



# ULAŞIM AĞ TASARIMI PROBLEMLERİNİN ÇÖZÜMÜNDE DİFERANSİYEL GELİŞİM ALGORİTMASI TABANLI ÇÖZÜM YAKLAŞIMLARI

## DIFFERENTIAL EVOLUTION ALGORITHM BASED SOLUTION APPROACHES FOR SOLVING TRANSPORTATION NETWORK DESIGN PROBLEMS

Özgür BAŞKAN<sup>1\*</sup>, Hüseyin CEYLAN<sup>1</sup>

<sup>1</sup>İnşaat Mühendisliği Bölümü, Mühendislik Fakültesi, Pamukkale Üniversitesi, Denizli, Türkiye.  
obaskan@pau.edu.tr, hceylan@pau.edu.tr

Geliş Tarihi/Received: 01.04.2014, Kabul Tarihi/Accepted: 07.08.2014  
\* Yazışılan yazar/Corresponding author

doi: 10.5505/pajes.2014.08379  
Özel Sayı Makalesi/Special Issue Article

### Öz

Diferansiyel Gelişim Algoritması son yıllarda mühendislik optimizasyon problemlerinin çözümünde etkin olarak kullanılan bir yöntem olarak karşımıza çıkmaktadır. Temel olarak Genetik Algoritma tekniğine benzer çalışma prensibine sahip olan Diferansiyel Gelişim algoritması, diğer sezgisel algoritmalara oranla yapısal olarak daha basit olmasına karşın optimum değerlere ulaşmada daha kararlı bir yöntemdir. Bu çalışmada, Diferansiyel Gelişim Algoritması ulaşım ağ tasarımı problemlerine uygulanmakta ve çözüm üzerindeki etkinliği incelenmektedir. Bu kapsamda, Birleştirilmiş Ulaşım Ağ Tasarımı ve kentiçi karayolu ağlarındaki yol kenarı park yerlerinin belirlenmesi problemlerinin çözümü için iki seviyeli programlama yaklaşımı altında DG algoritması tabanlı modeller geliştirilmiştir. Bu modellerde, üst seviyede optimum yatırım ve parklanma stratejileri araştırılırken, alt seviyede sürücü reaksiyonlarını temsil eden Deterministik Trafik Atama problemi Frank-Wolfe algoritması ve VISUM trafik modelleme yazılımı kullanılarak çözülmüştür. Önerilen modellerin etkinliklerinin belirlenmesi amacıyla Sioux-Falls test ağı üzerinde sayısal uygulamalar gerçekleştirilmiştir. Elde edilen sonuçlar Diferansiyel Gelişim Algoritmasının ulaşım ağ tasarımı problemlerinin çözümünde etkin şekilde kullanılabileceğini göstermiştir.

**Anahtar kelimeler:** Diferansiyel gelişim, Ulaşım ağ tasarımı, İki seviyeli programlama.

### Abstract

Differential Evolution algorithm has effectively been used to solve engineering optimization problems recently. The Differential Evolution algorithm, which uses similar principles with Genetic Algorithms, is more robust on obtaining optimal solution than many other heuristic algorithms with its simpler structure. In this study, Differential Evolution algorithm is applied to the transportation network design problems and its effectiveness on the solution is investigated. In this context, Differential Evolution based models are developed using bi-level programming approach for the solution of the transportation network design problem and determination of the on-street parking places in urban road networks. In these models, optimal investment and parking strategies are investigated on the upper level. On the lower level, deterministic traffic assignment problem, which represents drivers' responses, is solved using Frank-Wolfe algorithm and VISUM traffic modeling software. In order to determine the effectiveness of the proposed models, numerical applications are carried out on Sioux-Falls test network. Results showed that the Differential Evolution algorithm may effectively be used for the solution of transportation network design problems.

**Keywords:** Differential evolution, Transportation network design, Bi-level programming.

## 1 Giriş

Birleştirilmiş Ulaşım Ağ Tasarımı (BUAT) problemi genel olarak bütçe kısıtları altında ulaşım ağında yapılabilecek en uygun iyileştirmelerin belirlenmesi olarak tanımlanabilir. BUAT problemi, Ayrık Ulaşım Ağ Tasarımı (AUAT) ve Sürekli Ulaşım Ağ Tasarımı (SUAT) problemlerinin birlikte göz önüne alınması ile ortaya çıkmıştır. Diğer bir deyişle BUAT probleminde ulaşım ağına eklenmesi düşünülen bağlar ve kapasite artırımına aday bağların belirlenmesi problemleri beraber ele alınmaktadır. Yerel idareciler ve ulaşım planlamacılarının en sık karşılaştığı problemlerin başında gelen BUAT problemi literatürde çözümü zor ulaştırma problemlerinin başında gelmektedir. Literatürde AUAT ve SUAT problemlerinin ayrı olarak değerlendirildiği birçok çalışma bulunmasına karşın BUAT probleminin çözümü için yeterli çalışma bulunmamaktadır. Poorzahedy ve Turnquist [1], BUAT probleminin çözümü için iki-seviyeli programlama modeli geliştirmişlerdir. Çalışmada yeni bağ yatırımı ve bağ kapasite genişletmelerine bağlı olarak toplam seyahat süresinin en küçüklenmesi amaçlanmış ve Dal-Sınır (DS) yaklaşımı tabanlı algoritma ile çözüm gerçekleştirilmiştir. AUAT problemlerinin çözümünde sıklıkla kullanılan DS

metodunun, çok sayıda karar değişkeni içeren problemler için yüksek bellek ve uzun hesaplama süresi gereksinimleri gibi dezavantajları bulunmaktadır [2],[3].

