

## Makine Çizelgeleme Problemlerinin Çözümünde Pekiştirmeli Öğrenme Etkisinin Analizi

Orhan Engin<sup>1\*</sup>, Bünyamin Sarıcan<sup>2</sup>

<sup>1</sup>Endüstri Mühendisliği Bölümü, Mühendislik Fakültesi, Konya Teknik Üniversitesi, Konya, Türkiye.

<sup>2</sup>Endüstri Mühendisliği Bölümü, Lisansüstü Eğitim Enstitüsü, Konya Teknik Üniversitesi, Konya, Türkiye.

\*oengin@ktun.edu.tr

### Özet

Pekiştirmeli öğrenme, günümüz dünyasında birçok gerçek hayat problemine çözüm bulmak için aktif bir şekilde kullanılmakta ve endüstri içerisinde umut verici yöntemler arasında gösterilmektedir. Bu çalışmada, makine öğrenmesinin bir alt dalı olan pekiştirmeli öğrenmenin iş çizelgeleme problemlerinin çözümündeki etkisi araştırılmıştır. Bu kapsamda, öncelikle pekiştirmeli öğrenmede durum tanımı, eylem seçimi ve öğrenme algoritmaları açıklanmıştır. Ardından, iş çizelgeleme probleminin sınıflandırmasına yer verilmiştir. Literatürde yer alan iş çizelgelemede, pekiştirmeli öğrenme yönteminin kullanıldığı, son yirmi yılda yayımlanan, 50 makale çalışmasına yer verilmiştir. Literatürde yer alan çalışmaların çizelgeleme problemlerinin çözümü üzerinde gösterdiği etki değerlendirilmiştir. Son bölümde, pekiştirmeli öğrenmenin diğer çözüm yöntemlerine kıyasla güçlü ve zayıf yönlerine yer verilmiştir. Pekiştirmeli öğrenme algoritmaları, dinamik çevreden, önceden gelen, bilgileri kullanarak öğrenme gerçekleştirdikleri için üretim sürecinde meydana gelecek ani değişikliklere (makine arızası, işlerin rassal gelişi, sipariş iptali vd.) kolayca tepki vererek dinamik bir çizelgeleme yaklaşımı sunar. Bu çalışmanın sonunda, gelecekte yapılacak araştırmalar ile ilgili öneriler sunulmuştur.

**Anahtar Kelimeler:** Makine çizelgeleme optimizasyonu, Makine öğrenmesi, İş çizelgeleme, Pekiştirmeli öğrenme

## Analysis of Reinforcement Learning Effect in Solving Machine Scheduling Problems

### Abstract

Reinforcement learning is actively used to find solutions to many real-life problems in today's world and is shown among the promising methods in the industry. This study investigated the effect of reinforcement learning, which is a sub-branch of machine learning, in solving job scheduling problems. In this context, first, situation definition, action selection, and learning algorithms in reinforcement learning are explained. Then, the classification of the job scheduling problem is given. In the literature, 50 articles published in the last twenty years, in which the reinforcement learning method is used in job scheduling, are included. The effects of the studies in the literature on the solution of scheduling problems were evaluated. In the last section, the strengths and weaknesses of reinforcement learning compared to other solution methods are included. Reinforcement learning algorithms offer a dynamic scheduling approach by easily reacting to sudden changes that may occur in the production process (machine failure, random arrival of jobs, order cancellation, etc.) because they learn by using previous information from the dynamic environment. At the end of this study, suggestions for future research are presented.

**Keywords:** Machine scheduling optimization, Machine learning, Job scheduling, Reinforcement learning.

## 1. GİRİŞ

Üretim çizelgeleme, sınırlı üretim kaynaklarını zaman içinde paralel ve sıralı üretim faaliyetleri arasında tahsis eden bir optimizasyon süreci olarak tanımlanır. Bu tahsis, faaliyetler arasındaki zamansal ilişkileri ve bir dizi paylaşılan kaynağın kapasite sınırlamalarını yansıtan kısıtlara uymalıdır. Deterministik çizelgeleme probleminde, iş sayısı, makine sayısı, iş başına işlem sayısı, makinelerdeki işlem süreleri, öncelik kısıtlamaları ve problem kısıtlamaları gibi çizelgeleme probleminin türüne bağlı olarak değişebilen tüm parametrelerin önceden bilindiği varsayılır. Çözüm bulunmak istenen çizelgeleme problemi stokastik ise, mevcut bilgiler kesin değildir çünkü farklı işlere ait operasyonların işlem süreleri rassal değişkenler olarak modellenir. Bu, bütün işlerin işlem süresinin, çizelgeleme süresince tam olarak bilinmediği anlamına gelir. İşlem süreleri, teslim tarihleri ve diğer çizelgeleme ile ilgili parametrelerin tam olarak bilinmediği durumlarda, son yıllarda yoğun olarak kullanılan bir diğer yöntemde bulanık çizelgeleme modelleridir. Bulanık Çizelgeleme problemlerinde, işlem süreleri, işlerin teslim tarihleri ve diğer çizelgeleme problemi ile ilgili parametreler bulanık sayılar ile ifade edilir [1, 2].

Çizelgeleme problemlerinin çözümü, üretim sistemlerinde ve endüstriyel süreçlerde önemli olduğu için verimli çizelgeleme teknolojilerinin geliştirilmesi son elli yılda üzerinde en çok çalışılan konulardan biri olmuştur [3]. Çizelgeleme problemlerinin çözümünde farklı yöneylem araştırması teknikleri (doğrusal programlama, karışık tamsayı programlama vb.) kullanılır. Bu yaklaşımlar genellikle bir amaç fonksiyonu, bir dizi değişken ve bir dizi kısıtlama içeren bir modelin tanımını içerir. Yöneylem araştırması tabanlı teknikler, iyi tanımlanmış problemler için en uygun çözümleri elde etme yeteneğini göstermiştir, ancak yöneylem araştırması çözümleri statik modellerle sınırlıdır. Öte yandan yapay zekâ yaklaşımları, gerçek dünya problemlerinin daha esnek temsillerini sağlayarak insan uzmanlığının döngüde yer almasına izin verir. Çizelgeleme konusundaki araştırmaların büyük çoğunluğu, tam bilgi ve deterministik bir ortam varsayarak bir temel çizelge oluşturmak için kesin ve yeterli olmayan prosedürlerin geliştirilmesine odaklanmıştır. Ancak, gerçek dünya o kadar istikrarlı değildir, projeler yürütülürken beklenmedik olaylara maruz kalınabilir ve bu durum çok sayıda program kesintisine yol açabilir. Örneğin, kaynaklar kullanılamaz hale gelebilir (arızalar veya planlanmış bakımlar), yeni siparişler gelebilir, operasyonlar zarar görebilir, işler beklenenden daha uzun sürebilir vb. Bu tür durumlarla başa çıkmak için kullanılan bir yaklaşımda, sağlamlığın bir algoritmanın belirsizlikler altındaki performansını ifade ettiği, sağlam çizelgeler oluşturmaktır.

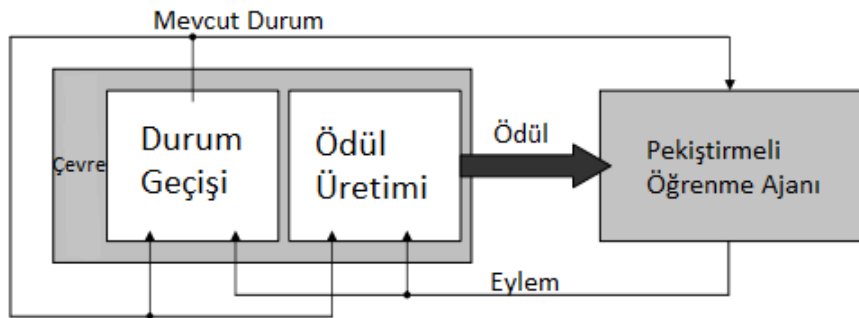
Pekiştirmeli Öğrenme (PÖ), sayısal bir ödül sinyalini maksimize etmek için ne yapılacağını (durumları eylemlerle nasıl eşleştireceğini) öğrenmektir. Bir ajanın, ortamıyla deneme yanılma etkileşimleri yoluyla en uygun davranışı öğrenmesini sağlar. Ajan, farklı durumlarda eylemleri tekrar tekrar deneyerek eylemlerinin sonuçlarını keşfedebilir ve her durum için en iyi eylemi belirleyebilir. Örneğin, beklenmedik olaylarla uğraşırken, önceki sonuçlardan 'öğrenebilecekleri' ve sonraki yinelemeler için belirli parametreleri değiştirebilecekleri için öğrenme yöntemleri önemli bir rol oynayabilir. Endüstrinin ihtiyaç duyduğu sistem, üretim tesisindeki koşullara tam olarak uyum sağlayan ve çok kısa sürede iyi çözümler üreten, optimize edilmiş çizelgelerdir. Endüstri 4.0 ile akıllı fabrikalarda optimize edilmiş çizelgeleme problemleri son yıllarda geliştirilmeye başlanmıştır [4]. Bu tip çizelgelerde kullanılan yöntemlerden biri de öğrenme tabanlı algoritmalarıdır. Bu algoritmalar, dinamik çizelge oluşturmaya imkân verir. Pekiştirmeli öğrenme yöntemlerinin son yıllarda iş çizelgeleme problemlerine aktif olarak uygulandığı görülmektedir. Bu çalışmada, literatürde pekiştirmeli öğrenme yöntemlerinin çizelgeleme problemlerinin çözümü üzerindeki etkisi araştırılmıştır. Bu amaçla “scholar.google.com” arama motoru yardımıyla “Reinforcement Learning and scheduling” anahtar kelimeleri kullanılarak, 2003 ile 2023 yılları aralığında tarama yapılmış, elde edilen çalışmalar değerlendirilmiştir. Bu çalışmalardan; tek makine, paralel makine, akış tipi, atölye tipi ve diğer makine çizelgeleme ile doğrudan ilgili 50 adet makale tespit edilmiş ve çalışmalar detaylı olarak incelenmiştir. Bu araştırma, pekiştirmeli öğrenme yöntemlerinin tek makine, paralel makine, akış tipi, atölye tipi ve diğer makine çizelgeleme problemlerinin çözümünde kullanımı ile ilgili literatürde yapılan detaylı bir çalışmadır.

Bu çalışma dört bölümden oluşmaktadır. İkinci, "Pekiştirmeli öğrenme" bölümü, pekiştirmeli öğrenme yöntemi, öğrenme algoritmaları, durum tanımı ve eylem seçimini açıklar. Üçüncü, "İş çizelgeleme" bölümü, iş çizelgelemenin temel yönlerini açıklar. Literatür araştırması olan dördüncü bölümünde, literatürde yer alan çalışmalara dört başlık altında sınıflandırılarak yer verilmiştir. Beşinci bölüm olan sonuç kısmında ise analiz ve değerlendirmelerde bulunulmuştur.

## 2. PEKİŞTİRMELİ ÖĞRENME

Pekiştirmeli öğrenme, makinenin amaç doğrultusunda ne yapması gerektiğini öğrenen bir makine öğrenme tekniğidir. "Pekiştirmeli öğrenme yaklaşımı, denetimli ve denetimsiz öğrenme yöntemlerinden farklılık gösterir. Denetimli öğrenme, etiketli verilerden eğitim ve test kümeleri oluşturularak bir modelin oluşturulması ve performansının test kümeleriyle değerlendirilmesi üzerine kuruludur. Bu sayede model, etiketsiz verileri tahmin etmek için kullanılabilir hale gelir. Öte yandan, denetimsiz öğrenme, genellikle etiketlenmemiş veri kümelerinde, veri gruplandırma üzerinde yoğunlaşır. Pekiştirmeli öğrenmede ise, bir öğrenen makine olan ajanlar karşılaştığı durumlara tepki verir ve buna karşılık sayısal bir ödül alır. Pekiştirmeli öğrenmenin en belirgin özelliği olan deneme yanılma yöntemiyle ajan, aldığı ödülü maksimize etmeyi hedefler [5].

Pekiştirmeli öğrenme sürecindeki temel zorluklardan biri, keşif ve sömürü dengesini sağlamaktır. Ajan, daha önce pozitif ödül aldığı eylemleri seçerek daha fazla ödül kazanmayı hedefler. Ancak, yeni durumlarda daha yüksek ödül kazanabilecek eylemleri de keşfetmelidir. Ajan, böylece farklı eylemler deneyerek optimum sonuçları elde etmeyi kademeli olarak öğrenir. Pekiştirmeli öğrenme ajanları, belirlenen hedeflere ulaşabilir, çevrelerini algılayabilir ve etkileyebilecek eylemleri seçebilir. Ajandan kastedilen, robot gibi bir varlık veya bir organizma değildir. Ajan denilince eylemleri gerçekleştiren ve öğrenen faktör akla gelmelidir. Aşağıda yer alan Şekil 1'de pekiştirmeli öğrenmeye yönelik şematik gösterim yer almaktadır. Pekiştirmeli öğrenme sistemlerinde, ajan ve çevrenin yanı sıra dört unsurdan oluşan bir yapı bulunur: politika, ödül, değer fonksiyonu ve çevre modeli.



Şekil 1. Pekiştirmeli öğrenmeye ilişkin şematik görünüm [6].

Politika, ajanın mevcut durumda alabileceği aksiyonu belirleyen bir yönerge kümesidir. Bu, etki-tepki ilişkisini ifade eder. Durum, bir etki olarak düşünüldüğünde, ajan buna tepki olarak bir aksiyon verir. Politika, basit bir aksiyon veya tüm durumları kapsayan bir arama tablosu olarak tanımlanabilir. Aynı zamanda politika, ajanın durumu değerlendirerek alabileceği aksiyonları dinamik olarak aramasından kaynaklanan bir niteliğe sahiptir [7]. Ödül, ajanın gerçekleştirdiği bir aksiyon sonucunda ortamdan aldığı sayısal bir puandır. Pekiştirmeli öğrenme ajanı, uzun vadede aldığı ödülleri maksimize etmeyi amaçlamaktadır. Ajanın politikası, edinilen ödülleri çerçevesinde süreç içinde değişir. Mesela, düşük bir puan elde edilen bir aksiyon sonrasında ajan aynı durumla karşılaştığında başka bir eylem seçebilir.

