaçlarını karşılayabildiği sürece önemli değildir ve yöntemden bağımsızdır. Çalışmamızda, **K-En Yakın Komşu (K-EYK)** algoritması kullanılmaktadır. Herhangi bir bölge ikiye bölündüğünde çocuk bölgeler, ebeveyn bölgenin öğrenme makinasını direkt olarak kullanamaz. Bu nedenle, her bölünmeden sonra, oluşan çocuk bölgeler, kendi öğrenme makinalarını kendine ait örneklerle eğitmek zorundadır.

Sunmakta olduğumuz **PH-AMM** yönteminde, her yeni öğrenme makinası, kendi alanına düşen her bir örneği ile teker teker eğitilir ve bu şekilde kendi hata listesini oluşturur. Böylece, her çocuk bölge, **kendi alanına düşen örnekler üzerinde kendi yaptığı hataları dikkate almış** olur. Ancak,

V-AMM yönteminde her çocuk bölge ebeveyn bölgenin tüm hatalarını devralacak şekilde tasarlanmıştır.

E. Öğrenme Gelişiminin Hesaplanması

Her bölgenin öğrenme gelişimi, kendi hata listesi üzerinden yapılır. Herhangi bir örnek için yapılan tahminin $S'(t)$ hatası $e_n(t)$, tahminin gerçek sonucundan $S(t)$ farkının karesi alınarak hesaplanır.

$$e_n(t) = ||S(t) - S'(t)||^2$$

Bir bölgenin R_n hata listesi E_n , herhangi bir t anında şu şekilde ifade edilmektedir:

$$e_n(t - \phi), e_n(t - \phi + \omega_1), e_n(t - \phi + \omega_2), \dots, e_n(t)$$

Burada $e_n(t - \phi)$, her çocuk bölgenin, ebeveyn bölgeden devraldığı, kendi alanına düşen ilk örneğin hatasını ifade etmektedir. İlgili hatalar, belirli aralıklarla gelmiyor olduğundan $e_n(t - \phi + \omega_1)$ ifadesi, çocuk bölgenin alanına düşen bir sonraki örneğin hatasını ifade eder. $e_n(t)$ ise çocuk bölge tarafından yapılmış en güncel tahminin hatasıdır.

Bir bölgenin öğrenme gelişiminin hızı, hata listesinde yakın zamanda yapılmış olan hata eğrisinin yumuşatılmış türevi ve daha önceki bir zamanda yapılmış olan hata eğrisinin yumuşatılmış türevi kullanılarak hesaplanır. θ yumuşatma parametresi ve τ zaman penceresi parametresi olmak üzere, matematiksel olarak,

$$\langle e_n(t) \rangle = \frac{\sum_{i=0}^{\theta} e_n(t-i)}{\theta}$$

$$\langle e_n(t - \tau) \rangle = \frac{\sum_{i=0}^{\theta} e_n(t-\tau-i)}{\theta}$$

$\langle e_n(t) \rangle$ yakın zamanda yapılmış olan hata seti eğrisinin yumuşatılmış türevini, $\langle e_n(t - \tau) \rangle$ daha önceki bir zamanda yapılmış olan hata seti eğrisinin yumuşatılmış türevini ifade eder.

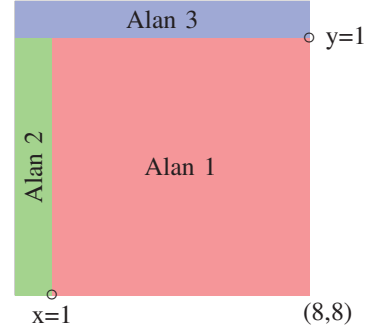
Tahminlerin ortalama hata oranındaki asıl düşüş $D(t)$ ve asıl öğrenme gelişimi $L(t)$ ise şu şekilde ifade edilir:

$$D(t) = \langle e_n(t) \rangle - \langle e_n(t - \tau) \rangle$$

$$L(t) = -D(t)$$

F. Aksiyon Seçimi

İçsel motivasyon sistemine sahip sistemlerde, aksiyon seçimi, elde edilen içsel ödülün azami düzeye çıkarılması esasına dayalı olarak yapılır. Bu nedenle, içsel motivasyon sistemi genellikle pekiştirmeli öğrenme metoduyla birlikte ele alınmaktadır. Bizim yöntemimizde, sensör-motor uzay sürekli olduğu için, bir sonraki sensör-motor girdi aday $SM(t+1)$, ilgili değer aralıkları arasından rastgele örnekleme yöntemi ile seçilir. 100 örneklik sette, her örneğin denk düşüğü bölge bulunur ve bu bölgenin öğrenme gelişimleri karşılaştırılır. ϵ -açgözlü (greedy) aksiyon seçimi ile, en yüksek öğrenme gelişimine sahip olan bölgeye düşen örnek, bir sonraki sensör-motor girdi olarak kullanılır. Seçilen aksiyon uygulandıktan ve sonuçları gözlemlendikten sonra sistem güncellenir. **PH-AAM** ve **V-AAM**'de öğrenme gelişiminin hesaplanması ve aksiyon seçimi aynı şekilde yapılır.