Poorzahedy ve Abulghasemi [4], BUAT probleminin çözümünde Karınca Sistemi optimizasyon yöntemini kullanmışlardır. Benzer şekilde Poorzahedy ve Rouhani [5], Karınca Sistemi tabanlı 7 farklı hibrit algoritma kullanarak BUAT problemini çözmüşlerdir. Geliştirilen algoritmalar Sioux-Falls ağı üzerinde kalibre edildikten sonra gerçek ulaşım ağında test edilmiştir. Sonuç olarak hibrit algoritmaların Karınca Sistemine oranla daha başarılı olduğu bulunmuştur. Luathep ve diğ. [6] BUAT probleminin çözümü için yeni bir optimizasyon algoritması önermişlerdir. Çalışmada üst seviyede bağ kapasite genişletme ve bağ ekleme problemleri birlikte ele alınmış, alt seviye problemi ise Wardrop Kullanıcı Dengesi (KD) prensibi göz önüne alınarak çözülmüştür. BUAT problemi karışık tamsayı doğrusal programlama problemine dönüştürülmüştür. Sayısal uygulamalar önerilen metodun oldukça başarılı olduğunu göstermiştir. Ceylan ve Ceylan [7] sinyal optimizasyonu probleminin çözümü amacıyla Armoni Araştırması Tekniği (AAT) ve Tepe Tırmanma optimizasyon tekniklerini birleştirmiştir. Sonuçlar geliştirilen tekniğin oldukça etkili olduğunu göstermiştir. Ceylan [8] optimal sinyal

sürelerinin belirlenmesi amacıyla Diferansiyel Gelişim (DG) algoritması tabanlı çözüm algoritması geliştirmiştir. Dell'Orco ve diğ. [9], SUAT problemini optimum trafik sinyal sürelerinin belirlenmesi olarak ele almış ve AAT tekniğinin performansı problemin çözümünde test edilmiştir. Başkan [10],[11] optimum bağ kapasite genişletmelerinin belirlenmesi probleminin çözümü için AAT ve Guguk Kuşu Algoritması optimizasyon metodlarını kullanmış ve literatürdeki çözüm yöntemleri ile karşılaştırmıştır. Literatürde BUAT alanında yapılan çalışmaların sınırlı olması nedeniyle, farklı sezgisel optimizasyon yaklaşımları kullanılarak yeni modellerin geliştirilmesinin faydalı olabileceği düşünülmektedir.

Çalışma kapsamında ele alınan bir diğer problem kentçi karayolu ağlarındaki yol kenarı park yerlerinin belirlenmesi problemidir. Günümüzde artan sosyo-ekonomik gereksinimler farklı günlük aktiviteleri tetiklemekte ve bunun bir sonucu olarak büyük ve gelişmekte olan şehirlerde yaşayanların hareketliliği artmaktadır. Sürekli değişen hareketlilik talebini karşılamak için insanlar özel taşıt türünü kullanma eğilimi göstermektedirler. Özel taşıt türüne olan talebin artması da kentçi yol ağlarında parklanma problemlerine yol açmaktadır. Shoup [12], zirve saatlerde önemli sayıda sürücünün uygun bir park yeri arayışı içinde seyrettiğini belirtmektedir. Çalışmada vurgulanan bir diğer husus da, ücretsiz/düşük ücretli ve plansız yol kenarı parklanmaların yol ağı kapasitesinde ciddi azalmalara neden olabileceği ve planlı/denge fiyatı üzerinden ücretlendirilen yol kenarı parklanmaların ideal bir kamusal gelir kaynağı olabileceğidir. Yousif ve Purnawan [13] farklı park yeri tasarımlarının park manevra süresi ve aralık kabul değerleri üzerindeki etkilerini araştırmışlardır. Portilla ve diğ. [14] yol kenarı park manevraları ve kötü park edilmiş taşıtların ortalama bağ seyahat süreleri üzerindeki etkilerini incelemiştir. Çalışmada her iki durumda da bağ seyahat süreleri ve ağ kapasitesinin önemli ölçüde etkilendiği belirtilmektedir. Yol kenarı park yerlerinin belirlenmesine ilişkin literatürdeki çalışmalar incelendiğinde, araştırmacıların genel olarak yol kenarı park yerlerinin ve buralardaki parklanma manevralarının bağ seyahat süreleri ve ağ kapasitesi üzerindeki etkileri üzerine yoğunlaştıkları görülebilmektedir. Diğer taraftan, ağ genelinde parklanmaya izin verilebilecek yol kesimlerinin belirlenmesine yönelik yaklaşımların geliştirilmesi faydalı olabilecektir.

Herhangi bir yol kesimindeki bir şeridin yol kenarı parklanmaya tahsis edilip edilmeyeceğini temsil eden 0-1 değişkenlerinin kullanımından dolayı bu problem AUAT kapsamında ele alınabilmektedir. Literatürde farklı AUAT problemlerinin çözümü için geliştirilmiş çeşitli yaklaşımlar bulunmaktadır. Bruynooghe [15], yol ağlarının performanslarının artırılması amacıyla gerçekleştirilecek optimum yatırım stratejilerinin belirlenmesi için tam sayılı programlama modeli geliştirmiştir. LeBlanc [16], AUAT probleminin çözümü için DS tekniği tabanlı bir çözüm yöntemi geliştirmiştir. Gao ve diğ. [17], yol ağına ilave edilecek yeni bağlar ve trafik hacimleri arasındaki ilişkiyi açıklamaya yönelik destek fonksiyonu tabanlı bir çözüm algoritması geliştirmişlerdir. Duthie ve Waller [18], AUAT probleminin gerek konveks ve sürekli olmayan yapısı gerekse DS tekniğinin dezavantajlarından dolayı sezgisel optimizasyon yöntemleri ile ele alınması gerektiğini belirtmişlerdir. Ceylan ve Ceylan [19], AUAT problemini karma tamsayılı programlama problemi olarak ele almışlar ve çözüm için AAT tabanlı bir çözüm yöntemi geliştirmişlerdir. Çalışmada, geliştirilen

modelin kapasite artırımı ve trafik yönlendirme problemlerinin çözümünde başarılı sonuçlar verdiği belirtilmiştir.

Bu çalışmada, BUAT ve yol kenarı park yerlerinin belirlenmesi problemlerinin çözümü için iki-seviyeli ikili tamsayı programlama modelleri geliştirilmiştir. Konveks olmayan yapıları ve içerdikleri 0-1 değişkenlerinden dolayı problemlerin çözümü için son yıllarda karmaşık mühendislik problemlerinin çözümü amacıyla oldukça sık olarak kullanılan DG algoritması kullanılmıştır. Geliştirilen modeller Sioux-Falls ulaşım ağı üzerinde test edilmiştir. Sonuçlar DG algoritmasının her iki problem için de etkin şekilde kullanılabilirliğini ortaya koymuştur.

Çalışmanın ikinci bölümünde problem formülasyonu, sonraki bölümde DG algoritmasının çalışma prensibi, dördüncü bölümde sayısal uygulamalar ve son bölümde sonuçlar verilmiştir.

