



Zaman pencereleli çok araçlı dağıtım toplamalı rotalama problemi için gerçek değerli genetik algoritma yaklaşımı

Barış Kiremitçi¹

*Ulaştırma ve Lojistik Yönetimi,
Ulaştırma ve Lojistik Yüksekokulu,
İstanbul Üniversitesi, İstanbul, Türkiye*

Serap Kiremitçi²

*Ulaştırma ve Lojistik Yönetimi,
Ulaştırma ve Lojistik Yüksekokulu,
İstanbul Üniversitesi, İstanbul, Türkiye*

Timur Kesintürk³

*Sayısal Yöntemler,
İşletme Fakültesi,
İstanbul Üniversitesi, İstanbul, Türkiye*

Özet

Bu çalışmada; çok araçlı, dağıtım toplamalı, zaman pencereleli rotalama problemlerinin, gerçek değerli kodlamalı genetik algoritma ile çözümü ele alınmıştır. Problemden rotalar, kapasite, zaman pencereleri, eşleşme ve öncelik kısıtları dikkate alınarak oluşturulmaktadır. Amaç fonksiyonu, toplam mesafenin minimizasyonu, araç sayısının minimizasyonu veya her ikisi birlikte olacak şekilde belirlenebilmektedir. Gerçek hayatta problemin geniş bir uygulama sahası olmasına rağmen araç rotalama literatüründe, problemin zorluğundan dolayı, çok fazla yayın yer almamaktadır. Çalışmamızda probleme özgün yeni bir gerçek değerli kodlamalı genetik algoritma geliştirilmiştir. Probleme ait değişkenler farklı bir yapıda, gerçek değerlerle kodlanmıştır. Böylelikle daha küçük boyutlu kromozomlarla, daha az değişkenle çözüm prosesi geliştirilmeye çalışılmıştır. Algoritma literatürdeki bir kısım problemler üzerinde denenmiş ve mevcut algoritmalar ile performans karşılaştırılması yapılmıştır.

Anahtar Kelimeler: Araç Rotalama, Genetik Algoritma, Dağıtım Toplamalı Zaman Pencereleli, Optimizasyon

A real valued genetic algorithm approach for the multiple vehicle pickup and delivery problem with time windows

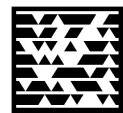
Abstract

The Multiple Vehicle Pickup and Delivery Problem with Time Windows (MV_PDPTW) which constitutes an important variant of the vehicle routing problems, deals with goods that have to be transported from origin to the destination points. In this problem, routes are designed in order to satisfy capacity, time windows, coupling and precedence constraints with the aim of minimization of total costs (which can be total distance, number of vehicles or both of them). Although many real life operations in logistics and transportation management can be modeled as MV_PDPTW, it has relatively less attention among vehicle routing literature because of its difficulty. In this paper we propose a real valued genetic algorithm approach to solve MV_PDPTW. Problem variables are presented by real valued chromosomes. By this way we assume to use less genes which improve search process. Proposed genetic algorithm approach has been tested on available benchmark problem sets and has compared with three previous GA results.

¹ baris@istanbul.edu.tr (B. Kiremitçi)

² serapy@istanbul.edu.tr (S. Kiremitçi)

³ tkturk@istanbul.edu.tr (T. Kesintürk)



Keywords: Vehicle Routing, Genetic Algorithm, Pickup and Delivery with Time Windows, Optimization

1. Giriş

Küresel pazarlarda kısa hayat döngüsüne sahip olan ürünlerin varlığı ve müşterilerin her geçen gün artan beklentileri üretim yapan işletmeleri lojistik sistemlerine yoğunlaşmaya ve bu sistemler üzerine yatırımlar yapmaya zorlamaktadır. İletişim ve ulaştırma teknolojilerinde meydana gelen gelişmeler (örneğin mobil internet, gecelik teslimatlar) lojistik sistemlerin sürekli geliştirilmesini zorunlu kılmıştır [1].

Ulaştırma ve lojistik faaliyetleri insan çabasının merkezinde yer almaktadır ve diğer sosyal ve ekonomik faaliyetleri desteklemesi nedeniyle önemlidir [2, 3]. Çok çeşitli oyuncuların farklı karar düzeylerini, belirsizlikleri önemli derecede sermaye harcamalarını kapsayan ulaştırma oldukça karmaşık bir alandır. Rekabete dayanabilmek için bu alanda karar vericiler güçlü bilgisayar ve iletişim teknolojileri kadar büyük miktarlarda veri, karmaşık matematiksel modeller ve optimizasyon tekniklerine çok daha fazla güvenmek durumundadırlar. Bu alanın çeşitliliği ve karmaşıklığı, çalışma alanlarının zenginliği, çeşitli yöntemler ve yazılımlar vasıtasıyla yansımaktadır [3].

Çoğu lojistik yapıların büyük bir bölümü depolara, perakendecilere veya müşterilere hizmet sunan araçlardan oluşan filonun yönetilmesini kapsar. Filo işletme maliyetlerini kontrol edebilmek için her bir araca ne kadar yükleme yapılması gerektiği, aracın ne zaman nereye gönderileceği ile ilgili olarak sürekli karar vermek gerekmektedir. Bu tip kararlar araç rotalama problemi kapsamına girmektedir [1].