Şekil 1. Deney ortamı

IV. DENEY ORTAMI

Deney düzeneğinde 8×8 birimlik 2 boyutlu bir ortam üzerinde hareket eden bir robot simüle edilmiştir. Deney alanı, üç farklı özelliğe sahip alanların birleşiminden oluşmaktadır. Robotun ortamlarla etkileşimi, içinde bulunduğu alana ve robotun yaydığı ses frekansına göre farklı sonuçlar doğurmaktadır. Robot, yatay ve dikey yönlerde hareket eder ve bu hareketinin hız komutları -1 ve 1 arasında değişen gerçek sayılardır. Ayrıca, robot hareketi esnasında 0 ve 1 aralığında değişen frekanslarda ses çıkarmaktadır ve bu ses frekansları, ortamlarla etkileşim üzerinde farklı etkilere sahiptir. Ses frekansı $f1 = [0; 0.33]$ olduğunda, hız komutlarının tersi işleme konur, (örneğin; $h = 0.1, v = 0.4$ olan komutlar $h = -0.1, v = -0.4$ şeklinde işleme konur), $f2 = [0.33; 0.66]$ aralığında hız komutları ne olursa olsun $h = 0, v = 0$ şeklinde işleme konur, $f3 = [0.66; 1]$ aralığında ise hız komutları oldukları şekilde işleme konur. Ses frekansının hız komutlarına etkisi değerlendirildikten sonra, eğer robot Alan-1'de ise, değerlendirilmiş olan hız komutlarını 3 ile çarparak kullanır; Alan-2'de ise değerlendirilmiş olan hız komutlarını 2'ye bölerek kullanır; Alan-3'te ise tamamen rastgele yerlere hareket eder. Robotun pozisyonundaki değişiklik, var olduğu x, y pozisyonunun üzerine yatay ve dikey h, v hız komutları ayrı ayrı eklenerek hesaplanır. Bu deneyde motor vektörü $M(t)$, yatay hız h , dikey hız v ve ses frekansı f 'ten oluşur: $M(t) = (h, v, f)$

Algı vektörü $S(t)$ ise, robotun alandaki x, y pozisyonundan oluşur: $S(t) = (x, y)$. Özetle, robot güncel sensör-motor girdi üzerinden, bir sonraki sensör durumunu tahmin etmeye çalışır.

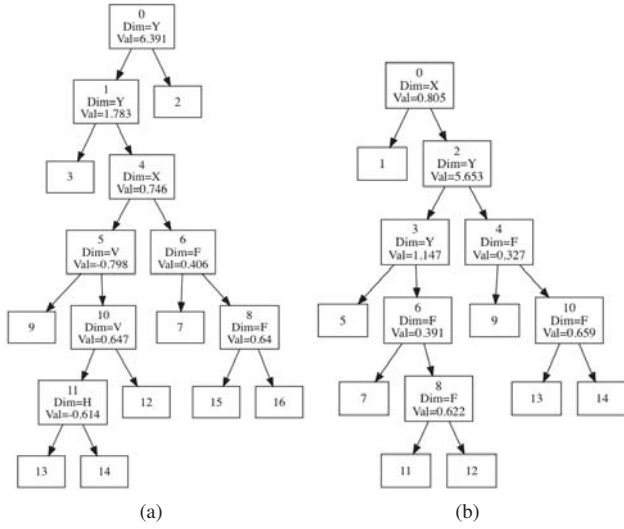
$$SM(t) = (h, v, f, x, y) \mapsto (x', y') = S(t+1)$$