## 2 Problem Formülasyonu

BUAT problemi, belirli bütçe kısıtı altında ulaşım ağındaki en uygun bağ ekleme-çıkarma ve bağ kapasite genişletme stratejilerinin belirlenmesi olarak tanımlanmaktadır. Düğüm (V) ve bağlardan (A) oluşan bir ulaşım ağı  $N(V,A)$  olarak temsil edilirse BUAT problemi aşağıda verildiği gibi ifade edilebilmektedir.

$$\begin{aligned} \text{Min}_y Z(\mathbf{y}) &= \sum_{(i,j) \in A \cup A_{y1}} x_{ij}^* t_{ij}(x_{ij}^*) \\ &\sum_{(i,j) \in A_y} C_{ij} y_{ij} \leq B \\ y_{ij} &= 0/1, (i,j) \in A_y \end{aligned} \quad (1)$$

Burada;  $x_{ij}^*$  alt seviye probleminin çözümünden elde edilen  $(i,j)$  bağındaki denge akımı,  $t_{ij}(x_{ij}^*)$   $(i,j)$  bağının maliyet fonksiyonu,  $\mathbf{y}$  yatırım vektörü,  $A_y$  yatırım yapılması planlanan bağlar kümesi,  $y_{ij} = 1/0$  aday  $(i,j)$  bağının yatırıma dahil edilip edilmemesini temsil eden parametre,  $A_{y1} = \{(i,j) \in A_y : y_{ij} = 1\}$  yatırım planlanan bağlar içinde kabul edilen projelerin kümesi,  $C_{ij}$  yatırıma aday  $(i,j)$  bağının iyileştirme yada yapım maliyeti ve  $B$  toplam yatırım bütçesi olarak verilmiştir. Uygulanması düşünülen herhangi bir yatırım vektörü ( $\mathbf{y}$ ) için Deterministik Trafik Atama (DTA) problemi aşağıdaki gibi ifade edilebilir.

$$\begin{aligned} \text{Min}_x \sum_{(i,j) \in A \cup A_{y1}} \int_0^{x_{ij}} t_{ij}(w) dw \\ \sum_{k \in K} f_k^{rs} = D_{rs} \quad \forall r \in R, s \in S, k \in K_{rs} \\ f_k^{rs} \geq 0 \quad \forall r \in R, s \in S, k \in K_{rs} \\ x_{ij} = \sum_{rs} \sum_{k \in K_{rs}} f_k^{rs} \delta_{ij,k}^{rs} \quad \forall r \in R, s \in S, k \in K_{rs}, \forall (i,j) \in A \cup A_{y1} \end{aligned} \quad (2)$$

Burada,  $D_{rs}$  Başlangıç-Varış (B-V) çifti  $r$ - $s$  arasındaki seyahat talebi,  $f_k^{rs}$  B-V çifti  $r$ - $s$  arasındaki  $k$  rotasındaki akım ve  $\delta_{ij,k}^{rs}$

bağ-rota matrisinin elemanı olup eğer  $(i,j)$  bağı  $k$  rotası üzerinde ise 1 aksi durumda 0 değerini almaktadır. DTA problemi konveks bir problem olduğu için sayısal olarak birçok farklı metotla çözülebilmektedir. Çalışmada, BUAT probleminde denge bağ akımlarının bulunmasında oldukça etkili olan ve literatürde sıkça kullanılan Frank-Wolfe (FW) metodu kullanılmıştır [20].

Çalışmada DG algoritmasının performansının test edileceği yol kenarı park yerlerinin belirlenmesi problemi parklanmaya tahsis edilecek yol kesimlerinin uzunluğunun maksimize edilmesi şeklinde ele alınarak aşağıdaki gibi formüle edilebilir:

$$\frac{x_m}{c_m} \leq 1 \quad (\forall m \in M) \quad (3)$$

kısıtına bağlı olarak,

$$\text{maks} \sum_{m=1}^M u_m l_m \quad (4)$$

Burada,  $M$  yol kenarı parklanmaya aday bağlar seti  $(\forall m \in M)$ ,  $x_m$   $m$  bağındaki trafik hacmi,  $c_m$   $m$  bağının kapasitesi ve  $l_m$   $m$  bağının uzunluğudur. Herhangi bir bağı bir şeridinin parklanmaya tahsis edilip edilmeyeceğini belirleyen  $u_m$  ikili değişkeni aşağıdaki gibi ifade edilebilir:

$$u_m = \begin{cases} 1 & m \text{ bağının bir şeridi parklanmaya tahsis edilecekse} \\ 0 & \text{aksi halde} \end{cases} \quad (5)$$

Denklem (3)'te verilen kısıt, parklanmaya aday bağlardaki KD trafik hacimlerinin kapasiteyi aşmamasını sağlamaktadır. KD trafik hacimleri Denklem (2)'de verilen DTA probleminin çözülmesi ile belirlenebilir. Yol kenarı park yerlerinin optimum uzunluğunun belirlenmesi probleminde bağ trafik hacimleri VISUM trafik modelleme yazılımı kullanılarak belirlenmiştir.

### 3 Diferansiyel Gelişim Algoritması

DG algoritması optimizasyon problemlerinin çözümü amacıyla Storn ve Price [21], tarafından geliştirilen toplum tabanlı sezgisel bir metottur. Yapısal olarak oldukça basit olmasına rağmen karmaşık problemlerin çözümünde oldukça etkilidir. DG algoritmasında oluşturulan başlangıç toplumu mutasyon, çaprazlama ve seçim operatörleri kullanılarak en iyi değer elde edilmesi amacıyla iterasyonlar boyunca iyileştirilmektedir [22]. Çözüm sürecini kontrol etmek amacıyla üç adet parametre kullanılmaktadır. Bunlardan birincisi tüm toplum tabanlı sezgisel metotlar da kullanıldığı gibi toplum büyüklüğünü temsil eden  $NP$  parametresidir. Bu değer verilen bir problem için dikkate alınan çözüm vektörlerinin sayısını göstermektedir. Algoritma içinde kullanılan bir diğer kontrol parametresi  $F$  ile temsil edilen mutasyon faktörüdür. Bu parametre toplum içinden rastgele seçilen ve birbirinden farklı 3 adet çözüm vektöründen yeni bir vektör üretilmesi amacıyla kullanılmaktadır.  $F$  parametresi için Storn ve Price, [21] tarafından tavsiye edilen kullanım aralığı [0.5-1]'dir. DG algoritmasında kullanılan son kontrol parametresi ise çaprazlama oranıdır ( $CR$ ). Bu parametre

mutasyon sonucu elde edilen vektörün dikkate alınma oranı olarak kullanılmaktadır.  $CR$  parametresi için [0.8-1] kullanım aralığı tavsiye edilmektedir [21]. Bu çalışmada sayısal uygulamalarda  $F$  ve  $CR$  parametreleri 0.8 olarak seçilmiştir. Şekil 1'de BUAT probleminin çözümünde kullanılan DG algoritmasının akış şeması verilmiştir.