Araç rotalama problemleri dağıtım yönetiminin merkezinde yer almaktadır. Binlerce firma ve organizasyon her gün çeşitli ürünlerin toplanması, teslim edilmesi veya insanların bir yerden bir yere taşınması ile karşılaşmaktadır. Pratikte karşılaşılan kısıtlar ve amaçlar oldukça değişken ve farklı olduğundan her bir işletme içi koşullar farklıdır [4].

Mal ve hizmet dağıtımlarının etkin ve etkili bir şekilde yönetilmesi hem kamu hem de özel sektörde önemli bir yere sahiptir. Birçok ulaştırma ve sistem maliyetlerinin önemli bir bölümü araçların rotalanması ve çizelgelenmesi ile ilgilidir [5]. Araçların etkili rotalanması ve çizelgelenmesi verimliliği arttırarak uzunlu dönemli planlamalara yardımcı olup işletmelere çok yüksek oranlarda tasarruf imkanı sağlayabilmektedir [6].

Araç rotalama problemi, talepleri bilinen müşteriler kümesine hizmet veren başlangıç ve bitiş noktası merkezi bir depo olan araç filosu için minimum maliyetli rotalar kümesinin belirlenmesiyle ilgilidir. Her müşteriye bir seferde hizmet verilmeli ve araçların kapasiteleri dikkate alınarak tüm müşteriler araçlara atanmalıdır. Araç rotalama problemine müşterilerin, son teslim tarihi veya en erken teslim zamanı gibi kısıtları eklemelerinden kaynaklanan kabul edilebilir teslim zamanları veya zaman pencerelerinin karmaşıklığı eklendiğinde problem araç rotalama ve çizelgeleme problemine dönüşür. Araç rotalama problemleri araç hareketlerinin konumsal yönüyle ilgilenirken araç rotalama ve çizelgeleme problemleri araç hareketlerinin hem konumsal hem de zamansal yönüyle ilgilidir [6].

Rota, bir aracın ardışık olarak ziyaret edeceği düğümlerin arka arkaya sıralanmasıdır. Rotalardaki düğümlerin sıralanması ile beraber araçların kalkış veya varış zamanlarının da eklenmesiyle rotalama ve çizelgeleme problemi elde edilmiş olur [7].

Bir başka tanımda aracın rotası, başlangıç ve bitiş noktası depo olan bir aracın sırasıyla geçeceği toplama ve/veya dağıtım noktalarının sıralanması olarak verilmiştir. Bir aracın çizelgesi, toplama veya dağıtım noktalarının sıralanması ile birlikte ilgili kalkış ve varış zamanlarının kümesidir. Araç noktalardan (müşterilerden) belirlenmiş sırada belirlenen zamanlarda geçmelidir. Noktalardaki varış zamanları önceden sabit olduğunda (örneğin

toplu taşıma sistemlerinde araçların ve sürücülerin çizelgelenmesi) problem çizelgeleme problemi olarak adlandırılmaktadır [8].

Varış zamanları belirlenmediği durumlarda problem doğrudan rotalama problemi olmaktadır. Zaman pencereleri veya öncelik ilişkileri bulunduğu durumlarda, rotalama ve çizelgeleme fonksiyonlarının her ikisinin birlikte gerçekleştirilmesi gerekir. Bu tip birleşik rotalama ve çizelgeleme problemleri ile uygulamada sıklıkla karşılaşmaktadır [6]. Lawrence Bodin, 1990 yılında yazdığı "Rotalama ve çizelgelemenin 20 yılı" başlıklı çalışmasında 2000'li yıllarda araç rotalama ve çizelgeleme sistemlerinin işletmelerin temel lojistik, dağıtım sistemlerinin önemli bir parçası olarak görüleceğini ifade etmiştir [8].

Bu çalışmada, Zaman Pencereli Çok Araçlı Dağıtım Toplamalı Rotalama Problemi (ZP-ÇDTRP) için bir metasezgisel olan Genetik Algoritma (GA) ile çözüm aranmıştır. Literatürdeki GA çözümlerinden farklı olarak yeni bir kodlama şekli ile problem çözülmeye çalışılmıştır. Amaç problem temsilinin daha az değişkenle yapılması ve çözüm uzayının daha etkin aranmasıdır. Bir sonraki bölümde, araç rotalama problemlerinden ve özelliklerinden bahsedilecektir. Üçüncü bölüm çalışmamıza konu olan ZP-ÇDTRP problemi tanıtılacak, matematiksel modeli verilecektir. Sonraki bölüm, problem için geliştirilen yeni GA yaklaşımını içermektedir. Son bölümde ise yöntemin performansı test edilmiş ve karşılaştırmalı sonuçlar verilmiştir. Aynı zamanda sonuçların yorumları, gelecekte yapılabilecek çalışmalar da bu bölümde yer almıştır.