A. Öğrenme Akışı

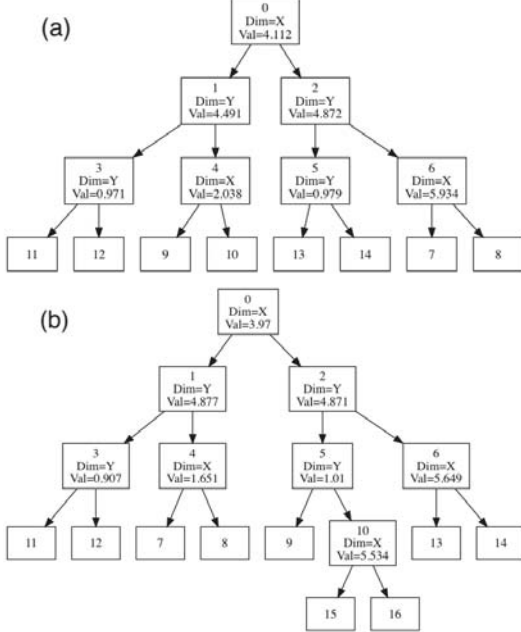
Herhangi bir t anında, sistem tarafından rastgele bir şekilde N tane olası SM vektörü üretilir ve aksiyon, ilk bölünme öncesi ve ϵ -açgözlü (greedy) aksiyon seçimi ile, sistemin bünyesindeki bölge ağacının her yaprak bölgesinin (leaf region) öğrenme gelişimi hesaplanarak seçilir. (N ortamın dinamiklerine bağlı olarak deneylerle gözlemlenmiş bir sayıdır ve bu deneyde $N = 100$ olarak alınmıştır.) Daha sonra robot, bu vektöre bağlı olarak bir sonraki sensör durumunu tahmin etmeye çalışır. Bunu yaparken, SM vektörünün düştüğü bölgenin öğrenme makinası kullanılır ve SM vektörü ilgili bölgenin örnekleri arasına kaydedilir.

B. Deney Parametreleri

Yaptığımız deneylerde sistem, 5000 iterasyon yapılarak eğitilmiş olup, bölünme sayısı kriteri $C = 1000$ olarak alınmıştır.



Şekil 2. PH-AAM tarafından oluşturulmuş sensör-motor bölge ağaçları



Şekil 3. V-AAM tarafından oluşturulmuş sensör-motor bölge ağaçları

Yani herhangi bir bölge, 1000 adet örnek barındırdıktan sonra iki ayrı bölgeye ayrılmaktadır. ϵ -açgözlü (greedy) aksiyon seçimi için $\epsilon = 0.3$ olarak alınmış, öğrenme gelişimi hesaplamada kullanılan yumuşatma parametresi $\theta = 30$ ve zaman penceresi parametresi $\tau = 5$ olarak kullanılmıştır. Ayrıca, bölünme öncesi olası bölünme yerlerinin tespit edilmesinde, her bölge 10 farklı yerden ikiye bölünerek, en doğru bölünme boyutu ve değeri hesaplanmıştır. Çıkan sonuçlar, V-AAM yöntemiyle aynı parametreler kullanılarak karşılaştırılmıştır.

V. SONUÇLAR

Sonuçların karşılaştırılmasında incelediğimiz kriterler, bölünme noktalarının ortamın dinamiklerine göre anlamlı olup olmadığı (V-A), bölgelerin öğrenme hızı (V-B) ve toplam hatadaki düşüş (V-C) olmak üzere üç kısımda incelenecektir.

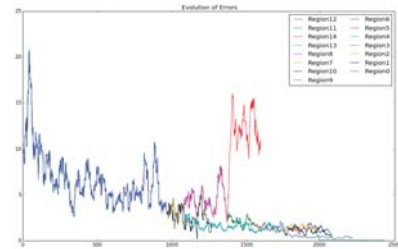
A. Bölünme Noktalarının Anlamlılığının Karşılaştırılması

Yöntemimiz, tahmin için sensör-motor uzayını anlamlı yerlerden bölerek, benzer aksiyonları gruplamayı ve böylece tahmin tutarlılığını yükseltmeyi hedeflemektedir. Bunun kıyaslanması için PH-AAM ve V-AAM yöntemleri 5000 iterasyonluk bir eğitim sürecinden geçirilmiştir. Bu alanların oluşturulmasında birçok çalıştırmanın sonucunda, rastgelelikten kaynaklı olarak farklı yapıda bölünme ağaçları elde edilmiştir. Yer kısıtlılığından dolayı, bunlar arasından temsilci olanları makalede gösterilmektedir. Ortaya çıkan bölünme ağaçları Şekil 2 ve 3'te gösterilmektedir. Şekil 2 ve 3'te düğümler (node) üzerindeki ilk satır **bölgenin numarasını**, ikinci satır **bölündüğü boyutu** (H, V, F, X, Y sırasıyla SM vektöründeki h, v, f, x, y boyutlarını ifade etmektedir) ve üçüncü satır ise **boyutun bölündüğü değeri** ifade etmektedir.