DG algoritmasında başlangıç toplumu  $i \in \{1, 2, \dots, N\}$  ve  $j \in \{1, 2, \dots, NP\}$  olmak üzere  $y = [y_i^j]$  çözüm vektörlerinden oluşacak şekilde 0-1 ikili değişkenleri ile rastgele oluşturulur. Burada,  $N$  iyileştirme ya da yapımı düşünülen bağların toplam sayısı veya parklanmaya aday bağların sayısını temsil etmektedir. Başlangıç toplumu içindeki her bir çözüm vektörü için Denklem (2) yardımıyla denge bağ akımları elde edilir. Bulunan denge bağ akımları ve Denklem (1) yardımıyla amaç fonksiyonu değerleri her bir çözüm vektörü için hesaplanır. Sonrasında toplum içinden rastgele seçilen üç adet çözüm vektörü ve mutasyon faktörü yardımıyla Denklem (6)'da verilen operatör kullanılarak deneme vektörü,  $m$ , oluşturulur. Çözümde 0-1 ikili değişkenleri ile çalışıldığı için mutasyon operatöründen sonra elde edilen yeni vektör elemanlarının sınır dışına çıkıp çıkmadığı kontrol edilmelidir. 0-1 değerleri arasında kalan değerler en yakın tamsayı değerine yuvarlatılır.

$$m_i^j = y_i^j + F(y_i^j - y_i^3) \quad (6)$$

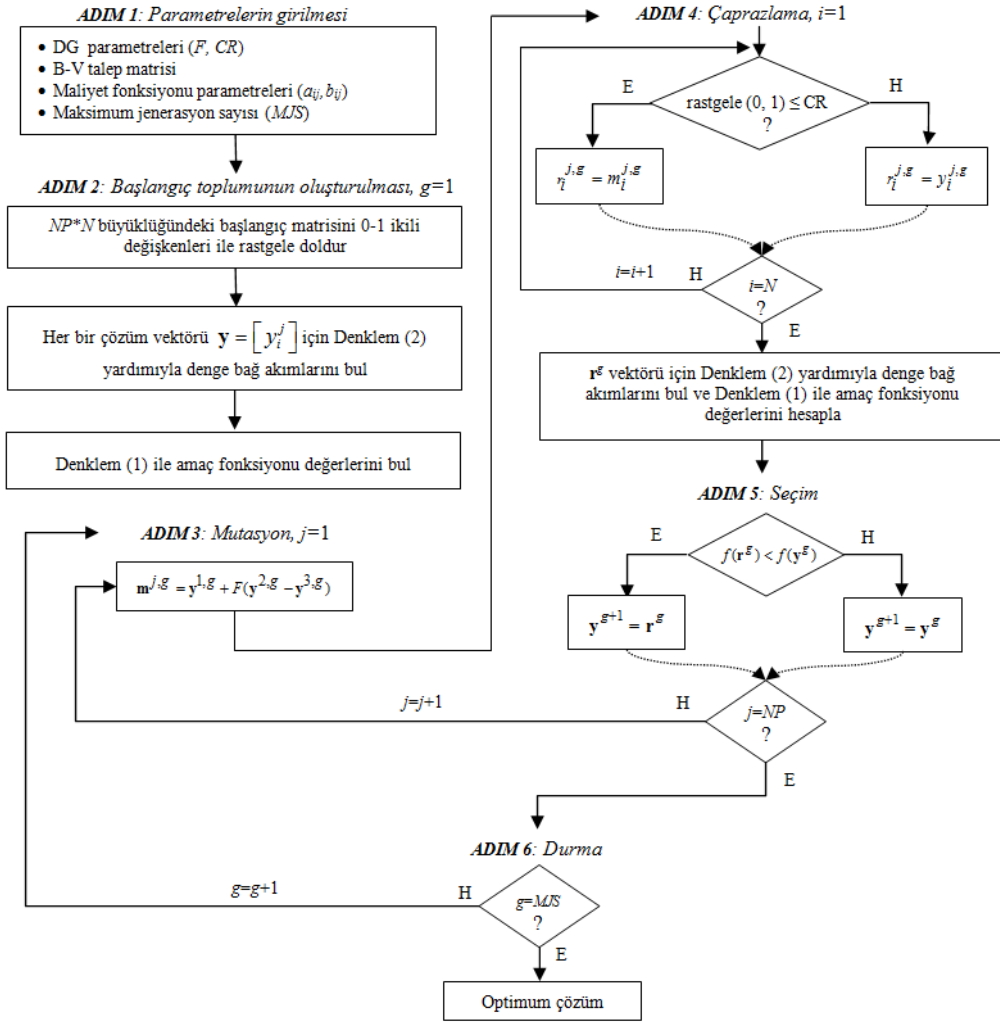
Burada  $y_i^1$ ,  $y_i^2$  ve  $y_i^3$  birbirinden farklı olarak  $[0, NP]$  aralığında rastgele seçilen karar değişkenlerini temsil etmektedir. Toplum içindeki her bir çözüm vektörü ile buna bağlı olarak oluşturulan  $m$  vektörüne Denklem (7)'de verildiği şekilde çaprazlama operatörü uygulanır ve  $r$  vektörü oluşturulur.

$$r_i^j = \begin{cases} m_i^j, & \text{eğer rastgele } (0,1) \leq CR \text{ yada } i = i_{\text{rastgele}} \\ y_i^j, & \text{aksi takdirde} \end{cases} \quad (7)$$