Durum değeri, ajanın mevcut durumdan başlayarak gelecekteki beklenen ödüllerin toplamını temsil eder. Durum değeri, uzun vadede, neyin iyi veya kötü olduğunu gösterirken ödülleri anlık olarak aksiyonların iyi veya kötü olduğunu ifade eder. İsteğe bağlı olarak sisteme entegre edilen bir unsur modeldir. Ortamın bir

simülasyonunu oluşturur ve ajanın aksiyon almadan elde edeceği ödülü ve oluşacak durumu tahmin etmesine olanak sağlar. Bu sayede planlama yapılır ve ajanın davranışında değişiklikler olabilir [7].

## 2.1 Öğrenme Algoritmaları

Ödül temsili, durum temsili, değer güncelleme prosedürü, politika uyarlama gibi konulara göre kullanılan algoritmalar, birbirinden farklılık gösterir. Aşağıda, pekiştirmeli öğrenme için literatürde sıklıkla kullanılan algoritmalar açıklanmıştır.

### 2.1.1 Q-Öğrenme

$Q$ -Öğrenme, modelden bağımsız bir pekiştirmeli öğrenme algoritmasıdır. Belirli bir durumdaki bir eylemin değerini öğrenmek için kullanılır. Çevre modeline ihtiyaç duymaz ve stokastik geçişler ve ödüllerle ilgili problemleri çözebilir ("modelsiz" olarak adlandırılır).  $Q$ -öğrenme, herhangi bir Sonlu Markov Karar Süreci (SMKS) için en uygun politikayı bulmak amacıyla, mevcut durumdan başlayarak, birbirini izleyen adımlarda, toplam ödülün beklenen değerini maksimize etmeye çalışır. Sınırsız keşif süresi ve kısmen rasgele bir politika kullanıldığında, herhangi bir SMKS için en uygun, eylem seçim politikasını belirleyebilir. " $Q$ " ise algoritmanın belirli bir durumda gerçekleştirilen bir eylem için beklenen ödülleri hesapladığı fonksiyonu temsil eder. Geleceğe  $\Delta t$  adım attıktan sonra ajan bir sonraki adımı seçer. Bu adımın ağırlığı  $\gamma^{\Delta t}$  şeklinde hesaplanır. Burada  $\gamma$  (Gamma) indirim faktörü olarak adlandırılır ve  $[0,1]$  arasında bir değer alır.  $\gamma$ , daha önce alınan ödüllere daha sonra alınanlardan daha yüksek bir ağırlık verme etkisine sahiptir, böylece "iyi bir başlangıcın" değerini yansıtır.  $\gamma$ , her  $\Delta t$  adımda başarılı olma olasılığı olarak da yorumlanabilir. Bu nedenle algoritma, bir durum-eylem kombinasyonunun kalitesini hesaplayan bir fonksiyona sahiptir.  $Q: S \times A \rightarrow \mathbb{R}$

Öğrenme başlamadan önce  $Q$ , muhtemel keyfi bir sabit değerle (programcı tarafından seçilen) başlatılır. Ardından, ajan her  $t$  zamanda seçtiği  $a_t$  eyleminde, bir  $r_t$  ödülü gözlemler ve yeni bir  $s_{t+1}$  durumuna girer (bu hem önceki  $s_t$  durumuna hem de seçilen eyleme bağlı olabilir) ve  $Q$  güncellenir. Algoritmanın temeli, basit bir değer yineleme güncellemesi olan Bellman denklemi kullanarak geçerli değerlerin ağırlıklı ortalamasını ve yeni bilgileri kullanmaktır [8].

### 2.1.2 SARSA

Durum-eylem-ödül-durum-eylem (SARSA), makine öğreniminin, pekiştirmeli öğrenme alanında kullanılan Markov karar süreci politikasını öğrenmek için kullandığı bir algoritmadır. Rummery ve Niranjan, "Değiştirilmiş Bağlantıcı  $Q$ -Öğrenme" (DBQ-Ö) adlı bir teknik önerdiler [9]. Bu ad,  $Q$ -değerini güncellemeye yönelik ana işlevin, " $S_1$ " ajanın mevcut durumuna, " $A_1$ " ajanın seçtiği eyleme, " $R$ " ajanın bu eylemi seçtiği için aldığı ödüle, " $S_2$ " ajanın bu eylemi yaptıktan sonra girdiği duruma ve son olarak " $A_2$ " ajanın yeni durumunda seçtiği bir sonraki eyleme bağlı olduğu gerçeğini yansıtır.  $(s_t, a_t, r_t, s_{t+1}, a_{t+1})$  beşlinin kısaltması SARSA'dır. Bir SARSA ajanı, çevre ile etkileşime girer ve gerçekleştirilen eylemlere dayalı olarak politikayı günceller, dolayısıyla bu, politikaya dayalı öğrenme algoritması olarak bilinir. Bu güncelleme işlemi denklem (1)'e göre yapılır.

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

Bir durum eylemi için  $Q$  değeri, alfa öğrenme oranı tarafından ayarlanan bir hatayla güncellenir.  $Q$  değerleri,  $s$  durumunda  $a$  eylemi gerçekleştirmek için bir sonraki zaman adımında alınan olası ödülü, artı bir sonraki durum eylemi gözleminde alınan indirimli gelecekteki ödülü temsil eder. Watkin'in  $Q$ -öğrenmesi, mevcut eylemlerin maksimum ödülüne dayalı olarak optimal durum-eylem değeri fonksiyonu  $Q^*$  tahminini günceller. SARSA kendi izlediği politikayı almakla ilişkili  $Q$  değerlerini öğrenirken, Watkin'in  $Q$ -öğrenmesi bir keşif/sömürü sürecini izlerken en uygun politikayı almakla ilişkili  $Q$  değerlerini öğrenir. Watkin'in  $Q$ -öğrenmesinin bazı optimizasyonları, SARSA'ya uygulanabilir [10].

### 2.1.3 R-Öğrenme

R-öğrenme, Schwartz tarafından, 1993 yılında önerilen ve ortalama ödülü dikkate alan bir PÖ algoritmasıdır. R-öğrenme, Q-öğrenme ile aynı Q-değerlerini güncellemek için mümkün olan en iyi eylemi göz önünde bulundurur, ancak ortalama ödül, aşağıdaki formül (2)'de gösterildiği gibi anlık ödülle değiştirilir. Denklem (3)'te,  $\rho$ ,  $\pi$  politikası kapsamında karar süresi adımı başına beklenen ödülün yaklaşık değeridir ve  $u$ , karar süresi adım sayısıdır [11].

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} - \rho + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)] \quad (2)$$

$$\rho = \sum_{t=1}^u \frac{r_t}{u} \quad (3)$$

### 2.1.4 SMART

Yarı Markov karar süreci problemlerinde, zincirdeki geçiş süreleri, sabit birim yerine genel dağılımlı rassal değişkenlerdir. Bu nedenle, bu tür problemlerde geçiş süreleri, tekleştirme süreciyle sabit birime dönüştürülerek, Q-öğrenme ve benzeri ödüllü öğrenme algoritmaları yardımıyla Markov karar süreci problemlerinin çözümü için kullanılabilir. Ancak, yarı Markov Karar Süreci (MKS) problemlerinin tam olarak MKS'ye dönüştürülmesi her zaman mümkün değildir. Ayrıca, tekleştirme süreci için geçiş olasılıklarının bilinmesi gerekmektedir. Literatürde yer alan SMART algoritması, yaklaşık optimal olarak yarı Markov karar sürecini, Markov karar süreci problemlerine dönüştürmeden çözebilen bir ödüllü öğrenme algoritmasıdır. Algoritmanın motivasyonu, (4)'te verilen Bellman'ın optimalite denkleminde kaynaklanmaktadır:

$$v(x) = \min_{a \in A(x)} [g(x, a) - \rho^* y(x, a) + \sum_{x' \in x} P(x'|x, a)v(x')] \quad (4)$$

### 2.1.5 Derin Q ağı

Derin Q ağı (DQA)'da Q-değerlerinin parametreleştirilmesiyle Q-fonksiyonuna yaklaşmak için bir sinir ağı uydurulur:  $Q(s, a, \theta_i)$ . Her  $k$  iterasyonun hedefi (5)'te tanımlanmıştır [12].

$$Y_{hedef} = r + \gamma \max_a Q(s', a'; \theta_i) \quad (5)$$

Ağı eğitmek için parametreler, kayıp işlevi ve bir yeniden oynatma arabelleği kullanılarak ayarlanır. Ağ, her bölümde  $\langle s, a, s', r \rangle$  deneyimlerini bir yeniden oynatma arabelleği  $\mathcal{M}$ 'de depolar. Ağ eğitirken, bir mini-toplu deneyim tekrar oynatma arabelleği  $(U(\mathcal{M}))'$ den eşit şekilde çekilir. Bir MKS yörüngesinin deneyimleri birbirine büyük ölçüde bağlı olduğundan, bu deneyimler üzerine eğitim, değişkenlik ve istikrarsızlık getirir. Tekrar oynatma arabelleğinde saklanan tüm önceki deneyimler üzerinde tekdüze örnekleme yoluyla, mini deneyim grubu, öğrenme sürecinin istikrarını artıran bağımsız ve aynı şekilde dağıtılmış bir küme haline gelir. Parametreler, kayıp fonksiyonunun stokastik gradyan inişiyle güncellenir. Kayıp işlevi, ağın beklenen değerini, gerçek değerle karşılaştırır ve karesi alınmış kaybın minimizasyonu, denklem (6) gibi tanımlanır [12].

$$L_i(\theta_i) = \mathbb{E}_{\langle s, a, s', r \rangle \sim U(\mathcal{M})} \left( \underbrace{r + \gamma \max_a Q(s', a'; \theta_i^-)}_{hedef} - Q(s, a; \theta_i) \right)^2 \quad (6)$$

DQA, "çalışan hedef" sorununu çözmek için hedef değerleri oluşturmak üzere ayrı bir hedef ağ tutar. Hedef değerlerini tahmin etmek ve ağırlıklarını güncellemek için tek bir ağ kullanılırsa, politikadaki salınımlar,

öğrenme sürecini istikrarsızlaştırır [13]. Her bölümde, politika ağının parametreleri, bellekteki deneyimler kullanılarak güncellenir. Her  $C$  yinelemede, hedef ağ, öğrenen ağdan güncellenen ağırlıkları alır [12].

### 2.1.6 İlişkisel pekiştirmeli öğrenme

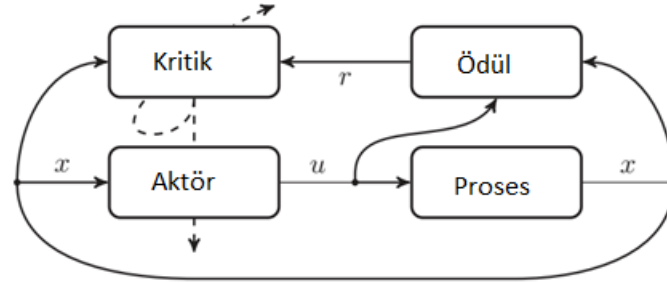
İstatistiksel ilişkisel öğrenme ve muhakeme, belirsiz ilişkileri inceler ve varlıkların ve onların ilişkilerinin yapılandırılmış temsillerini, bunların nasıl oluşturulacağına ilişkin kurullarla manipüle eder. Tümevarımsal Mantık Programlama (TMP), belirsiz mantık kurallarını olumlu ve olumsuz örneklerden öğrenir. Olasılıksal, istatistiksel ilişkisel öğrenme ile yakından ilişkilidir. Olasılığa dayalı TMP, kural tabanlı öğrenmeyi, istatistiksel öğrenmeyle bütünleştirir ve TMP'nin yüksek karmaşıklığının üstesinden gelir. Grafik modeller, istatistiksel ilişkisel öğrenme için önemli yaklaşımlardır. Yapay sinir ağlarının bağlantıcılık, paralel dağıtılmış işleme ve sinirsel hesaplama gibi alternatif isimleri vardır. Sembolizm, matematik ve mantık tarafından tanımlanan, sembolleri ve kuralları olan resmi bir dille ilgilidir. Sinir ağları ile ilişkisel öğrenme ve akıl yürütme, bağlantıcılık ve sembolizmi bütünleştiren bir yaklaşımdır. İlişkisel pekiştirmeli öğrenme, bilgi temsili ve muhakeme için PÖ'yü istatistiksel ilişkisel öğrenme ile bütünleştirir ve PÖ'yü klasik yapay zekâ ile birleştirir [14].

### 2.1.7 Zamansal fark öğrenmesi

Zamansal Fark (ZF) öğrenmesi, pekiştirmeli öğrenmenin temel yöntemlerinden biridir ve dinamik programlama ile Monte Carlo yöntemlerinin bir kombinasyonunu temsil eder. Bu yöntem, ortamın modelini öğrenme gerektirmediği gibi, sonucu beklemeksizin diğer tahminlerden yardım alarak tahminleri güncelleyebilme özelliğine sahiptir. Bu yöntemin ilk aşaması, tahmin aşamasıdır, yani bir politika için değer fonksiyonunu tahmin etmektir. Ardından, kontrol aşamasına geçerek en iyi politikayı bulmaya çalışır. ZF öğrenmenin, ortamın modeline ihtiyaç duymadan çalışabilen bir yöntem olduğu görülmektedir. Bu yönü, dinamik programlamaya kıyasla en büyük artısıdır. Monte Carlo yöntemine kıyasla en önemli artısı, bölüm sonunu beklemek zorunda kalmadan tahmini güncellemesidir [7]. Zamansal fark öğrenme yöntemlerinin, yeterince küçük adım boyutu kullanılarak herhangi bir sabit politikadan, optimum politikaya yakınsadığı kanıtlanmıştır. Ayrıca, matematiksel yönden kanıtlanmasa da uygulamada, zamansal fark öğrenme yöntemlerinin çoğunlukla Monte Carlo yöntemlerinden daha hızlı bir şekilde yakınsadığı gözlenmiştir [7].

