

Çok Kollu Haydutlar ile Dinamik Ambulans Konumlandırma

Dynamic Ambulance Redeployment via Multi-armed Bandits

Ümitcan ŞAHİN^{1,2}, Veysel YÜCESOY¹

¹Yapay Zeka ve Bilişim Tekn. Arş. Prg. Mdl., Aselsan Araştırma Merkezi, Ankara 06370, Türkiye

²Elektrik ve Elektronik Mühendisliği Bölümü, İ.D. Bilkent Üniversitesi, Ankara 06800, Türkiye
{ucshin,vyucesoy}@aselsan.com.tr

Özetçe —Bir ülkenin acil yardım sistemlerinin iyileştirilmesi, daha çok acil vakaya zamanında müdahale edilmesi ve daha çok hayatın kurtarılmasını sağlar. 112 Acil Yardım sisteminin bir parçası olan ambulans konumlandırma problemi, ambulansların vakalara mümkün olan en kısa sürede ulaşmasını sağlayacak şekilde konumlandırılmasını sağlayan birçok yöntemden oluşur. Bu çalışmada ambulanslar, literatürdeki yöntemlerin aksine, bir çok kollu haydut (ÇKH) algoritması kullanılarak konumlandırılmaktadır. OpenStreetMap (OSM) harita uygulaması kullanılarak oluşturulmuş iki yönlü kenarlardan ve toplam 2400 düğümden oluşan bir Ankara şehri haritası üzerinde konumlandırma işlemi yapılmaktadır. Düğümler üzerindeki vaka dağılımları ve aralarındaki seyahat süreleri ÇKH algoritması tarafından bilinmemektedir ve zamanla öğrenilmektedir. Bu öğrenim keşif ve istifade arasındaki ödünleşim sistemi ile sağlanmaktadır. Algoritma karşılaştırmaları için literatürde sıkça kullanılan ve dinamik bir konumlandırma yöntemi olan DMEXCLP modeli kullanılmıştır. Simülasyonlarda algoritma karşılaştırmaları için iki ölçüt değerlendirilmiştir: 1) vakalara ortalama müdahale süresi ve 2) 15 dakika altında varılan vaka yüzdesi. Sonuç olarak aynı şartlar altında önerilen ÇKH algoritmasının DMEXCLP modeline göre bu iki ölçüt açısından daha iyi performans verdiği gösterilmiştir.

Anahtar Kelimeler—Çok kollu haydut problemleri; ambulans konumlandırma.

Abstract—Improving a country's emergency medical services results in serving more calls on time and saving more lives in return. The ambulance redeployment problem, which is a part of the 112 Emergency Medical system in Turkey, consists of many methods that aim to redeploy ambulances in a way to minimize arrival times to calls. In this study, unlike many methods in the redeployment literature, ambulances are redeployed by a multi-armed bandit (MAB) algorithm. Using OpenStreetMap (OSM), a graph model that consists of 2400 nodes and bi-directional edges is constructed as a simplified map of Ankara for ambulance redeployment. Call distributions and travel times between the nodes are not known by the MAB algorithm beforehand and learned on the way. This learning process takes place via a mechanism called exploration and exploitation. The MAB algorithm is compared against a well-known dynamic redeployment optimization model called DMEXCLP. Two criteria are considered when comparing the performance of the algorithms during simulation: 1) the average arrival times and 2) the percentage of calls responded under 15 minutes. In conclusion, it is shown that under the same conditions the MAB algorithm performs better than the

DMEXCLP model in terms of the two criteria.

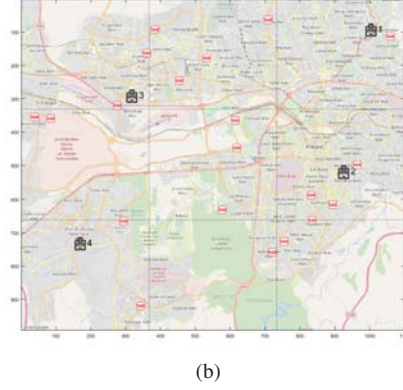
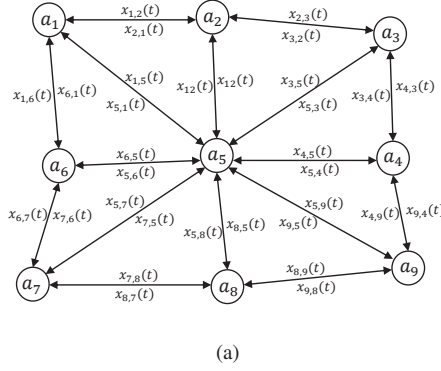
Keywords—Multi-armed bandit problems; ambulance redeployment.