Burada,  $(0,1)$  arasında rastgele üretilen değer  $CR$  parametresinden küçük ya da eşit ise  $r$  vektörünün elemanı  $m$  vektöründen aksi durumda ise  $y$  hedef vektöründen seçilir.  $i = i_{\text{rastgele}}$  koşulunun kullanılmasının amacı  $r$  vektörünün en az bir elemanının  $m$  vektöründen seçilmesinin sağlanmasıdır. Bu sayede gelecek jenerasyonlara yeni bireylerin aktarılması sağlanmaktadır. Son olarak çaprazlama sonucu elde edilen  $r$  vektörü ile  $y$  vektörüne ait amaç fonksiyonu değerleri karşılaştırılır. Öncelikle  $r$  vektörünün temsil ettiği çözüm vektörüne bağlı olarak Denklem (2) yardımıyla denge bağ akımları hesaplanır ve elde edilen bağ akımları ve Denklem (1) ile amaç fonksiyonu değeri hesaplanır. Bu aşamadan sonra Denklem (8)'de verilen koşullu ifade yardımıyla en iyi amaç fonksiyonu veren çözüm vektörü bir sonraki jenerasyona aktarılmaktadır.

$$y^{g+1} = \begin{cases} r^g, & \text{Eğer } f(r^g) \leq f(y^g) \\ y^g, & \text{aksi takdirde} \end{cases} \quad (8)$$

Burada  $g$  jenerasyon sayısını ifade etmektedir. DG algoritması belirlenen durma kriterinin sağlanması ya da maksimum jenerasyon sayısına ulaşılması durumunda sonlandırılır.



Şekil 1: DG algoritması akış şeması.

#### 4 Sayısal Uygulamalar

DG algoritmasının BUAT problemindeki performansının test edilmesi amacıyla Şekil 2'de verilen 24 düğüm ve 76 bağdan oluşan Sioux-Falls ulaşım ağı seçilmiştir. Şekilden görülebileceği gibi 10 çift bağda yatırım yapılması planlanmaktadır. 5 çift bağ yeni yapılması düşünülen bağlar olup diğerleri ise kapasite genişletmesi yapılması planlanan bağlardır. Bütçe kısıtını amaç fonksiyonuna dahil etmek amacıyla Denklem (1)'e ceza fonksiyonu eklenmiş ve Denklem (9)'da verilmiştir.

$$\text{Min } Z(\mathbf{y}) = \sum_{(i,j) \in A \cup A_1} x_{ij}^* t_{ij}(x_{ij}^*) + \alpha * \text{maks} \left( \sum_{(i,j) \in A_y} C_{ij} y_{ij} - B, 0 \right) \quad (9)$$

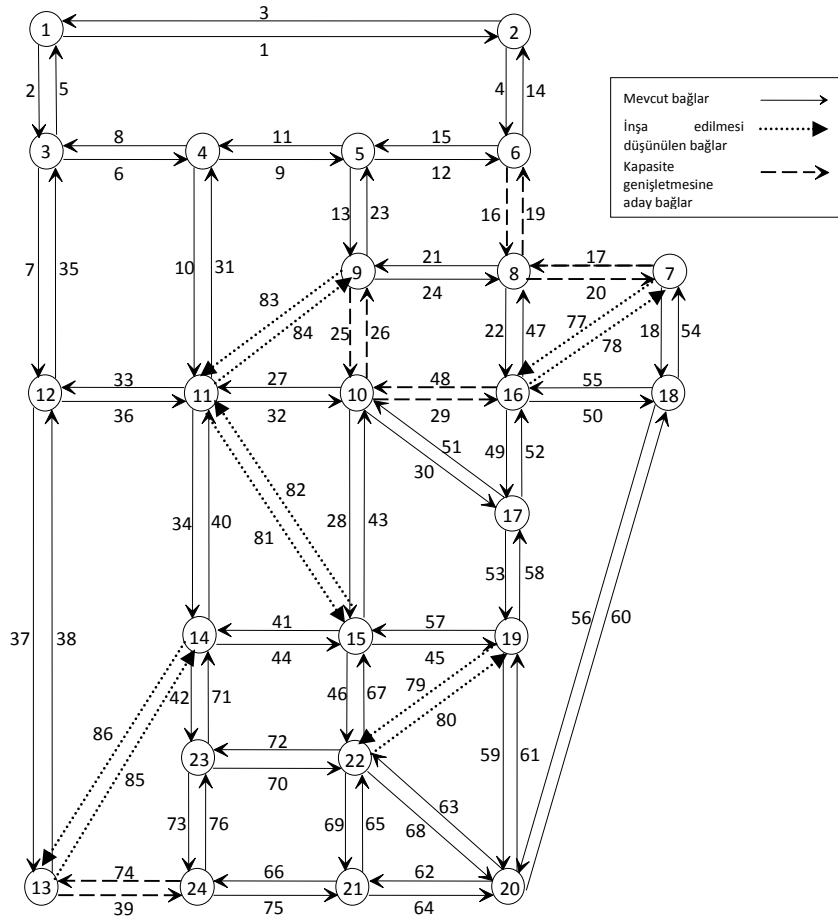
Burada,  $\alpha$  ceza fonksiyonu sabiti olup  $10^3$  olarak seçilmiştir. Toplam yatırım maliyetlerinin verilen bütçe kısıtını aşması durumunda amaç fonksiyonuna ceza uygulanmakta aksi durumda ise ceza fonksiyonu sıfır değerini almaktadır. Maliyet fonksiyonu,  $t_{ij} = a_{ij} + b_{ij} x_{ij}^4$ , parametreleri ve B-V talep matrisi Poorzahedy ve Turnquist [1] ve LeBlanc [16]'den alınmıştır.

BUAT probleminin çözümünde kapasite genişletmesi yapılması planlanan bağ çiftlerine (9-10, 10-9; 6-8, 8-6; 13-24, 24-13; 7-8, 8-7; 10-16, 16-10) ait yapım maliyetleri sırasıyla 625, 650, 850, 1000 ve 1200, yeni inşa edilmesi planlanan bağ çiftlerinin (7-16, 16-7; 19-22, 22-19; 11-15, 15-11; 9-11, 11-9; 13-14, 14-13) yapım maliyetleri ise 1500, 1650, 1800, 1950 ve 2100 olarak seçilmiştir [5]. Çalışmada 2700, 6500 ve 10820 olmak üzere 3 farklı bütçe için çözüm yapılmıştır. DG algoritmasının sonuçları Poorzahedy ve Rouhani [5], tarafından geliştirilen Karınca Sistemi (KS) sonuçları ile karşılaştırılmış sonuçlar Tablo 1'de verilmiştir.