### 2.1.8 Aktör-Kritik

Aktör-kritik yöntemleri, yalnızca aktör ve yalnızca eleştirmen yöntemlerinin avantajlarını birleştirmeyi amaçlar. Yalnızca aktöre yönelik yöntemler gibi, aktör-kritik yöntemler de sürekli eylemler üretme yeteneğine sahipken, yalnızca aktöre yönelik yöntemlerin politika gradyanlarındaki büyük varyansa bir kritik eklenerek karşı konulur. Kritik 'in rolü, aktörün öngördüğü mevcut politikayı değerlendirmektir. Prensip olarak, bu değerlendirme, ZF, artık gradyanlar gibi yaygın olarak kullanılan herhangi bir politika değerlendirme yöntemiyle yapılabilir. Kritik, örnekleri kullanarak değer işlevine yaklaşır ve günceller. Değer işlevi, daha sonra aktörün, politika parametrelerini, performans iyileştirme yönünde güncellemek için kullanılır. Bunlar, yalnızca kritik yöntemlerin aksine, genellikle politika gradyan yöntemlerinin istenen yakınsama özelliklerini korur. Aktör-kritik yöntemlerde, politika, değer işlevinden doğrudan çıkarılmaz. Bunun yerine, politika yalnızca küçük bir  $\alpha_a$  adım boyutu kullanılarak politika gradyanı yönünde güncellenir; bu, değer fonksiyonundaki bir değişikliğin, politikada yalnızca küçük bir değişikliğe yol açacağı ve politikada daha az veya hiç salınım davranışına yol açmayacağı anlamına gelir. Şekil 2, bir aktör-kritik algoritmasının şematik yapısını göstermektedir [15].



Şekil 2. Aktör-kritik algoritmasına şematik genel bakış.  
Kesikli çizgi, kritiğin aktörü ve kendisini güncellemekten sorumlu olduğunu gösterir.

Şekil 2’de görüldüğü gibi öğrenen ajan iki ayrı varlığa bölünmüştür: aktör (politika) ve eleştirmen (değer fonksiyonu). Aktör, yalnızca geçerli  $x$  durumu verildiğinde bir kontrol girişi  $u$  oluşturmaktan sorumludur. Eleştirmen, aldığı ödülleri işlemekten, yani değer fonksiyonu tahminini uyarlayarak mevcut politikanın kalitesini değerlendirmekten sorumludur. Eleştirmen tarafından bir dizi politika değerlendirme adımından sonra, aktör, eleştirmenden gelen bilgiler kullanılarak güncellenir. Burada açıklanan aktör-kritik algoritmaları için birleşik bir gösterim, aralarında daha kolay bir karşılaştırma yapılmasına izin verir. Ek olarak, çoğu algoritma, standart güncelleme kurallarının genel bir şablonuna uyarlanabilir. Aktör-kritik algoritmalarında veya bu konudaki diğer herhangi bir PÖ algoritmasında amaç, bazı durağan MKS verildiğinde mümkün olan en iyi politikayı bulmaktır. Bunun ön şartı, eleştirmenin belirli bir politikayı, doğru bir şekilde değerlendirebilmesidir. Başka bir deyişle, eleştirmenin amacı, o politika için Bellman denkleminde yaklaşık bir çözüm bulmaktır [15].

## 2.2 Eylem Seçimi

Pekiştirmeli öğrenmede ortaya çıkan zorluklardan biri, keşif ve sömürü arasındaki ödünleşimdir. Yüksek bir ödül elde etmek için, bir pekiştirmeli öğrenme ajanı, geçmişte denediği ve ödül üretmede etkili bulunduğu eylemleri tercih etmelidir. Ancak bu tür eylemleri keşfetmek için daha önce seçmediği eylemleri denemesi gerekir. Ajan, ödül elde etmek için zaten bildiklerini kullanmak zorundadır, ancak gelecekte daha iyi eylem seçimleri yapmak için de keşfetmek zorundadır. Buradaki ikilem ne keşif ne de sömürünün, görevde başarısız olmadan münhasıran takip edilemeyeceğidir [9]. Ajan, farklı eylemleri denemeli ve daha iyi olanları aşamalı olarak seçmelidir. Etkili bir öğrenme yöntemi oluşturmak için keşif ve sömürü arasındaki değiş tokuşun uygun şekilde kontrol edilmesi önemlidir. Yaygın olarak kullanılan üç eylem seçim yöntemi, greedy,  $\epsilon$ -greedy ve softmax kısaca aşağıda açıklanmıştır. Ajan, olası eylemler arasından en iyisini seçmeye karar verirse, açgözlü bir eylem seçim stratejisi izlediğini söyleyebiliriz. Bununla birlikte, her zaman en iyi eylemi seçmek, eylem ödülleri varyansına bağlı olarak yetersiz performansa yol açabilir [16].

Bu açgözlü davranışa bir alternatif,  $\epsilon$ -açgözlü seçim stratejisini izlemektir. Bu eylem seçim yöntemi, ajana çoğu zaman en iyi eylemini seçmesini, ancak bazen rastgele bir eylemi seçmesini söyler (mevcut  $s$  durumundaki her olası eylem  $a$  için eşit olasılıkla). Rastgele bir eylem seçme olasılığını,  $\epsilon$  değeri belirler.  $\epsilon$ -greedy, pekiştirmeli öğrenmede, keşif ve sömürüyü dengelemek için etkili ve popüler bir strateji olmasına rağmen, bir dezavantajı, keşifte tüm eylemler arasından eşit olarak seçim yapmasıdır [9]. Bu, görünen en kötü eylem ile aynı olasılıkla en iyiye yakın eylem arasında seçim yapabileceği anlamına gelir. Alternatiflerden biri, eylem olasılıklarını tahmini değerlerinin dereceli bir fonksiyonu olarak değiştirmektir; softmax eylem seçim stratejisinin yaptığı da budur. Açgözlü eylem yine de en yüksek olasılığa sahip olacaktır, ancak diğerleri değer tahminlerine göre sıralanır. Bu,  $m$  olası eylemden  $a$  eylemini seçme olasılığının, bu eylem seçim mekanizmasını kullanırken en sık kullanılan dağılım olan Boltzmann dağılımı tarafından verildiği anlamına gelir. Olasılık, denklem (7) ile hesaplanır.

$$\Pr(a) = \frac{e^{Q_t(a)/\tau}}{\sum_{b=1}^m e^{Q_t(b)/\tau}} \quad (7)$$

Denklem (7)'de  $\tau$ , ajanın ne kadar açgözlü davranacağını kontrol eden, sıcaklık adı verilen pozitif bir parametredir,  $m$ , mevcut eylemlerin sayısını temsil eder ve  $Q_t(a)$ ,  $t$  zaman adımıdaki  $a$  eyleminin tahminini temsil eder. Yüksek sıcaklıklar, eylemlerin tamamen (neredeyse) eşit olasılığa sahip olmasına neden olur. Düşük sıcaklıklar, değer tahminlerinde farklılık gösteren eylemler için seçim olasılığında daha büyük bir farka neden olur, başka bir deyişle, düşük sıcaklık değerleri, ajanın daha açgözlü davranmasına neden olur. Softmax eylem seçimi ile  $\varepsilon$ -greedy eylem seçimi arasındaki benzerlikleri fark etmek mümkündür, her iki yöntemin de ayarlanması gereken yalnızca bir parametresi vardır, ancak çözülmekte olan göreve bağlı olabileceğinden hangisinin daha iyi olabileceği açık değildir. Her iki yöntemin de zaman içinde parametrelerinin değerini ( $\varepsilon$  ve  $\tau$ ) düşürmesi mümkündür, bu, ajanların, öğrenme aşamasının başında daha fazlasını keşfedecekleri ve sonunda daha açgözlü hareket edecekleri anlamına gelir. Hangi eylem seçim stratejisinin daha iyi olduğunu belirlemek için çeşitli deneyler geliştirilmiştir. Deneyler, keşif stratejileri için farklı parametrelerin çok farklı sonuçlara yol açtığını göstermiştir. Çok az araştırma, öğrenme sürecinin sonunda yetersiz sonuçlara yol açarken, çok fazla araştırma, öğrenme sürecinde kötü performansa ve uzun öğrenme sürelerine yol açar. O halde en iyi  $\varepsilon$  seçimi, probleme bağlıdır [16].

### 2.3 Durum Tanımı

Durum tanımı, önemli bir konudur. Birçok problemde, durum uzayının büyük ve sürekli olması gibi zorluklar vardır. Bu zorluklarla başa çıkabilmek ve değer fonksiyonlarına yaklaşmak için farklı durum tanımlama stratejileri kullanılır. İlgili literatürde ele alınan stratejiler aşağıda verilmiştir.

1. *Tüm Durumlar*: Bu yöntemde, ortamın her bir durumu  $Q$ -tablosunda yer alır. Ancak bu yöntem yalnızca küçük ölçekli problemlere uygundur ve durumlar genellikle tek bir özellikle tanımlanır.
2. *Durum Toplama*: Durumlar, durum uzayını azaltmak için bir araya toplanır. Genellikle bir veya iki özellikle tanımlanan durumlar, önceden belirlenen aralıklara dayalı olarak kategorize edilir.
3. *Kümeleme*: Durumlar, bir dizi özellik tarafından tanımlanır ve benzerlik veya farklılık dikkate alınarak kümeler oluşturulur.
4. *Yapay Sinir Ağları*: İnsan beyninin nöral yapısından esinlenerek geliştirilen yapay sinir ağları, girdi verilerini, çıktı verileriyle eşleştiren ve genelleme yeteneği kazanan öğrenme teknikleridir.
5. *Vaka Tabanlı Akıl Yürütme*: Benzer problemlerin, benzer çözümleri olduğu fikrine dayanır. Ziyaret edilen durum-eylem çiftleri, bir vaka tabanında saklanır ve yeni bir durumla en benzer durumun eylemleri tahmin edilir.
6. *Regresyon*: İstatistiksel bir öğrenme tekniği olan regresyon, girdi değerlerine dayalı olarak çıktı değerlerini bulmayı amaçlayan bir fonksiyon bulma yöntemidir.
7. *Regresyon Ağacı*: Belirli bir karar ağacı türü olan regresyon ağacı, gerçek değerli değişkenleri işleyerek hedef değerleri tahmin etmeyi amaçlar.
8. *Gradyan İniş Yöntemi*: Durumlar, öznitelik vektörleriyle temsil edilir ve değer fonksiyonları, bu vektörlerin doğrusal bir kombinasyonu ile oluşturulur.

Bu stratejiler, durum uzayının boyutunu azaltma, benzer durumları gruplama ve tahmin etme gibi yöntemlerle, değer fonksiyonlarına yaklaşmayı kolaylaştırır. Ancak her bir stratejinin kendi avantajları ve zorlukları bulunmaktadır. Gelecekteki araştırmalar, bu stratejilerin etkinliğini ve farklı durum tanımlama yöntemlerinin karşılaştırılmasını daha da inceleyebilir ve çizelgeleme problemleri için daha iyi çözümler sunabilir.

### 3. İŞ ÇİZELGELEME PROBLEMİ

İşlerin, belirli hazırlık süreleri, işlem süreleri ve teslim zamanları olduğu ve bu işlerin belirli sayıda makinelerde işlendiği durumlarda, işlerin, optimize edilmiş bir sıralamasına ihtiyaç duyulmaktadır. Bu tür problemlere, iş sıralama veya iş çizelgeleme denir. Çizelgeleme problemlerinin çözümü için çeşitli teknikler ve yöntemler geliştirilmiştir. Simülasyon çalışmaları ve sezgisel yaklaşımlar, çizelgeleme problemlerinin geliştirilmesi ve uygulanmasında büyük fayda sağlamaktadır. Kullanılacak teknikler, problemin karmaşıklığına, model yapısına ve amaçlarına bağlı olarak değişiklik gösterir. Çizelgeleme



problemleri, işlerin geliş biçimine, makine sayısına, üretim şekline ve performans kriterine göre dört farklı şekilde sınıflandırılabilir. Çizelgeleme problemleri literatürde,  $\alpha / \beta / \gamma$  üçlü notasyon ile tanımlanmaktadır. Bu notasyonlarda,  $\alpha$ , makine çevresini ifade etmektedir. Diğer bir değişle çizelgeleme probleminin, akış tipi, atölye tipi, paralel makine vd. olduğunu ifade etmektedir.  $\beta$  notasyonu, işlem özelliklerini ifade etmektedir. Örneğin, beklemesiz olması, hazırlık süresinin olması, sıra bağımlı vd. olduğunu belirtmektedir.  $\gamma$  notasyonu da performans kriterini ifade etmektedir. Örneğin, maksimum tamamlanma zamanının minimize edilmesi, maksimum gecikmenin minimize edilmesi vd. gibi tanımlar [17].

İşlerin geliş biçimine göre sınıflandırıldığında, çizelgeleme problemleri statik ve dinamik olarak ikiye ayrılır. Statik durumda, tüm işler işlem alanında hazır durumdadır ve çizelgeleme süreci boyunca yeni işler eklenmez. Ayrıca işlerin ne zaman geldiği bilinmektedir. Dinamik durumda ise işlerin süresi ve geliş zamanı belirsizdir. İşler, sürekli olarak bir prosese göre gelir ve kuyuklar oluşabilir, bu nedenle sıralamada güncellemeler yapılması gerekmektedir. Dinamik durumlarda genel amaç, makine boş kalma süresini ve kuyukta bekleyen iş sayısını en aza indirmektir [18]. Çizelgeleme problemlerini, üretim tipine göre, akış tipi ve atölye tipi olarak iki kategori altında ele alabiliriz. Akış tipi çizelgeleme problemleri, birbirinden farklı  $m$  adet makine ve  $n$  adet iş'ten oluşmakta ve her bir iş, aynı sıra ile  $m$  farklı makinede işlem görmektedir [19]. Akış tipi çizelgeleme problemlerinde, esnek akış tipi, çok prosesli esnek akış tipi, beklemesiz akış tipi, bloklama kısıtlı akış tipi ve dağıtılmış akış tipi olmak üzere birçok versiyona sahiptir.