I. GİRİŞ

112 Acil Yardım sisteminin amaçlarından biri, acil vakalarda ambulansların olay yerine mümkün olan en kısa sürede ulaşmasını sağlamaktır. Bu amaç için elde olan kaynakların etkili bir şekilde kullanılması gerekmektedir. Ambulans konumlandırma problemi, sınırlı sayıda ambulansın vakalara en kısa sürede müdahale edecek şekilde konumlandırılması anlamına gelmektedir.

Ambulans konumlandırma probleminin çözüm yöntemleri, statik, dinamik ve öğrenme tabanlı olmak üzere üçe ayrılabilir. Statik çözüm yöntemlerinde vaka dağılımları istatistiksel olarak durağan kabul edildiği için ambulanslar, yerleri zamanla değişmeyen şekilde konumlandırılmaktadır. Bu yöntemlerin çıkış noktasında konum kümesi kapsama modeli, en çok konumu kapsama modeli ve en çok beklenen konumu kapsama modeli (MEXCLP) vardır [1],[2]. Statik yöntemlerin eksik kaldığı en önemli nokta, vakalara müdahale eden ambulanslar tarafından açık bırakılan alanlar için bir önlem alınmamasıdır. Bu da müsait ambulans sayısının yeterince verimli kullanılmamasına neden olmaktadır. Bu sorunu çözebilmek için gün içinde değişen müsait ambulans sayısı, vaka dağılımları ve trafik durumunu göz önüne alarak ambulans konumlandırma yapan dinamik yöntemler önerilmiştir. Bu yöntemler için [3-13] incelenebilir.

Çok kollu haydut problemleri, vaka dağılımları ve trafik durumu gibi olasılıksal ifade edilebilen ama özellikleri tam olarak bilinmeyen parametrelerin olduğu durumlarda, keşif ve istifade mekanizması ile öğrenme yapılmasını inceler. Bu mekanizma slot makinelerini oynayarak kazancını en çoklamak isteyen bir kumarbaz örneği ile açıklanabilir. Kumarbazın amacı hangi makineleri, hangi sırada ve ne kadar oynayacağına karar vermektir. Bunun için ilk önce hangi makinelerin iyi ödül verdiğini keşfetmeli, aynı zamanda da iyi ödül verdiğini düşündüğü makineleri oynayarak istifade etmelidir. Eğer keşfe çok fazla para harcarsa kazancını hiçbir zaman arttıramaz;



Şekil 1. (a) 9 düğümden oluşan ambulans konumlandırma ağı: ambulans yerleşim lokasyonu a ve düğüm a_i 'den a_j 'ye giden kenardaki trafik yoğunluğunu gösteren trafik indeksi $x_{i,j}(t)$ 'den oluşan yönlü graf modeli, (b) OpenStreetMap ile oluşturulmuş Ankara haritası üzerinde gösterilen 4 hastane ve 20 ambulans konumu

eğer istifadeye çok fazla para ayırırsa (bir başka deyişle yeterince keşif yapmazsa) en çok ödül veren makineyi hiçbir zaman bulamayabilir. Bu iki kavram arasındaki ödünleşimi inceleyen literatürdeki önemli çalışmalar için [14-17] incelenebilir. Sağlık ve tavsiye sistemleri gibi birçok alanda uygulaması olan çok kollu haydut problemi çalışmaları için de [19-21] incelenebilir.

Ambulans konumlandırma problemini çok kollu haydut problemi olarak modelleyen iki çalışma yapılmıştır [22], [23]. Bu çalışmalardan ilki trafiğin durumunu rastsal bir bağlam olarak modellemiş ve kullanılan bağlamsal çok kollu haydut algoritmasının zaman geçtikçe öğrenme yaparak ambulansları başarılı bir şekilde konumlandığını göstermiştir. Çalışmalardan ikincisi ise vakalara ortalama varış süresini enazlama hedefinin yanında, varış sürelerinin standart sapmasını da enazlayan bir risk metriği kullanmıştır. Bu risk metriği ile ortalama vakalara varış süresindeki artış pahasına, 10 ve 15 dakika altında gidilecek vaka sayısında artış sağlanabileceğini göstermiştir.