Tablo 1: Farklı Bütçe/Maliyet (B/M) oranları için sonuçların karşılaştırılması.

B/M	Bütçe	DG		KS		Z <sup>a</sup>	
		Ortalama DTA sayısı	Z <sup>a</sup>	B/M	Bütçe		Ortalama DTA sayısı
0.20	2700	38.0	65293	0.20	2700	19.2	76297
0.49	6500	33.7	53366	0.49	6500	23.9	65465
0.81	10820	14.0	49845	0.81	10820	18.1	58829

<sup>a</sup>: En iyi amaç fonksiyonu değeri.



Şekil 2: Sioux-Falls ulaşım ağı.

Tablo 1'den görüldüğü gibi amaç fonksiyonu değerlerine göre DG algoritması üç farklı  $B/M$  oranı için KS yöntemine göre daha iyi sonuçlar üretmektedir. Ortalama DTA sayıları açısından her iki algoritma arasında kayda değer bir farklılık görünmemektedir. 0.20 ve 0.49  $B/M$  oranları için DG algoritması daha yüksek sayıda DTA gerektirmesine rağmen, en yüksek bütçeli çözümde KS algoritması daha fazla DTA sayısında en iyi çözüme ulaşabilmektedir.

Orta bütçeli durumu ifade eden  $B/M=0.49$  durumu için elde edilen sonuçlar Şekil 3'te ulaşım ağı üzerinde gösterilmiştir. En iyi yatırım projesi şekilden görüldüğü gibi 13-24 ve 24-13 bağlarında kapasite genişletmesi yapılması, 19-22, 22-19; 11-15, 15-11 ve 9-11, 11-9 bağlarının ise ulaşım ağına eklenmesi durumudur. Ayrıca Tablo 2'de DG algoritması ile elde edilen en iyi yatırım projeleri her  $B/M$  oranı için verilmiştir.

Tablo 2: Farklı  $B/M$  oranları için en iyi yatırım stratejileri.

$B/M$	Yatırıma Aday Bağlar									
	9-10 10-9	6-8 8-6	13-24 24-13	7-8 8-7	10-16 16-10	7-16 16-7	19-22 22-19	11-15 15-11	9-11 11-9	13-14 14-13
0.20	1							1		
0.49			1				1	1	1	
0.81		1		1	1	1	1	1		1

Yol kenarı park yerlerinin belirlenmesi probleminin çözümünde parklanmaya ayrılan yol kesimlerinin toplam

uzunluğunun belirlenmesi için Denklem (4)'te verilen amaç fonksiyonu aşağıdaki gibi düzenlenmiştir:

$$\min \left( \sum_{i=1}^m G_i(v_i) - u_i l_i \right) \quad (10)$$

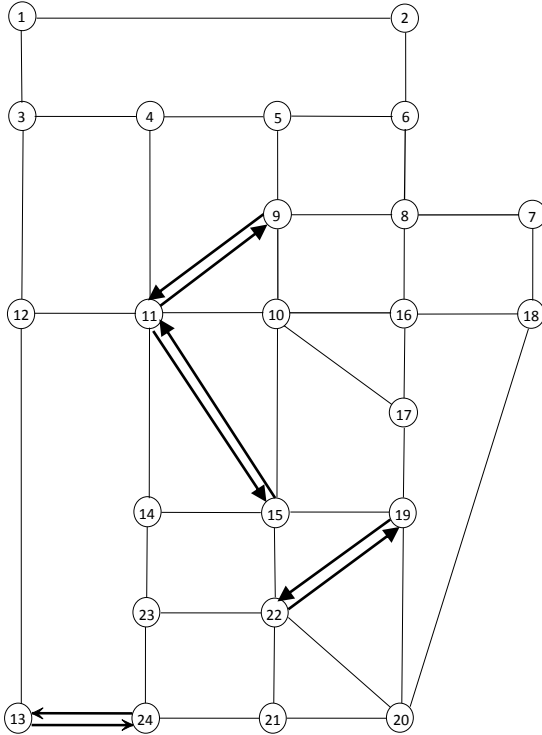
Burada,  $G_i(V_i)$  Denklem (4)'te verilen eşitsizlik kısıtının sağlanması amacıyla kullanılan ceza fonksiyonu olup aşağıdaki şekilde ifade edilmektedir:

$$G_i(v_i) = \begin{cases} \alpha & \text{eğer } \frac{v_i}{c_i} > 1 \quad (\forall i \in M) \\ 0 & \text{aksi halde} \end{cases} \quad (11)$$

DG algoritmasının yol kenarı park yerlerinin belirlenmesi problemindeki performansını test etmek amacıyla Şekil 2'de verilen Sioux Falls yol ağı üzerinde uygulama yapılmıştır. Tablo 3'te Sioux-Falls yol ağındaki bağların uzunlukları verilmektedir. Çözümde yol ağındaki bağların iki şeritten oluştuğu kabul edilmiştir. Bağ kapasiteleri ve serbest akım seyahat süreleri Suwansirikul ve diğ. [23], B-V talepleri LeBlanc [16] çalışmalarından elde edilebilir. DG algoritması parametreleri olan mutasyon faktörü ve çaprazlama oranı için 0.80 değeri kullanılmış ve popülasyon büyüklüğü 30 olarak kabul edilmiştir. Çözüm 1000. jenerasyonda durdurulmuştur. Ceza faktörü  $\alpha$ , hacim/kapasite oranı "1"den büyük olan her

bir bağ için 4 km alınarak amaç fonksiyonuna dahil edilmiştir. Bağ seyahat sürelerinin hesabında kullanılan fonksiyon Denklem (12)'de verilmektedir [24].

$$t_i = t_i^0 \left[ 1 + 0.15(x_i/c_i)^4 \right] \quad (12)$$

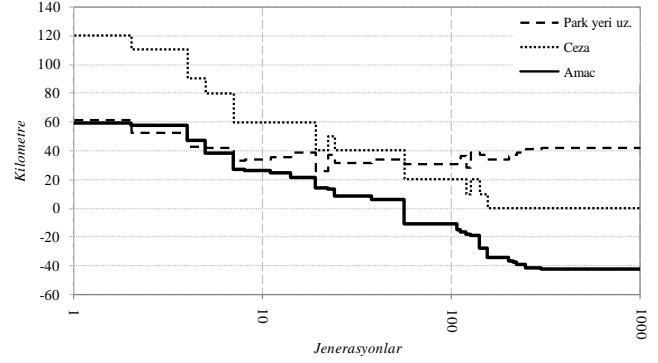


Şekil 3: B/M=0.49 için en iyi yatırım projesi.