Esnek akış tipi çizelgeleme problemlerinde,  $n$  adet iş  $k$  aşamada aynı sıra ile işlem görmekte ve her bir aşamada özdeş  $m_k$  adet paralel makine bulunmaktadır. Çok prosesli, esnek akış tipi çizelgeleme problemi de esnek akış tipi çizelgeleme probleminin özel bir hali olup her aşamada makinelerde birden fazla proses gerçekleştirilir [20]. Beklemesiz akış tipi çizelgeleme problemlerinde,  $m$  adet makinede aynı sıra ile işlem gören  $n$  adet işin, makinelerde işlem görürken, ardışık iki makine arasında beklemesine müsaade edilmez. Makineler bekleyebilir fakat işlerin beklemesine müsaade edilmez [21]. Bloklama kısıtlı, akış tipi çizelgeleme problemlerinde, işler, aynı sıra ile makinede işlem görürken, bir makinede işlem gördükten sonra bir sonraki makine boş değil ise son işlem gördüğü makinede beklemek zorunda kalır. Bir sonraki makine boşaldığında, iş, işlem görmek üzere bir sonraki makineye geçer. İşin, bir sonraki makinede işlem görmek için beklemesinden dolayı geçen zaman, bloklama süresi olarak isimlendirilir [22]. Dağıtılmış, akış tipi çizelgeleme problemlerinde,  $n$  adet iş,  $F$  adet özdeş fabrikaya dağıtılarak  $m$  adet makine de aynı sıra ile işlem görmektedir [23].

Atölye tipi çizelgeleme problemi de birbirinden farklı  $n$  adet iş  $m$  adet makineden oluşmaktadır. İşlerin makinelerde işlem görme sırası, her bir işin rotasına bağlı olarak değişmektedir. Atölye tipi çizelgeleme problemi de esnek atölye tipi, açık atölye tipi çizelgeleme olmak üzere birçok versiyona sahiptir. Esnek atölye tipi çizelgeleme problemlerinde,  $n$  adet iş,  $c$  adet iş merkezinde, iş rotalarına göre işlem görmekte ve her bir iş merkezinde özdeş  $m_c$  adet paralel makine bulunmaktadır [24]. Açık atölye tipi çizelgeleme problemi, birbirinden farklı  $n$  adet iş,  $m$  adet makineden oluşmaktadır. İşlerin makinelerde işlem görmesi ile ilgili herhangi bir iş sırası ya da rota bulunmamaktadır [25]. Çizelgeleme problemlerini değerlendirirken performans kriterleri kullanılır. Bu kriterler, tamamlanma süreleri, teslim zamanları ve tesisin verimli kullanımı ile ilgilidir. Tek bir performans kriteri kullanılabilmesi gibi birden fazla kriter de kullanılabilir. Çizelgeleme problemlerinin çözümünde kullanılan tekniklerin değerlendirilmesinde performans kriterlerinin dikkate alınması önemlidir. Bu kriterler kısaca aşağıda açıklanmıştır.

1. Tamamlanma süresiyle ilgili performans kriterleri [26]:

- Maksimum tamamlanma zamanı,
- Ortalama tamamlanma zamanı,
- Ortalama akış süresi,
- Ortalama bekleme süresi.

2. Teslim zamanıyla ilgili performans kriterleri işletmeler için daha önemli ve zor çözümler gerektirebilir. Bu kriterler:

- Maksimum gecikme süresi,
- Geciken iş sayısı,
- Ortalama gecikme.

3. Tesisin verimli kullanılmasıyla ilgili performans kriterleri:

- Bekleyen iş sayısı,
- Tamamlanmamış iş sayısı,
- Makinelerin boş bekleme süresidir.

Çizelgeleme problemlerinin çözümünde, literatürde tek ve çok performans kriterleri kullanılmaktadır. Çok performans kriterli çizelgeleme problemlerinin çözümü, tek performans kriterli problemlerinin çözümünden daha zor dur, çünkü aynı anda tüm performans kriterlerinin optimize edilmesi mümkün olmayabilir. Bir performans kriterinin değerini optimize ederken diğer performans kriterinin değeri azalabilir [27].

#### 4. LİTERATÜR ARAŞTIRMASI

Kayhan ve Yıldız [11], makine çizelgeleme problemlerinde, pekiştirmeli öğrenme uygulamaları ile ilgili 1995- 2020 yılları aralığını kapsayan bir literatür çalışması yapmışlardır. Bu çalışmada, literatürde ilk defa, pekiştirmeli öğrenme yöntemlerinin çizelgeleme problemlerinin çözümü üzerindeki etkisi, 2003-2023 yılları aralığında incelenmiştir. Araştırmada, “scholar.google.com” arama motoru yardımıyla, “Reinforcement Learning and scheduling” anahtar kelimeleri kullanılmıştır. Tarama sonucunda on dokuz bin yüz adet sonuç elde edilmiştir. Elde edilen sonuçlar, tekrar analiz edilerek, doğrudan makine çizelgeleme ile ilgili olan (tek makine, paralel makine, akış tipi, atölye tipi ve diğer makine çizelgeleme) 50 adet araştırma tespit edilmiş ve detaylı olarak incelenerek aşağıda sunulmuştur.

##### 4.1 Tek Makine Çizelgeleme Problemleri

Wang ve Usher [28], tek makinedeki sevk kuralı seçim problemine,  $Q$ -öğrenme uygulayarak, makine ajanının önceden tanımlanmış sevk kurallarını en iyi şekilde öğrenmesini sağlamışlardır. Wang, Yang ve Zhang [29], kötüleşen etkilere sahip, çok durumlu tek makine sisteminde, üretim çizelgeleme ve esnek önleyici bakıma entegre bir optimizasyon problemini incelemişlerdir. Zaman temelli, esnek önleyici bakım ve koşul temelli, esnek önleyici bakım kullanarak, proaktif bir şekilde, makine arızalarıyla başa çıkmışlar ve üretim çizelgesi üzerinde iyileştirmeler yapmışlardır. Yang, Li ve Wang [30], bozulma süreçleri yaşayan, çok durumlu, tek makineli, üretim sistemleri için üretim çizelgelemenin entegre optimizasyon problemini ele almışlardır. İşleme ve bakım maliyetleri ile tamamlama ödülleri birlikte dikkate alarak, optimum üretim politikalarını belirlemek için çalışmışlardır. Deliktaş [31], aynı anda tamamlama süresi ve toplam gecikmenin minimize edilmesini gerektiren iki hedefi göz önünde bulundurarak, öğrenme etkisi ve serbest bırakma sürelerine sahip, tek makine çizelgeleme problemi önermiştir. Memetik algoritmalara dayalı çeşitli optimizasyon teknikleri kullanmış ve iki amaçlı problemin çözümü için farklı skalarizasyon yöntemlerini birleştirmiştir.

##### 4.2 Paralel Makine Çizelgeleme Problemleri

Lopes Silva, Souza, Freitas Souza, Bazzan [32], optimizasyon için çok ajanlı bir çerçeve olan Metasezgisel için Çok Ajanlı Mimari (MÇAM) sunmuşlardır. Deneyler, ajanın öğrenme yeteneğinin, bireysel ve ekip çalışması açısından çözümlerin kalitesini doğrudan etkilediğini göstermiştir. Liangxun Guo [33], dinamik, özdeş olmayan paralel makine çizelgeleme problemini, makine hızı farklılıkları, belirsiz görev varış zamanı ve çoklu optimizasyon metrikleri ile karakterize ederek incelemiştir. Pekiştirmeli öğrenme yaklaşımı ile farklı hedefler ve tekrar tekrar eğitilen çok aşamalı bir öğrenme süreci kullanarak problemi çözmeyi önermiştir. Chien ve Lan [34], sıra bağımlı kurulum süresi ile bağımsız paralel makine çizelgeleme problemi için yeni bir ajan tabanlı yaklaşım geliştirmişlerdir. Çalışmalarında, derin pekiştirmeli öğrenme

ve hibrit genetik algoritmayı birleştirilerek bir planlama ajanını eğitmek için derin  $Q$  ağı (DQA) kullanmışlardır.

### 4.3 Akış Tipi Çizelgeleme Problemleri

Arviv, Stern ve Edan [35], özdeş iş ve makinelere sahip, iki robotlu bir akış tipi çizelgeleme problemini, dört farklı robot iş birliği seviyesi için tanımlamış ve değerlendirmişlerdir. Amaçları, farklı işlem sürelerine sahip makineler için maksimum tamamlanma zamanını minimize eden bir robot programı elde etmektir. Wang ve Tang [36], ayrık permütasyon akış tipi çizelgeleme problemini çözmek için, makine öğrenimi tabanlı çok amaçlı memetik algoritma önermişlerdir. Yöntemleri, her aşama için domine edilmemiş çözümleri depolayarak ayrı bir arşiv kullanıp yeni bir popülasyon güncelleme sunar. Ayrıca, önerdikleri yöntem adaptif çok amaçlı yerel arama geliştirmek için geçmiş verilerin analizini kullanır. Shao, Pi, Shao [37], aylak süresi olmayan, akış tipi çizelgeleme problemini, toplam gecikme kriteri ile çözmek için karma ayrık öğretim-öğrenme tabanlı meta-sezgisel algoritma önermişlerdir. Önerdikleri öğretim-öğrenme olgusunu, gerçek dünyada taklit etmek için olasılıksal modele dayalı, ayrık öğretim aşaması, hiyerarşik yapıya dayalı ayrık öğrenme aşaması ve pekiştirmeli öğrenme olmak üzere üç aşama sunmuşlardır. Han, Guo ve Su [38], hibrit akış tipi çizelgeleme problemini çözmek için pekiştirmeli öğrenme yöntemini incelemişlerdir. Pekiştirmeli öğrenmeyi, bir otomobil motoru fabrikasının metal işleme atölyesinde çizelgeleme için kullanmışlardır. Zhao vd. [39], maksimum montaj tamamlanma süresini en aza indirme hedefiyle, dağıtılmış montaj aylak süresi olmayan, akış tipi çizelgeleme problemini incelemişler, çözmek için su dalgası optimizasyon algoritmasını önermişlerdir. Chen vd. [40], çoklu makine iş birliği ile akış tipi çizelgeleme problemini ele almışlar ve maksimum tamamlama süresini ve toplam makine yükünü en aza indirme hedefleriyle pekiştirmeli öğrenmeye dayalı uyarlanabilir çok popülasyonlu gri kurt optimize edici geliştirmişlerdir. Pan vd. [41], maksimum tamamlanma süresini en aza indirme hedefiyle, permütasyon akış tipi çizelgeleme problemini çözmek için derin pekiştirmeli öğrenmeye dayalı bir optimizasyon algoritması önermişlerdir. Cai vd. [42], üretim, nakliye ve montajı içeren dağıtılmış, montaj hibrit akış tipi çizelgeleme problemine,  $Q$ -öğrenme ile yeni bir karışık kurbağa öğrenme algoritması önermişlerdir. Zhao, Hu, vd. [43], çok amaçlı, enerji açısından verimli, dağıtılmış montaj ve beklemesiz akış tipi çizelgeleme problemini çözmek için pekiştirmeli öğrenme güdümlü bir beyin fırtınası optimizasyon algoritması önermişlerdir. Problemin amaçları, maksimum montaj tamamlanma süresini ( $C_{max}$ ) ve toplam enerji tüketimini en aza indirme ile dengeli kaynak tahsisini sağlamayı içermektedir. Zhao, Wang, vd. [44], sıra bağımlı kurulum sürelerine sahip, dağıtılmış, heterojen, beklemesiz akış tipi çizelgeleme problemini ele almışlardır. Fabrika konfigürasyonundaki ve nakliye süresindeki farklılıkları modelde dikkate alarak  $Q$ -öğrenmeli, yapay bir arı kolonisi algoritması önermişlerdir. Zhao, Jiang, vd. [45], sıra bağımlı kurulum süresi ile enerji verimli, dağıtılmış, beklemesiz bir akış tipi çizelgeleme problemini, maksimum tamamlanma zamanı ve toplam enerji tüketimini en aza indirmek için  $Q$ -öğrenmeye dayalı işbirlikçi bir meta-sezgisel algoritma kullanarak çözmüşlerdir. Yan vd. [46], dağıtılmış, permütasyon akış tipi çizelgeleme probleminin, esnek periyodik önleyici bakım ile entegre bir optimizasyonunu incelemişlerdir. Bu problemin çözümü için açgözlülük oranı azalan, derin  $Q$  ağ tabanlı bir çözüm çerçevesi tasarlamışlardır. Nahhas vd. [47], gerçek dünyadaki dört aşamalı bir hibrit akış tipi çizelgeleme problemini çözmek için ayrık olay simülasyon modeli ve derin pekiştirmeli öğrenme tekniklerine dayanan bir yaklaşım sunmuşlardır. He vd. [48], akış tipi çizelgeleme problemini optimize etmek için geliştirilmiş bir  $Q$ -öğrenme algoritması önermişlerdir. Maksimum tamamlanma zamanı, geri bildirim sinyali olarak kullanmışlar ve çevresel durum değişikliği süreci, iş seçme süreci olarak tanımlamışlardır. Algoritmanın verimliliğini, NEH sezgisel yöntemi ile birleştirilerek iş ekleme modu değiştirilerek artırmışlardır. Dong vd. [49], Permütasyon Akış Tipi Çizelgeleme Problemini (PATÇP) çözmek için kombinatoriyal optimizasyon ve derin öğrenme alanının yenilikçi bir kombinasyonunu önermişlerdir. Yöntemleri, politika ağı, aday işlerin dağılımını tahmin etmek için PATÇP'nin grafik temsilini ve işlerin sıra bilgilerini birleştirir. Yang ve Xu [50], gerçek zamanlı ve akıllı karar verme için derin pekiştirmeli öğrenmeyi kullanan, dinamik iş gelişleriyle, dağıtılmış permütasyon akış tipi çizelgeleme problemini ele almışlardır. Amaçları, tüm işlerin toplam gecikme maliyetini en aza indirmektir. Farklı üretim durumları için verimli eylemler sağlamak üzere farklı problemlere özel dağıtım

kuralları geliştirmişlerdir. Ying ve Lin [51], dağıtılmış, montaj, permütasyon akış tipi çizelgeleme problemini, ilişkisiz, montaj makinelerini ve sıra bağımlı kurulum sürelerini dikkate alacak şekilde incelemişlerdir. Bir karma tamsayılı doğrusal programlama modeli ve pekiştirmeli öğrenme yinelemeli, açgözlü algoritması olan bir meta-sezgisel algoritma, bu problemin çözüm süresini en aza indirmek için önermişlerdir.