Bu çalışmada önceki çalışmalardan farklı olarak haydut algoritmaları, ambulans konumlandırma literatüründe çok iyi bilinen dinamik MEXCLP (DMEXCLP) yöntemiyle karşılaştırılmıştır. Ayrıca açık kaynaklı OpenStreetMap (OSM) harita uygulamasından alınan konum verileriyle bir Ankara haritası oluşturulmuş ve algoritmalar bu gerçekçi harita üzerinde yürütülmüştür. Çalışmanın gerçek uygulamaya yakınsayabilmesi için vaka dağılımları ve yollar üzerindeki seyahat süreleri algoritmalar tarafından bilinmemektedir. Optimizasyona dayalı DMEXCLP yönteminin aksine çok kollu haydut algoritmaları, gözlemlenen vakaları ve seyahat sürelerini de kullanarak en iyi ambulans konumlarını zamanla öğrenmektedir.

Makalenin planlaması şu şekildedir: II. Bölümde ambulans konumlandırma probleminin nasıl çok kollu haydut problemi olarak modellendiği anlatılmıştır ve bu problemi çözen haydut algoritması verilmiştir. III. Bölümde karşılaştırma yapılan DMEXCLP modelinden bahsedilmiş, simülatör girdileri verilmiş ve karşılaştırma sonuçları paylaşılmıştır. IV. Bölümde elde edilen test sonuçları değerlendirilmiştir.

II. PROBLEM TANIMI

Şekil 1(a)'da konumlandırma için kullanılan 9 düğüm-den oluşan örnek bir yönlü graf modeli gösterilmiştir. Bu

modelde düğümler a_i , $i = \{1, \dots, K\}$ ile gösterilmektedir. OSM uygulaması kullanılarak Şekil 1(b)'de gösterilen Ankara haritası oluşturulmuştur. Oluşturulan haritadan algoritmaların hızlı çalışması için rastsal bir şekilde Şekil 1(a)'da gösterilen graf özelliklerini sağlayan $K = 2400$ adet düğüm seçilmiştir. Bu düğümler birbirlerine tam bağlantılıdır. Haritadaki yolların uzunlukları gerçek uzunluklarla aynıdır. Simülasyonlarda kullanılan 4 hastane konumu da haritada gösterilmiştir. Bütün hastane, vaka ve ambulanslar bu 2400 düğüm üzerinde konumlandırılmaktadır.

$x_{i,j}(t)$ ¹ düğüm a_i 'den düğüm a_j 'ye giden yoldaki t anındaki trafik yoğunluğunu göstermektedir. $x_{i,j}(t) = 1$ iken yolda trafik yoktur ve ambulanslar maksimum hızlarıyla hareket edebilmektedir. $x_{i,j}(t) = 0$ ise düğüm a_i 'den a_j 'ye t anında doğrudan bir yol olmadığı göstermektedir. $x_{i,j}(t)$, t 'ye bağlı olarak değişmektedir ve böylece yollar üzerinde zamana bağlı değişen seyahat süreleri oluşturulmuştur. Bu sayede [13] çalışmasından farklı olarak gün içinde saatlere ve trafiğe göre ambulansların seyahat sürelerini değiştiren gerçekçi bir simülasyon ortamı hazırlanmıştır. Bu çalışmadaki trafik yoğunlukları, [22] çalışmasındaki anlatılan Markov trafik modeli kullanılarak üretilmiştir.

Ambulans konumlandırma için geliştirilmiş, çok seçimli ambulans konumlandırma yapmaya olanak sağlayan haydut algoritması ÇS-LinUCB, Algoritma 1'de verilmiştir. ÇS-LinUCB, [18]'deki LinUCB algoritmasının her turda çok kollu seçime olanak sağlayacak şekilde uyarlanmış halidir. Algoritmanın detayları için [18] çalışması incelenebilir. Geliştirilen algoritma yollar üzerindeki trafik bilgisi X_{t,a_t} 'yi bağlam bilgisi olarak kullanmakta ve t anına kadar gözlemlendiği vaka düğümleri c_t , ambulansların vakaya çıkış noktaları a_t ve varış süresi ödülleri r_{t,a_t} 'yi kullanarak A ve b parametrelerini güncellemektedir. Daha sonra bu parametreler kullanılarak düğüm ödüllерinin beklenen değerleriyle bağlam vektörü (trafik bilgisi) arasında doğrusallık ilişkisi kuran $\hat{\theta}$ parametresi kestirimi yapılmaktadır (7. adım). Kestirilen parametre zaman ilerledikçe giderek gerçek θ değerine yakınsamakta ve böylece öğrenme yapılmaktadır.