Bu çalışmada, yol ağının orijinal hali için hacim/kapasite oranı "1" in üzerinde olan 26 bağ parklanmaya aday bağlar listesinden çıkarılmış olup kalan 50 bağ yol kenarı parklanmaya tahsis edilebilecek bağlar olarak kabul edilmiştir. Söz konusu 50 bağın toplam uzunluğu 112 km'dir. Model sonucunda elde edilen yakınsama grafiği Şekil 4'de verilmektedir. Şekilden görüldüğü üzere yaklaşık 300 jenerasyon sonunda çözüme ulaşılmıştır. Ayrıca, ilk jenerasyonda 120 km mertebesinde olan ceza değerinin yaklaşık 200 jenerasyon sonunda sıfıra inmesi, Denklem (3)'de verilen hacim/kapasite kısıtının sağlandığını göstermektedir. Yol ağının çözüm sonrası görünümü ve sonuçlar sırasıyla Şekil 5 ve Tablo 4'te verilmektedir.

Tablo 3: Bağ uzunlukları.

Bağlar	Uzunluk (km)
9, 11, 16, 18, 19, 49, 52, 53, 54, 58, 65, 69, 73, 76	1.0
17, 20, 25, 26, 37, 38, 45, 46, 50, 55, 57, 66, 67, 75	1.5
2, 5, 6, 7, 8, 12, 15, 29, 34, 35, 39, 40, 42, 48, 56, 59, 60, 61, 70, 71, 72, 74	2.0
4, 13, 14, 22, 23, 27, 32, 41, 44, 47, 63, 68	2.5
1, 3, 10, 28, 31, 33, 36, 43, 62, 64	3.0
30, 51	4.0
21, 24	5.0



Şekil 4: Çözüm sürecinin yakınsama grafiği.

Tablo 4: Çözüm sonucunda elde edilen parklanma planı.

Parklanmaya tahsis edilecek linkler	Toplam uzunluk (km)
1, 2, 3, 5, 6, 7, 8, 9, 11, 18, 21, 24, 35, 37, 38, 50, 54, 55, 56, 60	42.0

Tablo 4'ten görüldüğü üzere çözüm sonucunda toplam 42 km uzunluğundaki 20 bağ yol kenarı parklanmaya tahsis edilmiştir. Ağ genelindeki toplam seyahat süresindeki ve ortalama hacim/kapasite oranındaki değişim Tablo 5'te verilmektedir.

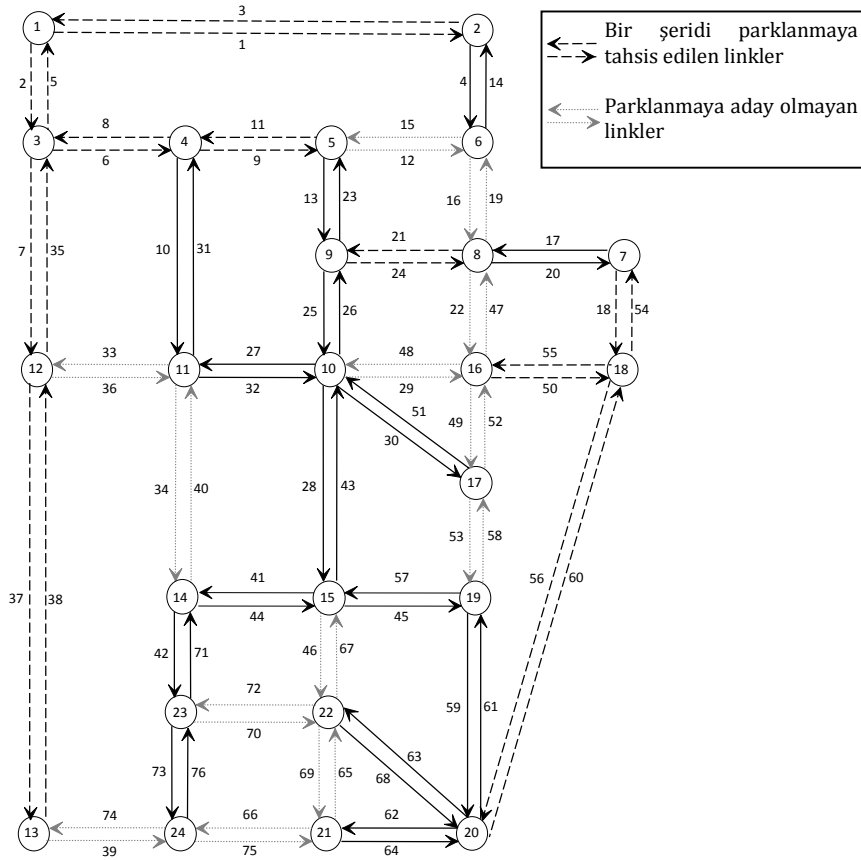
Tablo 5: Yol ağındaki toplam seyahat süresi ve ortalama hacim/kapasite oranı.

	Orijinal	Çözüm Sonrası	Değişim (%)
Toplam seyahat süresi (taşıt-sa)	18706	18784	0.42
Ortalama Hacim/kapasite oranı	0.76	0.83	9.21

Tablo 5'ten görülebildiği üzere çözüm sonrası yol ağındaki toplam seyahat süresindeki artış %1'in altında gerçekleşmektedir. Diğer taraftan ortalama hacim/kapasite oranının yaklaşık %9 arttığı görülmektedir. Ayrıca, parklanmaya aday bağların tamamı için çözüm sonrasında hacim/kapasite oranı "1" in altında kalmıştır.