#### 4.4 Atölye Tipi Çizelgeleme Problemleri

Gabel ve Riedmiller [52], atölye çizelgeleme problemlerini, sıralı karar süreçleri olarak yorumlayarak, her kaynağa iş gönderme kararlarını diğer ajanlardan bağımsız olarak veren bir pekiştirmeli öğrenme algoritması kullanarak, deneme yanılma yoluyla, sevk etme davranışını geliştiren uyarlanabilir bir ajan önermişlerdir. Luo [53], dinamik üretim durumlarıyla başa çıkmak ve her bir yeniden çizelgeleme noktasında en uygun eylemi (yani sevkiyat kuralını) öğrenmek için derin  $Q$ -ağ geliştirmiştir. Aynı zamanda, bir işlemi seçmek ve bir işlem tamamlandığında veya yeni bir iş geldiğinde uygun bir makineye atamak için altı bileşik dağıtım kuralı önermiştir. Han ve Yang [54], 3B ayrık grafik gönderimine dayanan çalışmalarında, uçtan uca derin pekiştirmeli öğrenme çerçevesini önermişlerdir. Bu çerçevede, bir kodlayıcı ve bir kod çözücünden oluşan değiştirilmiş bir işaretçi ağı, seçilen çizelgeleme özelliklerine göre çizelgelenecek işlemleri kodlamak için benimsemişlerdir. Magalhaes vd. [55], derin  $Q$ -öğrenme kullanarak çift kaynak kısıtlı, esnek atölye tipi çizelgeleme problemini çözmek için dikkat mekanizmasına sahip bir kodlayıcı-kod çözücü sinir ağı mimarisi önermişlerdir. Bu özellik, ajanın belirli bir işlemi programda daha uygun bir konuma taşımasını ve tüm işlemlerin işlem süreleri gibi açık problem bilgilerini almasını mümkün kılar. Feng vd. [56], maksimum tamamlanma süresini amaç fonksiyonu olarak ele almış ve esnek atölye tipi çizelgeleme problemini çözmek için derin pekiştirmeli öğrenme algoritması önermişlerdir. Long vd. [57], yapay arı kolonisi algoritması ile yavaş yakınsama hızı ve yerel optimuma ulaşma problemlerini çözmek için, pekiştirmeli öğrenmeye dayalı geliştirilmiş kendi kendine öğrenen yapay arı kolonisi algoritması önermişlerdir. Li vd. [58], hedef olarak maksimum tamamlanma zamanı ve toplam makine iş yükünü optimize etmek için bulanık işlem süresine sahip, çok amaçlı bir esnek atölye tipi çizelgeleme problemi incelemişlerdir. Problemi çözmek için önerilen yöntem, üç kurallı bir başlangıç stratejisi kullanarak yüksek kaliteli bir başlangıç popülasyonu elde etmektedir. Lei vd. [59], büyük ölçekli, dinamik esnek atölye tipi çizelgeleme problemini, otomatik olarak farklı ölçeklere sahip bir dizi alt probleme bölmek için yüksek seviyeli bir katman tasarlamışlardır. Grafik sinir ağına dayalı bir katmanın politikası, bir iş operasyonunu programlamak için eğitilir ve çok katmanlı algılayıcıya dayalı başka bir politika, iş operasyonunu işlemesi için bir makine atamak üzere eğitilir. Chang vd. [60], dinamik esnek atölye tipi çizelgeleme problemini, erken ve geç kalma cezalarını en aza indirmek amacıyla rastgele iş gelişiyile çözmek için derin pekiştirmeli öğrenme önermişlerdir. Problemin ölçeğine göre yumuşak e-açgözlü davranış politikası tasarlamışlardır. Liu vd. [61], dinamik esnek atölye çizelgeleme problemini çözmek için hiyerarşik ve dağıtık bir mimari önermişlerdir. Çift derin  $Q$ -ağı algoritması, çizelgeleme ajanlarını eğitmek, üretim bilgileri ve çizelgeleme hedefleri arasındaki ilişkiyi yakalamak ve sürekli iş gelen esnek bir atölye için gerçek zamanlı çizelgeleme kararları almak için kullanmışlardır. Zhang vd. [62], esnek atölye tipi çizelgeleme için çoklu ajan grafikleriyle, derin pekiştirmeli öğrenmeye dayalı DeepMAG adlı bir model önermişlerdir. DeepMAG'in iki önemli katkısı vardır: (1) Derin pekiştirmeli öğrenme ve çok ajanlı pekiştirmeli öğrenmeyi arasındaki entegrasyon. (2) Bir ajan, bir işbirlikçi eylem gerçekleştirmek için komşu ajanlarıyla iş birliği yapmasıdır. Oh vd. [63], esnek atölye tipi çizelgeleme problemini, çok ajanlı olarak yüksek değişkenlikle modelleyerek, bağımsız öğrenenleri, örtük niceliksel ağ ile birleştiren çizelgeleme yöntemini önermişlerdir. Zeng vd. [64], minimum ön alan bilgisi ile ayrık grafikleri durumlar olarak ve bir dizi genel dağıtım kuralını eylem alanı olarak alan esnek bir hibrit çerçeve önermişlerdir. Dikkat mekanizması, durumların öznitelik çıkarımı için grafik temsili öğrenme modülü olarak kullanılır ve her bir durumu en uygun gönderme kuralına eşlemek için öncelikli yeniden oynatma ve gürültülü ağlara sahip çift düellolu derin  $Q$ -ağı kullanılır. Cunha vd. [65], makine öğrenimi tekniklerine dayalı atölye tipi çizelgeleme problemlerini çözmek için yenilikçi bir yaklaşım sunmuşlardır. Genel performanslarını iyileştirmek ve mevcut yaklaşımların sunduğu sınırlamaların üstesinden gelmek için pekiştirmeli öğrenmeyi, çizelgeleme

sistemlerine dahil eden yeni bir mimari sunmuşlardır. Khuntiyaporn vd. [66], çok amaçlı, esnek atölye tipi çizelgeleme problemini çözmeye odaklanmışlardır. Önerilen yöntem,  $Q$ -Öğrenme algoritmasına sahip bir pekiştirmeli modelidir. Du vd. [67], hem maksimum tamamlanma süresi hem de toplam elektrik fiyatını aynı anda optimize eden bir dağıtım algoritması ve derin  $Q$ -ağ tahminine yönelik, hibrit çok amaçlı bir optimizasyon algoritması önermişlerdir. İşlem sırası, makine ataması ve işleme hızı atamasının tümü, üç boyutlu bir çözüm gösterimi kullanılarak açıklanmıştır. Wang vd. [68], gerçekçi üretim ortamını simüle etmek için dinamik, çok amaçlı esnek atölye tipi çizelgeleme problemini ele almışlardır. İş ekleme, iş iptali, iş operasyonu değişikliği, makine ekleme, makine aleti değiştirme ve makine arızası dahil olmak üzere problemde altı dinamik olaya yer vermişlerdir. Araştırmalarında, en uzun iş işleme süresi (maksimum tamamlanma zamanı), ortalama makine kullanımı ve ortalama iş işleme gecikme oranı olmak üzere üç hedefi bir dizi kısıtla birlikte gerçekleştirmek için çözüm sunmuşlardır. Chen vd. [69], süreç kaynak öncelik senaryosu altında esnek atölye tip çizelgeleme problemi modellemiş ve minimum çizelgeleme süresi hedefe ulaşmak için derin pekiştirmeli öğrenmeye dayalı iki katmanlı bir kural çizelgeleme algoritması önermişlerdir. Luo vd. [70], yeni iş eklemeleri ve makine arızaları ile dinamik, kısmi beklemesiz çok amaçlı esnek atölye tipi çizelgeleme problemini incelemişler, hiyerarşik, çözümü için hiyerarşik, çok ajanlı, derin pekiştirmeli öğrenme tabanlı, gerçek zamanlı bir yöntem geliştirmişlerdir. Önerdikleri yöntem, farklı uzay-zamansal ölçeklerde çalışan, üç yakın politika optimizasyonu tabanlı ajan, yani objektif ajani, iş ajani ve makine ajani içerir. Zhou vd. [71], atölye tipi çizelgeleme problemleri için arama alanını azaltan bir OpenAI gym ortamı sunmuş ve çok ajanlı esnek atölye tipi problemler için performansla sahip sezgisel kılavuzlu bir  $Q$ -Öğrenme çözümü sunmuşlardır. Popper ve Ruskowski [72], esnek atölye tipi çizelgeleme problemlerini çözmek için derin pekiştirmeli öğrenmeye dayalı çok ajanlı bir sistem önermişlerdir.

#### 4.5 Diğer Çizelgeleme Problemleri

Park vd. [73], pekiştirmeli öğrenme tabanlı, gerçek zamanlı bir çizelgeleyici olan ScheduleNet'i önermişlerdir. ScheduleNet'in karar verme prosedürü, ajan-görev grafiği ile çizelgeleme probleminin durumunu temsil etmeyi, düğüm gömmelerini ve ajanlar ve görevler arasındaki önemli ilişkisel bilgileri kullanmayı içerir. Yang ve Xu [50], derin pekiştirmeli öğrenmeyi kullanarak dinamik iş gelişi ile zeki çizelgeleme ve yeniden yapılandırmayı incelemişler ve akıllı üretimde, akıllı çizelgeleme ve yeniden yapılandırma sistem mimarisini önermişlerdir. Tek makine çizelgeleme problemleri, paralel makine çizelgeleme problemleri, akış tipi çizelgeleme problemleri, atölye tipi çizelgeleme problemleri ve diğer çizelgeleme problemleri başlıkları altında yer verilen makalelere dair yazar, yıl ve makale içeriği bilgilerinin yer aldığı liste Tablo 1'de sunulmuştur. Üretim sistemlerinde, makine çizelgeleme süreçlerinde, ortam dinamik (makine arızaları, işlerin rassal gelişi vd.) olduğu için bu ortama uyum sağlayacak, öğrenebilen, çözüm yöntemlerinin kullanılması kaçınılmazdır. Tablo 1'de görüleceği gibi pekiştirmeli ve/veya derin pekiştirmeli öğrenme yöntemlerinin meta sezgisel algoritmalarla ya da diğer sezgiseller ile dinamik ortamlardaki makine çizelgeleme problemlerinde kullanılması, dinamik çizelge oluşturma imkânı sağlamaktadır. Endüstri 4.0 ve 5.0'e geçiş sürecinde, makine çizelgeleme problemlerinin çözümünde, öğrenme tabanlı yöntemler yardımı ile dinamik çizelgeler sunan metotlar kullanılmak zorundadır.

Tablo 1. Literatürde incelenen makalelerin sınıflandırılması

Tek Makine Çizelgeleme Problemleri			
Yazar	Yıl	Makale İçeriği	
1	Wang ve Usher [28]	2005	Ajan tabanlı, üretim çizelgeleme de $Q$ -öğrenme yöntemini, işlerin tek makineye sevk edilmesinde kullanmışlardır.
2	Wang vd. [29]	2021	Çok durumlu, tek makine sisteminde, esnek önleyici bakımı kullanarak proaktif bir şekilde, makine arızalarını azaltacak, üretim çizelgesi önermişlerdir.
3	Yang vd. [30]	2021	Tek makineli, çok durumlu üretim sisteminde, işleme ve bakım maliyetlerini birlikte dikkate alan pekiştirmeli bir öğrenme yöntemi önermişlerdir.
4	Deliktaş [31]	2022	Çok amaçlı, tek makine probleminde, öğrenme etkisini, serbest bırakma zamanı ile dikkate alan, kendini uyarlayan bir Memetik Algoritma geliştirmişlerdir.
Paralel Makine Çizelgeleme Problemleri			
Yazar	Yıl	Makale İçeriği	
1	Lopes vd. [32]	2019	Paralel makinelerde, rotalama ve çizelgeleme problemlerini çözmek için pekiştirmeli öğrenme tabanlı, çok ajanlı bir çerçeve sunmuşlardır.
2	Liangxun Guo [33]	2020	Dinamik, çok amaçlı, özdeş olmayan paralel makine çizelgeleme problemini, makine hızı farklılıkları, belirsiz görev varış zamanı metriklerine göre çok aşamalı, pekiştirmeli öğrenme yöntemi ile çözmüşlerdir.
3	Chien & Lan [34]	2021	Sıra bağımlı kurulum süresi ile bağımsız, paralel makine çizelgeleme problemini çözmek için derin pekiştirmeli öğrenme ve hibrit genetik algoritmayı birleştiren bir planlama ajanını eğitmek için derin $Q$ ağı önermişlerdir.
Akış Tipi Çizelgeleme Problemleri			
Yazar	Yıl	Makale İçeriği	
1	Arviv vd. [35]	2016	Akış tipi çizelgeleme problemlerinde, iki robot ile iş transferi için işbirlikçi pekiştirmeli öğrenme yöntemi önermişlerdir.
2	Wang ve Tang [36]	2017	Memetik algoritma tabanlı makine öğrenmesi yöntemini, çok amaçlı, permütasyon akış tipi çizelgeleme probleminin çözümü için geliştirmişlerdir.
3	Shao vd. [37]	2018	Aylak süresi olmayan, akış tipi çizelgeleme problemini, toplam gecikme kriteri ile çözmek için karma ayırık öğretme-öğrenme tabanlı meta-sezgisel algoritma önermişlerdir. Algoritma üç aşamadan oluşmuştur.
4	Han vd. [38]	2019	Hibrit akış tipi çizelgeleme problemini çözmek için pekiştirmeli öğrenme yöntemini geliştirmişlerdir. Bir otomobil motoru fabrikasının metal işleme atölyesinde, çizelgeleme sürecinde kullanmışlardır.