¹Bildiri boyunca düğüm a_i ve bütün düğümler arasındaki t anındaki trafik indeksleri, K boyutlu X_{t,a_i} trafik vektörü ile gösterilmiştir. Yani $x_{i,j}(t) \in X_{t,a_i}$, $\forall a_j$.

Algoritma 1 Çok seçimli haydut algoritması: ÇS-LinUCB

```
1: Girdiler:  $\pi(\hat{n}_{amb}, K, \mathcal{H}_t, A, b)$ 
2: Çıktılar:  $I_t$  ( $t + 1$  turu için yeni ambulans konumları)
3: for Her yeni  $\{c_t, a_t, r_{t,a_t}, X_{t,a_t}\} \in \mathcal{H}_t$  için do
4:    $A = A + X_{t,a_t}^T X_{t,a_t}$ .
5:    $b = b + X_{t,a_t}^T r_{t,a_t}$ .
6: end for
7:  $\hat{\theta} = A^{-1}b$ . (doğrusal ilişki parametresi kestirimi)
8:  $t$  turundaki bütün konumlandırma noktaları  $a_t$ 'lerin bağ-
   lamlarını içeren bağlam matrisi  $X_t$ 'yi oluştur.
9:  $h_t = X_t^T \hat{\theta} + \text{diag}(\sqrt{X_t^T A^{-1} X_t})$ .
10: En büyük  $h_t$  terimine sahip ilk  $\hat{n}_{amb}$  tane konuma ambu-
   lans konumlandırma yap  $\rightarrow I_t$ .
```

III. TESTLER

Simülasyonlarda modellenen Ankara şehri için kullanılan parametreler Tablo I'de verilmiştir. Vakalar $\lambda = 1/10$ (dak.) ile Poisson dağılımına göre gerçekleşmektedir. Şekil 1(b)'de gösterildiği gibi harita 9 eşit bölgeye ayrılmıştır ve her bölge sırasıyla C_1, C_2, \dots, C_9 Poisson ikiterimli dağılımına göre vaka çıkarmaktadır ($C_1 + C_2 + \dots + C_9 = 1/10$). Eğer bölge i 'de C_i 'ye göre bir vaka gerçekleşiyorsa, $K = 2400$ tane düğümden bu bölgede yer alan düğümler arasından eşit olasılıklı bir örnek (uniform) dağılım kullanılarak biri seçilmektedir ve vaka bu düğümden gerçekleşmektedir. Algoritma 2'nin 3. ve 4. satırlarında gözüktüğü gibi toplam 100 simülasyon koşusu yapılmaktadır ve her koşuda bölge dağılım parametrelerine rastsal bir şekilde $C_1 + C_2 + \dots + C_9 = 1/10$ sağlayacak yeni değerler atanmaktadır. Algoritmalar 10 günlük simülasyon süresine karşılık gelecek şekilde yürütülmüştür. 9. satırda görüldüğü gibi turlar dakikalara karşılık gelmektedir ve toplamda 14400 tur vardır. Algoritma testlerinin sonuçları 100 simülasyon koşusunun ortalaması olarak hesaplanmakta ve her bir koşu 10 günlük simülasyon süresine denk gelmektedir. ÇS-LinUCB ve DMEXCLP algoritmaları C_1, C_2, \dots, C_9 parametrelerini bilmemektedir.

Ambulansların seyahat süreleri, yol uzunlukları ve bu yollar üzerindeki anlık trafik yoğunluklarına $x_{i,j}(t)$, $i, j = \{1, \dots, K\}$ göre belirlenmektedir. Yollar üzerindeki maksimum hız 100 km/s olarak alınmıştır. Ambulans yol boyunca hareket ederken trafik değerleri de değişebileceği için 19. satırda görüldüğü gibi ambulanslar vakalara ulaştığı anda tarih kümesi güncellenmektedir. Bir vakaya müdahale edecek ambulans belirlenirken o an içinde bulunulan t anına göre seyahat süresi en kısa ambulans seçilmektedir. Aynı şekilde bir ambulans hastaneye giderken t anına göre seyahat süresi en kısa mesafedeki hastane seçilmektedir.