## 5 Sonuçlar

Bu çalışmada, BUAT ve yol kenarı park yerlerinin belirlenmesi problemleri ele alınmış ve çözüm için karmaşık optimizasyon problemlerinin çözümünde sıkça kullanım alanı bulan DG algoritması kullanılmıştır. Söz konusu problemlerin yapısından dolayı çözümde 0-1 ikili değişkenleri kullanılmıştır. BUAT probleminin çözümü için Sioux-Falls ulaşım ağı üzerinde farklı bütçe/maliyet oranları için çözümler yapılmış ve literatürdeki sonuçlar ile karşılaştırılmıştır. Sonuçlar DG algoritmasının BUAT probleminin çözümünde oldukça başarılı olduğunu ayrıca kullanım kolaylığı açısından gerçek ulaşım ağlarında kullanılabileceğini göstermiştir. Kentiçi karayolu ağlarında yol kenarı park yerlerinin belirlenmesi probleminin çözümü için iki-seviyeli simülasyon/optimizasyon modeli geliştirilmiştir. Önerilen modelde üst seviyede parklanmaya tahsis edilebilecek yol kesimlerinin maksimizasyonu ele alınırken, alt seviyede DTA problemi VISUM yazılımıyla çözülmüştür. Geliştirilen model Sioux Falls yol ağına uygulanmıştır. Sonuçlar DG algoritması tabanlı modelin yol kenarı park yerlerinin belirlenmesi probleminde etkin bir şekilde kullanılabileceğini göstermiştir.



Şekil 5: Model uygulaması sonucunda ağın görünümü.

## 6 Teşekkür

Bu çalışma 10. Ulaştırma Kongresi'nde "Birleştirilmiş Ulaşım Ağ Tasarım Probleminin Diferansiyel Gelişim Algoritması ile Çözümü" ve "Kentiçi Karayolu Ağlarında Yol Kenarı Park Yerlerinin Diferansiyel Gelişim Algoritması Kullanılarak Belirlenmesi" başlıklı bildiriler kapsamında sunulmuştur.

## 7 Kaynaklar

- [1] Poorzahedy H, Turnquist MA. "Approximate Algorithm for the Discrete Network Design Problem". *Transportation Research Part B*, 16(1), 45-55, 1982.
- [2] Heragu SS. *Facilities Design*. Boston, USA, PWS Publishing Company, 1997.
- [3] Pinedo ML. *Scheduling Theory. Algorithms and Systems*. 3rd ed. New York, USA, Springer Verlag LLC, 2008.
- [4] Poorzahedy H, Abulghasemi F. "Application of Ant System to Network Design Problem". *Transportation*, 32, 251-273, 2005.
- [5] Poorzahedy H, Rouhani OM. "Hybrid Meta-Heuristic Algorithms for Solving Network Design Problem". *European Journal of Operational Research*, 182, 578-596, 2007.
- [6] Luathep P, Sumalee A, William HKL, Li ZC, Lo HK. "Global Optimization Method for Mixed Transportation Network Design Problem: A Mixed-Integer Linear Programming Approach". *Transportation Research Part B*, 45(5), 808-827, 2011.
- [7] Ceylan H, Ceylan H. "A Hybrid Harmony Search and TRANSYT Hill Climbing Algorithm for Signalized Stochastic Equilibrium Transportation Networks". *Transportation Research Part-C*, 25, 152-167, 2012.
- [8] Ceylan H, "Optimal Design of Signal Controlled Road Networks Using Differential Evolution Optimization Algorithm", *Mathematical Problems in Engineering*, 2013, 1-11, 2013.
- [9] Dell'Orco M, Baskan O, Marinelli M. "A Harmony Search Algorithm Approach for Optimizing Traffic Signal Timings". *Promet Traffic&Transportation*, 25(4), 349-358, 2013.
- [10] Baskan O. "Determining Optimal Link Capacity Expansions in Road Networks Using Cuckoo Search Algorithm with Levy Flights". *Journal of Applied Mathematics*, 2013, 1-11, 2013.
- [11] Baskan O. "Harmony Search Algorithm for Continuous Network Design Problem with Link Capacity Expansions". *KSCE Journal of Civil Engineering*, 18(1), 273-283, 2014.
- [12] Shoup DC. "The Ideal Source of Public Revenue". *Regional Science and Urban Economics*, 34(6), 753-784, 2004.
- [13] Yousif S, Purnawan. "Traffic Operations at On-Street Parking Facilities". *Proceedings of the Institution of Civil Engineers-Transport*, 157(3), 189-194, 2004.
- [14] Portilla AI, Orena BA, Berodia JLM, Diaz FJR. "Using M/M/∞ Queuing Model in On-Street Parking Maneuvers". *Journal of Transportation Engineering*, 135(8), 527-535, 2009.

- [15] Bruynooghe M. "An Optimal Method of Choice of Investments in a Transport Network". Presentation, Planning&Transport Research & Computation Seminars on Urban Traffic Model Research, London, England, 1972.
- [16] LeBlanc LJ. "An Algorithm for the Discrete Network Design Problem". *Transportation Science*, 9(3), 183-199, 1975.
- [17] Gao ZY, Wu JJ, Sun HJ. "Solution Algorithm for the Bi-Level Discrete Network Design Problem". *Transportation Research Part B*, 39(6), 479-495, 2005.
- [18] Duthie J, Waller ST. "Incorporating Environmental Justice Measures into Equilibrium-Based Network Design". *Journal of the Transportation Research Board*, 2089, 58-65, 2008.
- [19] Ceylan H, Ceylan H. "Şehirçi Karayolu Ağlarının Sezgisel Harmoni Araştırması Optimizasyon Yöntemi ile Ayrık Tasarımı". *İMO Teknik Dergi*, 24(1), 6211-6231, 2013.
- [20] Frank M, Wolfe P. "An Algorithm for Quadratic Programming". *Naval Research Logistics Quaterly*, 3(1-2), 95-110, 1956.
- [21] Storn R, Price K. "Differential Evolution: A Simple and Efficient Adaptive Scheme for Global Optimization Over Continuous Spaces". ICSI, USA, Technical Report, TR-95-012, 1995.
- [22] Liu H, Cai Z, Wang Y. "Hybridizing Particle Swarm Optimization With Differential Evolution for Constrained Numerical and Engineering Optimization". *Applied Soft Computing*, 10(2), 629-640, 2010.
- [23] Suwansirikul C, Friesz TL, Tobin RL. "Equilibrium Decomposed Optimisation: A Heuristic for the Continuous Equilibrium Network Design Problem". *Transportation Science*, 21(4), 254-263, 1987.
- [24] Bureau of Public Roads. "Traffic Assignment Manual". Department of Commerce, Washington DC, USA, 1964.