5	Zhao vd. [39]	2021	Maksimum montaj tamamlanma süresi minimizasyonu performans kriterini dikkate alan, dağıtılmış, montaj aylak süresi olmayan, akış tipi çizelgeleme problemini çözmek için, pekiştirmeli öğrenme metodunu kullanan, su dalgası optimizasyon algoritması önermişlerdir.
6	Chen vd. [40]	2021	Çoklu makine iş birliği olan akış tipi çizelgeleme problemini, maksimum tamamlama süresini ve toplam makine yükünü en aza indirme performans kriterini dikkate alarak çözmek için pekiştirmeli öğrenmeyi kullanan çok popülasyonlu gri kurt optimizasyon algoritmasını geliştirmişlerdir.
7	Pan vd. [41]	2021	Maksimum tamamlanma süresi minimizasyonu performans kriterli, permütasyon akış tipi çizelgeleme problemini çözmek için derin pekiştirmeli öğrenmeye dayalı bir optimizasyon algoritması önermişlerdir.
8	Cai vd. [42]	2022	Üretim, sevkiyat ve montaj süreçlerinden oluşan, dağıtılmış, montaj hibrit akış tipi çizelgeleme problemini çözmek için $Q$ -öğrenmeyi kullanan bir kurbağa algoritması geliştirmişlerdir.
9	Zhao vd. [43]	2022	Çok amaçlı, enerji etkinliği, dağıtılmış, montaj süreçli beklemesiz akış tipi çizelgeleme problemini çözmek için pekiştirmeli öğrenme güdümlü bir beyin fırtınası optimizasyon algoritması önermişlerdir.
10	Zhao vd. [44]	2022	Sıra bağımlı hazırlık sürelerine sahip, dağıtılmış, heterojen, beklemesiz akış tipi çizelgeleme problemini çözmek için $Q$ -öğrenmeyi kullanan yapay bir arı kolonisi algoritması geliştirmişlerdir.
11	Zhao, Jiang vd. [45]	2022	Sıra bağımlı hazırlık süresi olan enerji verimli, dağıtılmış, beklemesiz bir akış tipi çizelgeleme problemini, tamamlanma süresi ve toplam enerji tüketimi minimizasyonu performans kriterli olarak çözmek için $Q$ -öğrenmeye dayalı, işbirlikçi bir meta-sezgisel algoritma önermişlerdir.
12	Yan vd. [46]	2022	Dağıtılmış, permütasyon akış tipi çizelgeleme problemini, esnek periyodik önleyici bakım ile entegre şekilde çözmek için açgözlülük oranı azalan derin $Q$ ağ tabanlı bir yöntem geliştirmişlerdir.
13	Nahhas vd. [47]	2022	Gerçek yaşamdaki dört aşamalı bir hibrit akış tipi çizelgeleme problemini çözmek için ayrık olay simülasyon modeli ve derin pekiştirmeli öğrenme tekniklerine dayanan bir metod sunmuşlardır.
14	He vd. [48]	2022	Akış tipi çizelgeleme problemini, tamamlanma süresi performans kriterine göre çözmek için bir $Q$ -öğrenme algoritması önermişlerdir.
15	Dong vd. [49]	2022	Permütasyon akış tipi çizelgeleme problemini, geç iş minimizasyonu performans kriterli çözmek için derin pekiştirmeli bir öğrenme yaklaşımı geliştirmişlerdir.
16	Yang vd. [50]	2022	Gerçek zamanlı, akıllı karar verme için derin pekiştirmeli öğrenmeyi kullanan, dinamik iş gelişleriyle, dağıtılmış, permütasyon akış tipi çizelgeleme problemini çözmüşlerdir.
17	Ying & Lin [51]	2022	Dağıtılmış, montaj süreçli, permütasyon akış tipi çizelgeleme problemini, ilişkisiz montaj makinelerini ve sıra bağımlı kurulum sürelerini dikkate alacak şekilde çözmek için bir karma tamsayılı doğrusal programlama modeli ve pekiştirmeli öğrenmeyi kullanan yinelemeli açgözlü algoritma önermişlerdir.

			<b>Atölye Tipi Çizelgeleme Problemleri</b>
<b>Yazar</b>	<b>Yıl</b>	<b>Makale İçeriği</b>	
1	Gabel vd. [52]	2008	Atölye tipi çizelgeleme problemlerini, sıralı karar süreçleri olarak yorumlayıp, her kaynağa iş gönderme kararlarını, diğer ajanlardan bağımsız olarak veren ve bir pekiştirmeli öğrenme algoritması kullanarak davranışını geliştiren ajanlar önermişlerdir.

2	Luo [53]	2020	Araya yeni iş alınan, esnek atölye tip çizelgeleme problemlerini çözmek için derin pekiştirmeli öğrenme tabanlı dinamik çizelge sunmuşlardır.
3	Han & Yang [54]	2021	Esnek atölye tipi çizelgeleme problemlerinin çözümü için derin pekiştirmeli öğrenme tabanlı bir yöntem önermişlerdir.
4	Magalhaes vd. [55]	2021	Çift kaynak kısıtlı, esnek atölye tip çizelgeleme problemini çözmek için derin $Q$ -öğrenme kullanan bir kodlayıcı-kod çözücü sinir ağı mimarisi geliştirmişlerdir.
5	Feng vd. [56]	2021	Maksimum tamamlanma süresi performans kriterli, esnek atölye tipi çizelgeleme problemini çözmek için derin pekiştirmeli öğrenme algoritması önermişlerdir.
6	Long vd. [57]	2022	Esnek atölye tipi çizelgeleme problemini çözmek için pekiştirmeli öğrenmeye dayalı kendi kendine öğrenen yapay arı kolonisi algoritması önermişlerdir.
7	Li vd. [58]	2022	İki amaçlı (maksimum tamamlanma zamanı ve toplam makine iş yükünü optimize edilmesi), bulanık esnek akış tipi çizelgeleme problemini çözmek için üç kurallı bir başlangıç stratejisi kullanan, pekiştirmeli öğrenme tabanlı bir yöntem önermişlerdir.
8	Lei vd. [59]	2022	Büyük ölçekli, dinamik esnek atölye tipi çizelgeleme problemini, otomatik olarak farklı ölçeklere sahip bir dizi alt probleme bölmek için yüksek seviyeli bir katman tasarlamışlardır. Bu problem için uçtan uca hiyerarşik pekiştirmeli öğrenme yöntemi sunmuşlardır.
9	Chang vd. [60]	2022	Dinamik, esnek atölye tipi çizelgeleme problemini, erken ve geç kalma cezalarını en aza indirmek amacıyla rastgele iş gelişimiyle çözmek için derin pekiştirmeli öğrenme kullanmışlardır.
10	Liu vd. [61]	2022	Esnek atölye tipi çizelgeleme için derin pekiştirmeli öğrenme ile gerçek zamanlı çizelgeleme kararları alan dinamik bir çizelge sunmuşlardır.
11	Zhang vd. [62]	2022	Esnek atölye tipi çizelgeleme için çoklu ajan grafikleriyle, derin pekiştirmeli öğrenmeye dayalı, DeepMAG adlı bir model önermişlerdir.
12	Oh vd. [63]	2022	Yüksek değişkenli, esnek atölye tipi çizelgeleme probleminin çözümü için çok ajanlı, bağımsız öğrenenleri, örtük niceliksel ağ ile birleştiren bir yöntem önermişlerdir.
13	Zeng vd. [64]	2022	Dinamik atölye tipi çizelgeleme problemi için pekiştirmeli öğrenme ve dikkat mekanizmasını kullanan, hibrit zekâ tabanlı yöntem geliştirmişlerdir.
14	Cunha vd. [65]	2021	Atölye tipi çizelgeleme problemlerini çözmek için makine öğrenimi tekniklerine dayalı, pekiştirmeli öğrenme yaklaşımını sunmuşlardır.
15	Khuntiyaporn vd.[66]	2021	Çok amaçlı, esnek atölye tipi çizelgeleme problemini çözmek için $Q$ -Öğrenme algoritmasına sahip bir pekiştirmeli öğrenme modeli önermişlerdir.
16	Du vd. [67]	2022	Esnek atölye tipi çizelgeleme problemi için maksimum tamamlanma süresi ve toplam elektrik fiyatını aynı anda optimize eden bir dağıtım algoritmasını, derin $Q$ -ağ tahmini ile sunmuşlardır.
17	Wang vd. [68]	2022	Belirsiz durumların meydana geldiği (iş ekleme, iş iptali, iş operasyonu değişikliği, makine ekleme, makine aleti değiştirme ve makine arızası) dinamik, çok amaçlı, esnek atölye tip çizelgeleme problemini çözmek için çok amaçlı pekiştirmeli öğrenme yöntemini önermişlerdir.
18	Chen vd. [69]	2022	Süreç kaynak öncelik senaryosu altında, esnek atölye tipi çizelgeleme problemini modellemişler ve minimum çizelgeleme süresi hedefine ulaşmak için derin pekiştirmeli öğrenmeye dayalı iki katmanlı bir kural çizelgeleme algoritması geliştirmişlerdir.



19	Luo vd. [70]	2022	Yeni iş eklemeleri ve makine arızaları ile dinamik, kısmi beklemesiz, çok amaçlı, esnek atölye tipi çizelgeleme problemini çözmek için hiyerarşik, çok ajanlı, derin pekiştirmeli, öğrenme tabanlı, gerçek zamanlı, çizelgeleme yöntemi geliştirmişlerdir.
20	Zhou vd. [71]	2022	Çok ajanlı, atölye tipi çizelgeleme problemleri için pekiştirmeli $Q$ -Öğrenmeye dayalı bir çözüm sunmuşlardır.
21	Popper vd. [72]	2022	Esnek atölye tipi çizelgeleme problemlerini çözmek için derin pekiştirmeli öğrenmeye dayalı çok ajanlı bir sistem konsepti önermiştir.

<b>Diğer Çizelgeleme Problemleri</b>			
	<b>Yazar</b>	<b>Yıl</b>	<b>Makale İçeriği</b>
1	Park vd. [73]	2021	Pekiştirmeli öğrenme tabanlı gerçek zamanlı bir çizelgeleyici olan ScheduleNet'i önermiştir.
2	Kim & Lee [74]	1998	Heterojen makine çizelgeleme probleminin çözümü için genetik takviyeli öğrenme yaklaşımı önermişlerdir.
3	Kim vd. [75]	2020	Esnek bir akıllı üretim sisteminde dağıtılmış zeka için çok etmenli sistem ve güçlendirmeli öğrenme yaklaşımı geliştirmişlerdir.
4	Wang vd. [76]	2022	Bakım ile dağıtılmış, üç aşamalı, montaj çizelgeleme problemi için takviyeli öğrenmeye sahip, uyarlanabilir yapay arı kolonisi algoritması önermişlerdir.
5	Yang & Xu [50]	2022	Derin pekiştirmeli öğrenmeyi kullanarak dinamik iş gelişi ile zeki çizelgeleme ve yeniden yapılandırılmayı incelemiş ve akıllı üretimde akıllı çizelgeleme ve yeniden yapılandırma sistem mimarisi önermişlerdir.

#### 4. SONUÇ VE ÖNERİLER

Pekiştirmeli öğrenme yöntemleri son yıllarda iş çizelgeleme problemlerinin çözümünde aktif olarak kullanılmaktadır. Bu çalışmada, ilk defa pekiştirmeli öğrenme yöntemlerinin çizelgeleme problemlerinin çözümü ile ilgili literatürde yapılan çalışmalar araştırılmıştır. Bu amaçla “scholar.google.com” arama motoru yardımıyla “Reinforcement Learning and scheduling” anahtar kelimeleri kullanılarak, 2003 ile 2023 yılları aralığında tarama yapılmış, tek makine, paralel makine, akış tipi, atölye tipi ve diğer makine çizelgeleme ile doğrudan ilgili 50 adet makale tespit edilmiş ve çalışmalar detaylı olarak incelenmiştir. Yapılan araştırmalar, pekiştirmeli öğrenme algoritmalarının önceden elde edilen bilgileri kullanarak beklenmeyen durumlara hızla uyum sağladığını ve bu şekilde çizelgeleme sürecini hızlandırdığını ortaya koymuştur. Bu özellik, pekiştirmeli öğrenme algoritmalarının gerçek dünya problemlerine daha uygulanabilir hale gelmesini sağlar. Pekiştirmeli öğrenmenin diğer yöntemlere kıyasla avantajları ve güçlü yönleri bulunmaktadır. Literatürde değinilen bu yönleri aşağıda belirtilmiştir.