Harita için toplam 4 adet hastane kullanılmıştır ve hastanelerin yerleri Şekil 1(b)'de gösterilmiştir. Simülasyon gerçekliğini arttırmak için 20. ve 23. satırlarda gözüktüğü gibi vaka olay yerinde ve hastanede bekleme süreleri de değerlendirilmiştir. Ayrıca ambulanslar hastaneden çıkana kadar müsait değildir. Hiç müsait ambulans kalmadığı durumda oluşan yeni bir vaka, bekleme kuyruğunun sonuna eklenmektedir. Ambulanslar müsait oldukça kuyruğun en başından başlayarak vakalara müdahale etmektedir ve arada geçen kuyrukta bekleme süreleri varış sürelerine eklenmektedir.

Kıyas amacıyla kullanılan algoritma dinamik konumlandırma yapan ve [13]'te verilen DMEXCLP modelidir. Bu

Algoritma 2 Konumlandırma simülasyonu

```
1: Girdi:  $K = 2400$ 
2: for Toplam ambulans sayısı
    $n_{amb} = \{5, 10, 15, 20\}$  için do
3:   for Simülasyon koşusu  $s = \{1, 2, \dots, 100\}$  do
4:     Vaka dağılımları  $C_1, \dots, C_9$  değerlerini yeniden belirle.
5:     Tarih kümesi:  $\mathcal{H}_0 \leftarrow \emptyset$ .
6:      $A \leftarrow I_K$ . ( $I_K$ ,  $K \times K$  boyutlu birim matrisidir)
7:      $b \leftarrow 0_K$ . ( $0_K$ ,  $K \times 1$  boyutlu sıfır vektördür)
8:     Müsait ambulans sayısı,  $\hat{n}_{amb} = n_{amb}$ .
9:     for Tur sayısı,  $t = 1, 2, \dots, 14400$  do
10:      Her düğüm  $a_i$  için trafik bilgisini içeren konteks
        vektörü  $X_{t,a_i}$ 'yi gözlemler.
11:      if Yeni bir vaka  $c_t$  ve  $\hat{n}_{amb} > 0$  then
12:        Seyahat süresi olarak en yakın ambulans  $a_t$ 'yi
         $c_t$ 'ye yönlendir.
13:         $\hat{n}_{amb} = \hat{n}_{amb} - 1$ .
14:      else
15:        Gelen vakayı kuyruğun sonuna ekle. (ilk gelen
        ilk servis sistemi)
16:      end if
17:      if Ambulans  $a_\tau$ 'nin vaka  $c_\tau$ 'ya varması then
18:        Alınan ödülü belirle:  $r_\tau = 1/(t - \tau)$ .
19:        Tarih kümesini güncelle:
         $\mathcal{H}_t \leftarrow \mathcal{H}_{t-1} \cup \{c_\tau, a_\tau, r_{\tau,a_\tau}, X_{\tau,a_\tau}\}$ .
20:        Müdahale için 5 tur bekle ve ambulansı seyahat
        süresi en kısa hastaneye yönlendir.
21:      end if
22:      if Ambulans  $a_\tau$ 'nin hastaneye varması then
23:        Hastanede 15 tur bekle.
24:        if Kuyrukta bekleyen vaka varsa then
25:          Ambulansı hastaneden vakaya yönlendir.
26:        else
27:          Ambulansı boşa çıkar ve  $I_t$  tarafından belirlen-
          nen konuma yönlendir.
28:           $\hat{n}_{amb} = \hat{n}_{amb} + 1$ .
29:        end if
30:      end if
31:      if Ambulans konumlandırma durumu then
32:        Ambulansları yeni düğümlere konumlandır:
         $I_t \leftarrow \pi(\hat{n}_{amb}, K, \mathcal{H}_t, A, b)$ .
33:      end if
34:    end for
35:  end for
36: end for
```