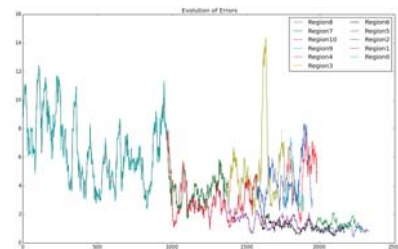
Şekil 2'de PH-AAM yöntemi ile aynı parametrelerle oluşturulmuş farklı deney sonuçları gösterilmektedir. Şekil 2-a'da, 4. bölgenin bölünme yeri, ortamın dinamiklerine uygun bir şekilde hesaplanmış, 6. bölge tamamıyla Alan-1'i kapsamakta olup, bundan sonraki bölünmeler F değerlerinden gerçekleştirilerek, yöntemimiz F 'in yarattığı değişimin farkında olan bir tahminde bulunmuştur. Şekil 3'te ise V-AAM yöntemiyle ortaya çıkan bölge ağaçları gösterilmektedir. Bu yöntem ortamın dinamiklerini, öğrenme bölgelerine yansıtma yeterince başarılı olamamıştır. Örneğin, deneylerin hiçbirinde F parametresine bağlı bir bölünme gözlenmemektedir.

B. Bölgelerin Öğrenme Hızının Karşılaştırılması

5000 iterasyonla eğitilmiş PH-AAM'nin oluşturduğu bölge ağacı Şekil 2'de gösterilmiştir. Bu ağaçta, ilk bölgenin (0. bölge) bölünmesi $X = 0.805$ 'ten gerçekleşmiştir. Bu durumda ağacın yapısına bakıldığında sistemin, X 'in belirtilen değerinden büyük yerleri keşfetmenin, öğrenme gelişimi açısından daha uygun olduğuna karar verdiğini görüyoruz.

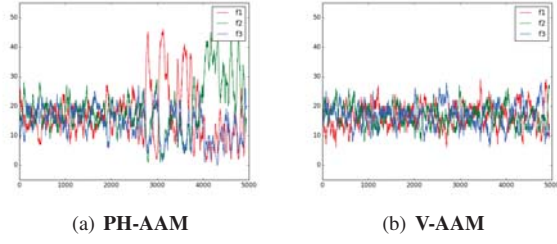


(a) PH-AAM



(b) V-AAM

Şekil 4. Yumuşatılmış hata eğrisi değişimleri



Şekil 5. Robot tarafından tercih edilen ses frekansı değişimi

Daha sonra 3. bölgenin $Y = 1.147$ 'den bölünmesi ve çocuk bölgelerinden biri olan 5. bölgede ilerleme kaydedilmemesi, o bölgenin öğrenilemez olmasından kaynaklanmaktadır. 2. bölgenin her iki kolunda da görülebileceği üzere, sistem F parametresini, fark yaratan noktalardan bölmeyi başarmıştır.

PH-AAM yönteminden oluşturulan ağaç yapısının yumuşatılmış hata eğrileri Şekil 4-a'da gösterilmektedir. Şekil 4-a'da her bölge, kendi ebeveyn bölgesinin devamı olacak şekilde gösterilmiştir, böylece bölünme sonrası hatadaki azalma gözlenebilmektedir. Bu grafikten de görüleceği üzere, bölge 5'in hata oranı, tamamen rastgele olan Alan-3'e denk geldiği için eşzamanlı diğer bölgelere kıyasla daha yüksek olmaktadır. Sistem alt bölgelere ayrıldıkça, tahmindeki tutarlılık oranı artmaktadır. Aynı şekilde, **V-AAM** yönteminin yumuşatılmış hata eğrisi Şekil 4-b'de gösterilmektedir. Bu yöntemde, hatalar ebeveyn bölgeden tamamen devralınmaktadır ve hatanın buna bağlı olarak değişimi gözlenmektedir. İki grafikte hata eğrilerinin ulaştığı son değerler kıyaslandığında, yöntemimizin diğer yöntemlere göre daha düşük hataya sahip tahminler yapabildiği gözlenmektedir.

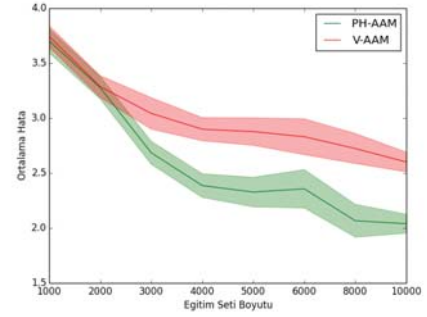
İki yöntemin deney boyunca ilk bölünme öncesi ve ϵ -açgözlü (greedy) aksiyon seçimi ile, kullanmayı tercih ettiği ses frekanslarının değeri Şekil 5'te gösterilmiştir. Tercih edilen frekansları karşılaştırdığımızda, yöntemimizin sensoryomotor bölge ağacında da görüldüğü gibi belirli zaman aralıklarında belirli ses frekanslarını kullanmayı tercih ettiğini görüyoruz. Ancak diğer yöntemde böyle bir ayırım 5000 iterasyonda keşfedilememiş durumdadır. Ayrıca, 10000 iterasyonluk eğitim evresi sonrasında da bu parametrenin keşfedilmediği gözlenmektedir.