Pekiştirmeli öğrenme algoritmaları, simülasyonlarla tam bilgiye ihtiyaç duymadan iyi bir şekilde öğrenir ve hızla optimuma yakınsar. Bazı meta sezgisel yöntemler, statik çizelgeleme problemlerinde iyi sonuçlar verir. Ancak bu yöntemler, sistemdeki ani değişikliklere hızlı bir şekilde uyum sağlayamazlar çünkü sistem değiştiğinde tüm programı yeniden oluştururlar. Pekiştirmeli öğrenme algoritmaları ise daha önce elde edilen bilgileri kullanarak çizelgelemeye sıfırdan başlamazlar. Bu nedenle, pekiştirmeli öğrenme yaklaşımı, ani değişikliklere kolayca tepki verebilir ve programı daha hızlı bir şekilde değiştirebilir. Gerçek üretim sistemlerinde makine arızası, işlerin rastgele gelişi, sipariş iptali gibi dinamik durumlar bulunur. Pekiştirmeli öğrenme, bu dinamik sistemi çevreden gelen geribildirimlerle öğrenir ve bu bilgileri yeni ve görülmemiş durumlara genelleme yeteneğine sahiptir. Bu özellikler, pekiştirmeli öğrenmenin uyarlanabilir, esnek ve genişletilebilir bir yaklaşım olmasını sağlar. Bu nedenle, gerçek üretim sistemleri için uygun bir yöntem olabilir. Sevk kuralları, iş sıralama problemlerinde pratik ve kullanışlıdır. Ancak hangi gönderim kuralının hangi durumda uygulanacağına karar vermek uzmanlık gerektirir. Pekiştirmeli öğrenme algoritmaları, gönderim kurallarını seçmek için uygulandığında, sistem durumuna bağlı olarak otomatik karar verme sağlar, karar verme süresini azaltır ve tam otomatik üretim sistemlerinde daha uygulanabilir hale gelir. Ayrıca, pekiştirmeli öğrenme algoritmaları sistemdeki daha fazla özelliği dikkate alabilir, farklı gönderim kuralları arasında geçiş yapabilir ve bunları uygun şekilde birleştirebilir. Bu özellikler, çok amaçlı problemlerde, pekiştirmeli öğrenme algoritmalarına tek gönderim kuralı seçim politikalarına karşı üstünlük sağlar.

Pekiştirmeli öğrenmenin avantajları ve güçlü yönlerinin yanı sıra pratikte uygulanması bazı zorluklar içermektedir. Literatürde değinilen bu yönleri aşağıda belirtilmiştir. Karmaşık, gerçek dünya problemlerinde pekiştirmeli öğrenmenin uygulanması, geniş ve sürekli durum uzaylarıyla başa çıkma zorluğunu beraberinde getirir. Durum uzayının doğru tasarımı ve durum özelliklerinin doğru seçimi, algoritmanın etkinliği ve ölçeklenebilirliği açısından önemlidir. Ödül sisteminin tasarımında kısıtlamalar ve katı kurallar bulunmamaktadır. Ödül sisteminin doğru şekilde tasarlanması, çok sayıda seçenek olduğundan dolayı zor olabilir. Özellikle çok amaçlı problemlerde, ödülün, sisteminin hedeflerin önem derecesine göre tasarlanması önemlidir. Ödül işlevi, her hedefi uygun şekilde temsil etmeli ve dengelemelidir. Pekiştirmeli öğrenmenin ölçeklenebilirliği, problemin tasarımıyla doğrudan ilişkilidir. Tüm durumların  $Q$ -tablosunda temsil edilmesi yalnızca küçük ölçekli problemlerde uygulanabilir ve ölçeklendirme zorluğuna yol açar. Ayrıca, problemdeki iş veya makine sayısına özgü durumların tasarımı, daha büyük ölçekli problemlere ölçeklendirmeyi zorlaştırır. Durum toplama veya fonksiyon yaklaşımı gibi yöntemler, bu sorunu çözmek için kullanılır. Problemin tasarımı ve uygun fonksiyon veya durum toplama seçimi, durum-eylem uzayının boyutunu problemin boyutundan bağımsız hale getirir.

Akış tipi, birden fazla üretim aşamasından oluşur ve her aşamada, bir çizelgeleme problemi ortaya çıkarır. Akış tipi çizelgeleme problemlerinde, bu sorunu çözmek için aynı iş sırasını uygulamak gibi yaklaşım kullanılabilir. Ancak, bu yaklaşımlar, algoritmanın sonuçlarını olumsuz etkileyebilir ve her aşamaya

uygulanan algoritma, yüksek boyutlu durum uzayına yol açar. Atölye tip çizelgeleme problemlerinde ise iş tiplerinin farklı rotaları bulunur, bu da bazı işlerin ilk aşamada işlenemeyebileceği anlamına gelir. Bu durumda, her aşamada bir karar verme problemi ortaya çıkarır ve algoritmanın uygulanması zorlaşır, yüksek boyutlu durum uzayı problemiyle karşılaşılır.

Karar verme önemli bir faktör olduğunda, pekiştirmeli öğrenme algoritmasının açıklanabilir olması karar vericiler için önem kazanır. Ancak, birçok makine öğrenimi algoritmasında olduğu gibi, pekiştirmeli öğrenme algoritmasının da açıklanabilir olması eksikliği bulunmaktadır. Ajanın eylemlerinin nedeni karar vericilere açık bir şekilde açıklanamaz. Özellikle, fonksiyon kullanılan büyük boyutlu problemlerde, bir eylemin diğerine tercih edilme nedenini açıklamak karmaşık hale gelir. Pekiştirmeli öğrenme algoritmaları optimum sonuçları garanti etmez.

Gelecekteki araştırmalar için öneriler aşağıda sunulmuştur:

- İncelenen problemi daha gerçekçi hale getirmek için gerçek üretim ortamındaki çizelgeleme problemlerinde sıklıkla karşılaşılan makine arızası, iş iptali, sıraya bağlı kurulum süresi ve öncelik kısıtlamalarına daha fazla yer verilebilir.
- Pekiştirmeli öğrenme algoritmalarının, çok amaçlı problemlere uygulanması derinlemesine analiz edilmelidir, çünkü gerçek hayattaki çizelgeleme problemleri genellikle birden fazla amaç içerir.
- Meta sezgisel teknikler, makine çizelgeleme problemlerinde önemli bir rol oynar ve özellikle büyük ölçekli statik problemlerde iyi sonuçlar verir. Pekiştirmeli öğrenme algoritmalarının performansı, meta sezgisel yöntemlerle karşılaştırılabilir ve algoritmanın üstün ve zayıf yönleri ortaya çıkarılabilir.
- Atölye çizelgeleme problemleri genellikle tamamlama süresi ve geç kalma kriterleriyle ilgilenen çalışmalara odaklanmıştır. Akış süresiyle ilgili kriterlerin incelenmesi gelecekteki araştırmalar için önemli olabilir. Ayrıca, iş gecikmesinin önemi bazen müşteri anlaşmalarına göre değişebilir, bu durumda ağırlıklı geç kalma kriterinin minimizasyonu önemli bir konu haline gelir. Bu nedenle, pekiştirmeli öğrenme algoritmasının gerçek hayattaki çizelgeleme problemlerine uygulanması için bu alanlar üzerinde daha fazla araştırma yapılabilir.
- Pekiştirmeli öğrenme algoritmasının etkinliğini, karşılaştırmalı problemler üzerinde analiz eden çalışmalar, genellikle küçük ölçekli statik problemlerle sınırlıdır. Pekiştirmeli öğrenme algoritmasının büyük ölçekli problemlerdeki etkinliğini araştırmak için daha fazla analiz yapılması gerekmektedir.
- Pekiştirmeli öğrenmenin performansını önemli ölçüde etkileyen öğrenme algoritması çeşitliliği üzerine yapılan çalışmalar sınırlıdır. Farklı öğrenme algoritmalarının performanslarını karşılaştırmak ve analiz etmek literatüre katkı sağlayabilir.

## REFERANSLAR

- [1] Engin, O., Kahraman, C. & Yılmaz, M.K. (2009). A Scatter Search Method for Multiobjective Fuzzy Permutation Flow Shop Scheduling Problem: A Real-World Application. U.K. Chakraborty (Ed.): Computational Intelligence in Flow Shop and Job Shop Scheduling. SCI, 230, 169- 189. Springer-Verlag Berlin Heidelberg.
- [2] Engin, O., Yılmaz, M. K., Baysal, M. E & Sarucan, A. (2013). Solving Fuzzy Job Shop Scheduling Problems with Availability Constraints Using a Scatter Search Method. J. of Mult. -Valued Logic & Soft Computing, 21, 317- 334.
- [3] Engin, O., Özmete, A., İpek, S. & Karoğlu, Y.E. (2023). Çizelgeleme Problemlerinin Çözümünde Hibrit Biyocoğrafya Tabanlı Optimizasyon Algoritmasının Kullanımı. Harran Üniversitesi Mühendislik Dergisi, 8(1), 68-77. <https://doi.org/10.46578/humder.1256671>

- [4] Manzak, R., Engin, O. (2023). Akıllı Fabrikalarda Çizelgeleme Yöntemlerinin Analizi, Verimlilik Dergisi, 57, 4, 761- 774. <https://doi.org/10.51551/verimlilik.1136778>
- [5] Oppermann A. (2023). Self Learning AI-Agents Part I: Markov Decision Processes. [Erişim Tarihi: 01.11.2023] <https://towardsdatascience.com/self-learning-ai-agents-part-i-markov-decision-processes-baf6b8fc4c5f>
- [6] Thomas, G. (2009). Multi-Agent Reinforcement Learning Approaches for Distributed Job-Shop Scheduling Problems. Computer Science, 1-173.
- [7] Sutton, R. S., & Barto, A. G. (2015). Reinforcement Learning: An Introduction, Second edition, in progress, 1- 352, The MIT Press Cambridge, Massachusetts London, England.
- [8] Dietterich, T. G. (2000). Hierarchical Reinforcement Learning with the MAXQ Value Function Decomposition. Journal of Artificial Intelligence Research (C. 13).
- [9] Sutton, R. S., & Barto, A. G. (1998). Reinforcement learning: an introduction. MIT Press.
- [10] Wiering, M., Ch, M., Urgan, J. ", & Ch, S. J. (1998). Fast Online  $Q(\lambda)$ . Machine Learning (C. 33).
- [11] Kayhan, B. M., & Yildiz, G. (2023). Reinforcement learning applications to machine scheduling problems: a comprehensive literature review, Journal of Intelligent Manufacturing. 34, 905-929, Springer. <https://doi.org/10.1007/s10845-021-01847-3>
- [12] De Koning, M. C. T. C. (2020). Fleet Planning Under Demand Uncertainty A Reinforcement Learning Approach. <https://stmed.net/sites/default/files/airport-wallpapers-28369-9089125.jpg>.
- [13] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540), 529-533. <https://doi.org/10.1038/nature14236>
- [14] Li, Y. (2018). Deep Reinforcement Learning. <http://arxiv.org/abs/1810.06339>
- [15] Grondman, I., Busoniu, L., Lopes, G. A. D., & Babuška, R. (2012). A survey of actor-critic reinforcement learning: Standard and natural policy gradients. İçinde IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews (C. 42, Sayı 6, ss. 1291-1307). <https://doi.org/10.1109/TSMCC.2012.2218595>
- [16] Martínez Jiménez, Y. (2012). A Generic Multi-Agent Reinforcement Learning Approach for Scheduling Problems. VUBPRESS Brussels University Press. [www.vubpress.be](http://www.vubpress.be)
- [17] Başar, R., Engin, O. (2022). Beklemez Akış Tipi Çizelgeleme Problemlerinin Analizi ve Hibrit Dağınk Arama Yöntemi ile Çözümü, Çanakkale Onsekiz Mart University Journal of Advanced Research in Natural and Applied Sciences, 8 (2) 293- 308. <https://doi.org/10.28979/jarnas.936151>
- [18] Tanyaş, M., & Baskak, M. (2012). Üretim Planlama ve Kontrol. İrfan Yayıncılık.
- [19] Engin, O., Fığlalı, A. (2002). Akış Tipi Çizelgeleme Problemlerinin Genetik Algoritma Yardımı ile Çözümünde Uygun Çaprazlama Operatörünün Belirlenmesi. Doğu Üniversitesi Dergisi, 6, 27- 35.

- [20] Engin, O., Engin, B. (2018). Hybrid Flow Shop with Multiprocessor Task Scheduling Based on Earliness and Tardiness Penalties, *Journal of Enterprise Information Management*, 31, 6, 925- 936. <https://doi.org/10.1108/JEIM-04-2017-0051>
- [21] Engin, O., Günaydın, C. (2011). An Adaptive Learning Approach for No-Wait Flowshop Scheduling Problems to Minimize Makespan. *International Journal of Computational Intelligence Systems*, 4, 4, 521- 529.
- [22] Saç, İ., Engin, O. (2018). Bloklama Kısıtlı Akış Tipi Çizelgeleme Problemlerinin Maymun Arama Algoritması ile Çözümü. *Journal of Social and Humanities Science Research*, 5, 24, 1815- 1821.
- [23] Baysal, M. E., Sarucan, A., Büyüközkan, K. & Engin, O. (2022) Artificial Bee Colony Algorithm for Solving multi-objective Distributed Fuzzy Permutation Flow Shop Problem. *Journal of Intelligent & Fuzzy Systems*, 42, 439- 449. <https://doi.org/10.3233/JIFS-219202>
- [24] Külahlı, S., Engin, O., Koç, İ. (2021). A New Hybrid Scatter Search Method for Solving the Flexible Job Shop Scheduling Problems. *Celal Bayar University Journal of Science*, 17, 4, 347- 359. DOI: 10.18466/cbayarfbe.926756
- [25] Baysal, M. E., Durmaz, T., Sarucan, A., Engin, O. (2012). Açık Atölye Tipi Çizelgeleme Problemlerinin Paralel Kanguru Algoritması ile Çözümü. *Gazi Üniv. Müh. Mim. Fak. Der.* 27, 4, 855- 864.
- [26] Vollmann, T. E., Berry, W. L., Whybark, D. C., & Jacobs F.R. (2005). *Manufacturing Planning and Control for Supply Chain Management*. Mc Graw-Hill Book Companies Inc.
- [27] Kılıç, M. (2021). Bir Tekstil Firmasının Boyahane Bölümünde Paralel Makine Çizelgeleme Problemi İçin Bir Matematiksel Model Önerisi ve Farklı Çizelgeleme Kurallarının Karşılaştırılması. Necmettin Erbakan Üniversitesi, Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı, Konya
- [28] Wang, Y. C., & Usher, J. M. (2005). Application of reinforcement learning for agent-based production scheduling. *Engineering Applications of Artificial Intelligence*, 18(1), 73-82. <https://doi.org/10.1016/j.engappai.2004.08.018>
- [29] Wang, H., Yan, Q., & Zhang, S. (2021). Integrated scheduling and flexible maintenance in deteriorating multi-state single machine system using a reinforcement learning approach. *Advanced Engineering Informatics*, 49. <https://doi.org/10.1016/j.aei.2021.101339>
- [30] Yang, H., Li, W., & Wang, B. (2021). Joint optimization of preventive maintenance and production scheduling for multi-state production systems based on reinforcement learning. *Reliability Engineering and System Safety*, 214. <https://doi.org/10.1016/j.ress.2021.107713>
- [31] Deliktaş, D. (2022). Self-adaptive memetic algorithms for multi-objective single machine learning-effect scheduling problems with release times. *Flexible Services and Manufacturing Journal*, 34(3), 748-784. <https://doi.org/10.1007/s10696-021-09434-7>
- [32] Lopes Silva, M. A., de Souza, S. R., Freitas Souza, M. J., & Bazzan, A. L. C. (2019). A reinforcement learning-based multi-agent framework applied for solving routing and scheduling problems. *Expert Systems with Applications*, 131, 148-171. <https://doi.org/10.1016/j.eswa.2019.04.056>