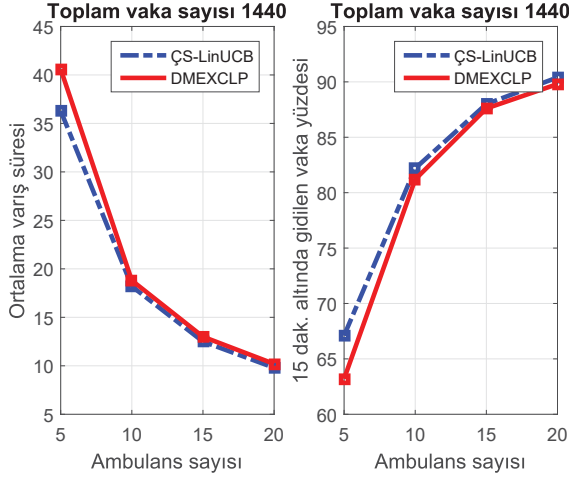
model, MEXCLP modeliyle aynı kapsama konseptini kullanmaktadır ve bir kaba kuvvet algoritmasıdır. Fakat bütün ambulanslar yerine sadece müsait olan ambulanslar için yeni konumlar hesaplandığı için gerçek zamanlı çalışmaya uygundur. [13]'te algoritmanın ambulansları konumlandırırken C_1, \dots, C_9 parametrelerini bildiği kabul edilmektedir. Fakat bu çalışmada, kullanılan DMEXCLP modeli haydut algoritması gibi vaka dağılımlarını bilmemektedir ve her turda gözlemlendiği vakalar ile C_1, \dots, C_9 kestirimi yapmaktadır. Kestirim için ortalama örneklem (average sampling) yöntemi kullanılmaktadır.

Literatürdeki birçok çalışmada olduğu gibi algoritmaların performansları kıyaslanırken iki performans ölçütü kullanılmıştır. Bunlardan ilki vakalara ortalama varış süresi, ikincisi de toplam T turda belirli bir sürenin altında varılan vakaların toplam vakalara oranıdır. Tipik acil yardım sistemlerinde ol-

TABLO I. ANKARA ŞEHİRİ İÇİN AMBULANS KONUMLANDIRMA PARAMETRELERİ

Parametre	Büyükük	Seçim
λ	1/10 dakika	Gün içi vaka dağılımı parametresi
K	2400	Düğüm sayısı
H	4	Bölgedeki hastane sayısı

duğu gibi bu çalışmada modellenen Ankara şehri için bu süre 15 dakika olarak seçilmiştir.



Şekil 2. Ortalama varış süresi (sol) ve 15 dakika altında varılan vaka yüzdesine (sağ) göre ÇS-LinUCB algoritması ile DMEXCLP algoritmasının performans karşılaştırması

Ambulans sayıları $n_{amb} = \{5, 10, 15, 20\}$ için algoritma test sonuçları Şekil 2'de gösterilmiştir. Sonuçlar 100 simülasyon koşulunun her birinde gerçekleşen toplam 1440 vakaya varış süresinin ortalaması alınarak hesaplanmaktadır. Aynı şekilde 15 dakika altında gidilen vaka yüzdesi, toplam 100 simülasyon koşulunun ortalaması alınarak hesaplanmıştır. ÇS-LinUCB algoritmasının DMEXCLP yöntemine göre vakalara ortalamada daha hızlı ulaştığı ve daha fazla vakaya 15 dakikanın altında erişebildiği görülmektedir.

IV. SONUÇLAR

Bu çalışmada ambulans konumlandırma problemi çok kollu haydut problemi olarak modellenmiş ve OSM açık kaynaklı harita uygulaması kullanılarak hazırlanan bir Ankara haritası üzerinde ambulans konumlandırma yapılmıştır. Daha önce bu konuda yapılan çalışmalardan farklı olarak, geliştirilen algoritma, optimizasyon literatüründe çok iyi bilinen ve dinamik ambulans yerleşimine olanak sağlayan DMEXCLP yöntemiyle karşılaştırılmıştır. Performans kıyası için ortalama varış süresi ve 15 dakika altında gidilen toplam vaka yüzdesi olmak üzere iki ölçüt kullanılmıştır. Farklı ambulans sayıları için de testler yapılmıştır. Hazırlanan haritada bağlamsal çok kollu haydut algoritması ÇS-LinUCB'nin DMEXCLP'ye göre her iki ölçüt açısından da daha iyi performans sergilediği gösterilmiştir. Ambulans konumlandırma problemine optimizasyon yöntemleriyle çözüm getiren çalışmalarda, vaka dağılımları ve düğümler arası seyahat süreleri gibi birçok parametrenin dağılım özellikleri biliniyor kabul edilmektedir. Gerçek hayattaki acil yardım sistemlerinde çözüm kümesini etkileyen, önceden bilinemeyip öğrenilmesi gereken birçok parametre olduğu göz

önüne alındığında, öğrenme tabanlı bir yaklaşımın önemi daha iyi anlaşılmaktadır.