C. Toplam Hatadaki Düşüşün Karşılaştırılması

İki yöntem için, farklı sayıda örneklerle eğitilmiş sistemlerin ortalama hata grafiği Şekil 6'da gösterilmiştir. Bu grafik, sistem belirtilen sayıda örneklerle eğitildikten sonra, her birinde 2000 tane örnek olan 10 farklı test setindeki ortalama hata hesaplanarak oluşturulmuştur. Başlangıçta 1000'lik eğitim setinde iki sistemin de performans açısından farkı yoktur, çünkü henüz ilk bölünme yaşanmamıştır ve sistem henüz kendi içsel motivasyonunu kullanarak bir aksiyon seçiminde bulunmamıştır. 2000'lik eğitim setinden sonra, iki sistem arasındaki performans farkı belirginleşmektedir.

VI. SONUÇ

Bu çalışmada, öğrenme uzayının bölgelere ayrılması ve böylece öğrenme gelişiminin iyileştirilmesi amacıyla özgün bir yöntem sunulmuş ve varolan bir başka yöntem ile karşılaştırılması yapılmıştır. Basit bir deney düzeneğinde, önerilen metodun, varolan metoda göre daha iyi bir sonuç verdiği gözlen-



Şekil 6. **PH-AAM** ve **V-AAM** Karşılaştırılması

miştir. Yöntemin geçerliliğinin daha karmaşık robotik problemlerde test edilmesi ve geliştirilmesi planlanmaktadır.

KAYNAKÇA

- [1] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen, "Autonomous mental development by robots and animals," *Science*, vol. 291, no. 5504, pp. 599–600, 2001.
- [2] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions," *Contemporary educational psychology*, vol. 25, no. 1, pp. 54–67, 2000.
- [3] V. Sperati and G. Baldassarre, "Learning where to look with movement-based intrinsic motivations: A bio-inspired model," in *Development and Learning (ICDL-Epirob), 2014 Joint IEEE International Conferences on*. IEEE, 2014, pp. 461–468.
- [4] J. Weng, "A theory for mentally developing robots," in *Development and Learning, 2002. Proceedings. The 2nd International Conference on*. IEEE, 2002, pp. 131–140.
- [5] P.-Y. Oudeyer, F. Kaplan, and V. V. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE transactions on evolutionary computation*, vol. 11, no. 2, pp. 265–286, 2007.
- [6] X. Huang and J. Weng, "Novelty and reinforcement learning in the value system of developmental robots," 2002.
- [7] J. Marshall, D. Blank, and L. Meeden, "An emergent framework for self-motivation in developmental robotics," 2004.
- [8] A. G. Barto, S. Singh, and N. Chentanez, "Intrinsically motivated learning of hierarchical collections of skills," in *Proceedings of the 3rd International Conference on Development and Learning*. Citeseer, 2004, pp. 112–19.
- [9] J. M. Herrmann, K. Pawelzik, and T. Geisel, "Learning predictive representations," *Neurocomputing*, vol. 32, pp. 785–791, 2000.
- [10] F. Kaplan and P.-Y. Oudeyer, "Motivational principles for visual know-how development," 2003.
- [11] J. Schmidhuber, "Curious model-building control systems," in *Neural Networks, 1991. 1991 IEEE International Joint Conference on*. IEEE, 1991, pp. 1458–1463.
- [12] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.
- [13] M. Schembri, M. Mirolli, and G. Baldassarre, "Evolution and learning in an intrinsically motivated reinforcement learning robot," *Advances in Artificial Life*, pp. 294–303, 2007.
- [14] S. Niekum, A. G. Barto, and L. Spector, "Genetic programming for reward function search," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 2, pp. 83–90, 2010.
- [15] E. Ugur and J. Piater, "Emergent structuring of interdependent affordance learning tasks using intrinsic motivation and empirical feature selection," *IEEE Transactions on Cognitive and Developmental Systems*, 2017.
- [16] M. T. Chan, R. Gorbet, P. Beesley, and D. Kulić, "Curiosity-based learning algorithm for distributed interactive sculptural systems," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 3435–3441.