- [33] Liangxun Guo, Z. Z. H. W. Q. (2020). Optimization of Dynamic Multi-Objective Non-Identical Parallel Machine Scheduling with Multi-Stage Reinforcement Learning. 2020 16th IEEE International Conference on Automation Science and Engineering (CASE). [https://doi.org/10.0/Linux-x86\\_64](https://doi.org/10.0/Linux-x86_64)
- [34] Chien, C. F., & Lan, Y. B. (2021). Agent-based approach integrating deep reinforcement learning and hybrid genetic algorithm for dynamic scheduling for Industry 3.5 smart production. *Computers and Industrial Engineering*, 162. <https://doi.org/10.1016/j.cie.2021.107782>
- [35] Arviv, K., Stern, H., & Edan, Y. (2016). Collaborative reinforcement learning for a two-robot job transfer flow-shop scheduling problem. *International Journal of Production Research*, 54(4), 1196-1209. <https://doi.org/10.1080/00207543.2015.1057297>
- [36] Wang, X., & Tang, L. (2017). A machine-learning based memetic algorithm for the multi-objective permutation flowshop scheduling problem. *Computers and Operations Research*, 79, 60-77. <https://doi.org/10.1016/j.cor.2016.10.003>
- [37] Shao, W., Pi, D., & Shao, Z. (2018). A hybrid discrete teaching-learning based meta-heuristic for solving no-idle flow shop scheduling problem with total tardiness criterion. *Computers and Operations Research*, 94, 89-105. <https://doi.org/10.1016/j.cor.2018.02.003>
- [38] Han, W., Guo, F., & Su, X. (2019). A reinforcement learning method for a hybrid flow-shop scheduling problem. *Algorithms*, 12(11). <https://doi.org/10.3390/a12110222>
- [39] Zhao, F., Zhang, L., Cao, J., & Tang, J. (2021). A cooperative water wave optimization algorithm with reinforcement learning for the distributed assembly no-idle flowshop scheduling problem. *Computers and Industrial Engineering*, 153. <https://doi.org/10.1016/j.cie.2020.107082>
- [40] Chen, R., Yang, B., Li, S., Wang, S., & Cheng, Q. (2021). An Effective Multi-population Grey Wolf Optimizer based on Reinforcement Learning for Flow Shop Scheduling Problem with Multi-machine Collaboration. *Computers and Industrial Engineering*, 162. <https://doi.org/10.1016/j.cie.2021.107738>
- [41] Pan, Z., Wang, L., Wang, J., & Lu, J. (2021). Deep Reinforcement Learning Based Optimization Algorithm for Permutation Flow-Shop Scheduling. *IEEE Transactions on Emerging Topics in Computational Intelligence*. <https://doi.org/10.1109/TETCI.2021.3098354>
- [42] Cai, J., Lei, D., Wang, J., & Wang, L. (2022). A novel shuffled frog-leaping algorithm with reinforcement learning for distributed assembly hybrid flow shop scheduling. *International Journal of Production Research*, 1-19. <https://doi.org/10.1080/00207543.2022.2031331>
- [43] Zhao, F., Hu, X., Wang, L., Xu, T., Zhu, N., & Jonrinaldi. (2022). A reinforcement learning-driven brain storm optimisation algorithm for multi-objective energy-efficient distributed assembly no-wait flow shop scheduling problem. *International Journal of Production Research*. <https://doi.org/10.1080/00207543.2022.2070786>
- [44] Zhao, F., Wang, Z., & Wang, L. (2022). A Reinforcement Learning Driven Artificial Bee Colony Algorithm for Distributed Heterogeneous No-Wait Flowshop Scheduling Problem With Sequence-Dependent Setup Times. *IEEE Transactions on Automation Science and Engineering*, 1-16. <https://doi.org/10.1109/tase.2022.3212786>

- [45] Zhao, F., Jiang, T., & Wang, L. (2022). A Reinforcement Learning Driven Cooperative Meta-Heuristic Algorithm for Energy-Efficient Distributed No-Wait Flow-Shop Scheduling with Sequence-Dependent Setup Time. *IEEE Transactions on Industrial Informatics*, 1-12. <https://doi.org/10.1109/tii.2022.3218645>
- [46] Yan, Q., Wu, W., & Wang, H. (2022). Deep Reinforcement Learning for Distributed Flow Shop Scheduling with Flexible Maintenance. *Machines*, 10(3). <https://doi.org/10.3390/machines10030210>
- [47] Nahhas, A., Kharitonov, A., & Turowski, K. (2022). Deep Reinforcement Learning Techniques for Solving Hybrid Flow Shop Scheduling Problems: Proximal Policy Optimization (PPO) and Asynchronous Advantage Actor-Critic (A3C). <https://hdl.handle.net/10125/79538>
- [48] He, Z., Wang, K., Li, H., Song, H., Lin, Z., Gao, K., & Sadollah, A. (2022). Improved Q-learning algorithm for solving permutation flow shop scheduling problems. *IET Collaborative Intelligent Manufacturing*, 4(1), 35-44. <https://doi.org/10.1049/cim2.12042>
- [49] Dong, Z., Ren, T., Weng, J., Qi, F., & Wang, X. (2022). Minimizing the Late Work of the Flow Shop Scheduling Problem with a Deep Reinforcement Learning Based Approach. *Applied Sciences (Switzerland)*, 12(5). <https://doi.org/10.3390/app12052366>
- [50] Yang, S., & Xu, Z. (2022). Intelligent scheduling and reconfiguration via deep reinforcement learning in smart manufacturing. *International Journal of Production Research*, 60(16), 4936-4953. <https://doi.org/10.1080/00207543.2021.1943037>
- [51] Ying, K. C., & Lin, S. W. (2022). Reinforcement learning iterated greedy algorithm for distributed assembly permutation flowshop scheduling problems. *Journal of Ambient Intelligence and Humanized Computing*. <https://doi.org/10.1007/s12652-022-04392-w>
- [52] Gabel, T., & Riedmiller, M. (2008). Adaptive Reactive Job Shop Scheduling with Reinforcement Learning Agents. *International Journal of Information Technology and Intelligent Computing*.
- [53] Luo, S. (2020). Dynamic scheduling for flexible job shop with new job insertions by deep reinforcement learning. *Applied Soft Computing Journal*, 91. <https://doi.org/10.1016/j.asoc.2020.106208>
- [54] Han, B. A., & Yang, J. J. (2021). A deep reinforcement learning based solution for flexible job shop scheduling problem. *International Journal of Simulation Modelling*, 20(2), 375-386. <https://doi.org/10.2507/IJSIMM20-2-CO7>
- [55] Magalhaes, R., Martins, M., Vieira, S., Santos, F., & Sousa, J. (2021). Encoder-Decoder Neural Network Architecture for solving Job Shop Scheduling Problems using Reinforcement Learning. 2021 IEEE Symposium Series on Computational Intelligence, SSCI 2021 - Proceedings. <https://doi.org/10.1109/SSCI50451.2021.9659849>
- [56] Feng, Y., Zhang, L., Yang, Z., Guo, Y., & Yang, D. (2021). Flexible Job Shop Scheduling Based on Deep Reinforcement Learning. *Proceedings of 2021 5th Asian Conference on Artificial Intelligence Technology, ACAIT 2021*, 660-666. <https://doi.org/10.1109/ACAIT53529.2021.9731322>
- [57] Long, X., Zhang, J., Qi, X., Xu, W., Jin, T., & Zhou, K. (2022). A self-learning artificial bee colony algorithm based on reinforcement learning for a flexible job-shop scheduling problem. *Concurrency and Computation: Practice and Experience*, 34(4). <https://doi.org/10.1002/cpe.6658>

- [58] Li, R., Gong, W., & Lu, C. (2022). A reinforcement learning based RMOEA/D for bi-objective fuzzy flexible job shop scheduling. *Expert Systems with Applications*, 203. <https://doi.org/10.1016/j.eswa.2022.117380>
- [59] Lei, K., Guo, P., Wang, Y., Xiong, J., & Zhao, W. (2022). An End-to-end Hierarchical Reinforcement Learning Framework for Large-scale Dynamic Flexible Job-shop Scheduling Problem. *Proceedings of the International Joint Conference on Neural Networks*, 2022-July. <https://doi.org/10.1109/IJCNN55064.2022.9892005>
- [60] Chang, J., Yu, D., Hu, Y., He, W., & Yu, H. (2022). Deep Reinforcement Learning for Dynamic Flexible Job Shop Scheduling with Random Job Arrival. *Processes*, 10(4). <https://doi.org/10.3390/pr10040760>
- [61] Liu, R., Piplani, R., & Toro, C. (2022). Deep reinforcement learning for dynamic scheduling of a flexible job shop. *International Journal of Production Research*, 60(13), 4049-4069. <https://doi.org/10.1080/00207543.2022.2058432>
- [62] Zhang, J.-D., He, Z., Chan, W.-H., & Chow, C.-Y. (2022). DeepMAG: Deep reinforcement learning with multi-agent graphs for flexible job shop scheduling. *Knowledge-Based Systems*, 110083. <https://doi.org/10.1016/j.knosys.2022.110083>
- [63] Oh, S. H., Cho, Y. I., & Woo, J. H. (2022). Distributional reinforcement learning with the independent learners for flexible job shop scheduling problem with high variability. *Journal of Computational Design and Engineering*, 9(4), 1157-1174. <https://doi.org/10.1093/jcde/qwac044>
- [64] Zeng, Y., Liao, Z., Dai, Y., Wang, R., Li, X., & Yuan, B. (2022). Hybrid intelligence for dynamic job-shop scheduling with deep reinforcement learning and attention mechanism. <http://arxiv.org/abs/2201.00548>
- [65] Cunha, B., Madureira, A., Fonseca, B., & Matos, J. (2021). Intelligent scheduling with reinforcement learning. *Applied Sciences (Switzerland)*, 11(8). <https://doi.org/10.3390/app11083710>
- [66] Khuntiyaporn, T., Songmuang, P., & Limprasert, W. (2021). The Multiple Objectives Flexible Jobshop Scheduling Using Reinforcement Learning. *16th International Joint Symposium on Artificial Intelligence and Natural Language Processing, iSAI-NLP 2021*. <https://doi.org/10.1109/iSAI-NLP54397.2021.9678152>
- [67] Du, Y., Li, J. qing, Chen, X. long, Duan, P. yong, & Pan, Q. ke. (2022). Knowledge-Based Reinforcement Learning and Estimation of Distribution Algorithm for Flexible Job Shop Scheduling Problem. *IEEE Transactions on Emerging Topics in Computational Intelligence*. <https://doi.org/10.1109/TETCI.2022.3145706>
- [68] Wang, H., Cheng, J., Liu, C., Zhang, Y., Hu, S., & Chen, L. (2022). Multi-objective reinforcement learning framework for dynamic flexible job shop scheduling problem with uncertain events. *Applied Soft Computing*, 109717. <https://doi.org/10.1016/j.asoc.2022.109717>
- [69] Chen, Z., Zhang, L., Wang, X., & Gu, P. (2022). Optimal Design of Flexible Job Shop Scheduling Under Resource Preemption Based on Deep Reinforcement Learning. *Complex System Modeling and Simulation*, 2(2), 174-185. <https://doi.org/10.23919/csms.2022.0007>
- [70] Luo, S., Zhang, L., & Fan, Y. (2022). Real-Time Scheduling for Dynamic Partial-No-Wait Multiobjective Flexible Job Shop by Deep Reinforcement Learning. *IEEE Transactions on Automation Science and Engineering*, 19(4), 3020-3038. <https://doi.org/10.1109/TASE.2021.3104716>



- [71] Zhou, H., Gu, B., & Jin, C. (2022). Reinforcement Learning Approach for Multi-Agent Flexible Scheduling Problems. <http://arxiv.org/abs/2210.03674>
- [72] Popper, J., & Ruskowski, M. (2022). Using Multi-Agent Deep Reinforcement Learning For Flexible Job Shop Scheduling Problems. *Procedia CIRP*, 112, 63-67. <https://doi.org/10.1016/j.procir.2022.09.039>
- [73] Park, J., Bakhtiyar, S., & Park, J. (2021). ScheduleNet: Learn to solve multi-agent scheduling problems with reinforcement learning. <http://arxiv.org/abs/2106.03051>
- [74] Kim, G.H., Lee, C.S.G. (1998). Genetic Reinforcement Learning Approach To The Heterogeneous Machine Scheduling Problem, *IEEE Transactions On Robotics And Automation*, 14, 6, 879- 893.
- [75] Kim, Y. G., Lee, S., Son, J., Bae, H., & Chung, B. Do. (2020). Multi-agent system and reinforcement learning approach for distributed intelligence in a flexible smart manufacturing system. *Journal of Manufacturing Systems*, 57, 440-450. <https://doi.org/10.1016/j.jmsy.2020.11.004>
- [76] Wang, J., Lei, D., & Cai, J. (2022). An adaptive artificial bee colony with reinforcement learning for distributed three-stage assembly scheduling with maintenance. *Applied Soft Computing*, 117. <https://doi.org/10.1016/j.asoc.2021.108371>