KAYNAKLAR

- [1] L. Brotcorne, G. Laporte, and F. Semet, "Ambulance location and relocation models," *EU J. Op. Res.*, pp. 451-63, 2003.
- [2] M.S. Daskin, "A maximum expected location model: Formulation, properties and heuristic solution," *Trans. Sci.*, pp. 48-70, 1983.
- [3] M. Gendreau, G. Laporte, and F. Semet, "A dynamic model and parallel Tabu search heuristic for real-time ambulance relocation," *Parallel Comput.*, pp. 1641-53., 2001.
- [4] R. Nair and E. Miller-Hooks, "Fleet management for vehicle sharing operations," *Trans. Sci.*, pp. 524-40, 2011.
- [5] M.S. Maxwell, S.G. Henderson, and H. Topaloglu, "Ambulance redeployment: An approximate dynamic programming approach," *Winter Sim. Conf.*, pp. 1850-60, 2009.
- [6] S.G. Henderson, "Operations research tools for addressing current challenges in emergency medical services," *Wiley Enc. Op. Res. and Mgmt. Sci.*, 2011.
- [7] Y. Yue, L. Marla, and R. Krishnan, "An efficient simulation-based approach to ambulance fleet allocation and dynamic redeployment," *AAAI*, 2012.
- [8] A. Ingolfsson, S. Budge, and E. Erkut, "Optimal ambulance location with random delays and travel times," *Health Care Mgmt. Sci.*, pp. 262-74, 2008.
- [9] P. L. van den Berg and K. Aardal, "Time-dependent MEXCLP with start-up and relocation cost," *EU J. Op. Res.*, vol. 242, pp. 383-89, 2015.
- [10] V. Schmid, "Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming," *EU J. Op. Res.*, vol. 219, pp. 611-21, 2012.
- [11] D. Degel, L. Wiesche, S. Rachuba, and B. Werners, "Time-dependent ambulance allocation considering data-driven empirically required coverage," *Health Care Mgmt. Sci.*, pp. 444-58, 2015.
- [12] V. Bélanger, Y. Kergosien, A. Ruiz, and P. Soriano, "An empirical comparison of relocation strategies in real-time ambulance fleet management," *Computers Industrial Eng.*, pp. 216-29, 2016.
- [13] C. J. Jagtenberg, S. Bhulai, R.D. van der Mei, "An efficient heuristic for real-time ambulance redeployment," *Op. Res. Health Care*, vol. 4, pp. 27-35, 2015.
- [14] W.R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285-94, 1933.
- [15] T.L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Adv. Appl. Math.*, vol. 6, no. 1, pp. 4-22, 1985.
- [16] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time Analysis of the Multiarmed Bandit Problem," *Machine Learning*, vol. 47, no. 2/3, pp. 235-256, 2002.
- [17] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and non-stochastic multi-armed bandit problems," *Found. Trends Mach. Learn.*, vol. 5, no. 1, pp. 1-122, 2012.
- [18] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," *Proc. 19th Int. conf. on World Wide Web*, ACM, pp. 661-670, 2010.
- [19] C. Tekin, S. Zhang, and M. van der Schaar, "Distributed online learning in social recommender systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 4, pp. 638-652, 2014.
- [20] C. Tekin, O. Atan, and M. van der Schaar, "Discover the expert: Contextadaptive expert selection for medical diagnosis," *IEEE Trans. Emerging Topics Comput.*, vol. 3, no. 2, pp. 220-234, 2015.
- [21] L. Song, C. Tekin, and M. van der Schaar, "Online learning in largescale contextual recommender systems," *IEEE Trans. Services Comput.*, vol. 9, no. 3, pp. 433-445, 2016.
- [22] Ü. Şahin, V. Yücesoy, A. Koç, and C. Tekin, "Learning traffic congestion by contextual bandit problems for optimum localization," *25th Signal Proc. Com. App. Conf. (SIU)*, 2017, pp. 1-4.
- [23] Ü. Şahin, V. Yücesoy, A. Koç, and C. Tekin, "Risk-averse ambulance redeployment via multi-armed bandits," *26th Signal Proc. Com. App. Conf. (SIU)*, 2018, pp. 1-4. .