2. Araç Rotalama Problemi

Araç rotalama problemi (ARP) ilk olarak Dantzig ve Ramser tarafından 1959 yılında yazdıkları çalışma ile ortaya çıkmıştır [9]. ARP bugün hiç olmadığı kadar popülerdir ve hakkında oldukça zengin bir bilimsel yayın literatürü vardır [10]. Ekşioğlu ve arkadaşlarının yaptıkları "Araç Rotalama Problemi: Sınıflandırma Taraması" isimli yayınlarında "araç rotalama" terimi ile yaptıkları araştırma sayılarının görülmesi açısından incelenebilir [11].

Bilgisayarların optimizasyon problemlerinin çözümünde kullanılmaya başlanması ARP tipi bileşi optimizasyon problemlerinin daha verimli bir şekilde çözülmesine olanak sağlamıştır. Hesaplama gücünün artması bir çok araştırmacıya daha önce çözülememiş büyüklükte ARP probleminin çözüme imkânı vermiştir.

Yöneylem araştırması literatüründe ARP'nin bir çok ilginç uygulaması vardır. Uygulamaların pek çoğu karayolu araçlarını içerse de gemiler, römorkör, helikopter gibi diğer taşıma modlarına ait taşıtlarda uygulamalarda yer bulmuştur.

Günümüzde araç rotalama uygulamalarına çokça rastlanmaktadır. Uygulamalar bir çok farklı endüstriyi kapsamaktadır. Gazete, yiyecek, içecek gibi birçok ürün çeşidinin ticari dağıtımının günlük olarak yapılması gerekmektedir. Ticari dağıtım yapıları haricinde, atıkların toplanması, sokak süpürme ve kargo teslimi gibi uygulamalar da vardır [12, 13]. Partyka and Hall 2000). Banka ATM makinelerine nakit teslimi ve çizelgelenmesi, petrolün dinamik tedarik edilmesi ve taşınması, restoranlardan atık yağların toplanması, ev aletleri tamir hizmet ve teslimi, evlere internet tabanlı yiyecek teslimi, süt toplama ve stok yönetimi, evlerden yardım bağışlarının toplanması, portatif tuvalet teslimatı, hapishanelerle mahkemeler arasında hükümlülerin taşınması, engellilerin minübüs ve taksilerle taşınması, toptancı depolarından perakendecilere ürün dağıtımı, posta teslimi yapan araçların rotalanması da uygulamalara örnek olarak verilebilir [13].

2.1 Araç Rotalama Probleminin Özellikleri

ARP, çok bilinen zor ve önemli bir bileşi (combinatorial) optimizasyon problemidir [3, 4]. ARP, gezgin satıcı probleminin geliştirilmiş, gerçekçi kısıtlamalara sahip halidir [14, 15].

ARP, gezgin satıcı probleminden farklı olarak birden fazla araç içermektedir. Üstelik bu araçların kapasiteleri de gezgin satıcı problemindeki gibi sınırsız değildir.

ARP, müşteriler kümesine hizmet götürecek olan araç filosunun takip edeceği optimal rotalar kümesinin belirlenmesi olarak da ifade edilmiştir [16]. Talebi bilinen müşteriler kümesini kapsayacak, başlangıç ve bitiş noktası depo olan rotalar kümesi bulunmaya çalışılırken amaç, kat edilen toplam mesafeyi, kullanılan araç sayısını, her ikisinin kombinasyonunu veya toplam maliyeti minimize etmek olabilir [17]. Araç rotalama probleminin temel bileşenleri müşteriler, depolar, araçlar, sürücüler ve yol şebekesidir.

Müşteriler, şebekede düğümlerde gösterilirler. Müşterinin talebi, müşteriye teslim edilmesi gereken veya alınması gereken farklı türlerde de olabilen ürün miktarlarıdır. Müşterilere günün/ayın belirli zamanlarında (zaman pencerelerinde) hizmet vermek mümkündür.

Depo/depolar ise araçların rotalarının başlangıç ve bitiş noktasıdır. Şebekede müşteriler gibi depolar da düğümlerde gösterilirler. Müşterilere ürünlerin dağıtımını gerçekleştiren araçların kapasiteleri taşıyabilecekleri maksimum ağırlık veya hacim veya palet olarak ifade edilebilir. Araçların oluşturmuş olduğu filo homojen veya heterojen olabilmektedir. Homojen bir filoda, araçların hızları, sabit maliyetleri, değişken maliyetleri, ekipmanları ve büyüklükleri denktir. Heterojen bir filo ise farklı özelliklere sahip araçlardan oluşmaktadır [16].

Araç rotalama problemlerinin çeşitli hatta bazen çelişen amaçları olabilir. Bunlardan en önemlileri:

- Toplam seyahat mesafesine (veya toplam seyahat süresine) ve filoda kullanılan araçların sabit maliyetlerine dayanan toplam taşıma maliyetlerinin minimizasyonu
- Tüm müşterilere hizmet vermek için gereken araç sayısının minimizasyonu
- Seyahat süresi ve araç yükü bakımından rotaların dengelenmesi
- Müşterilerin parçalı hizmet görmesiyle ilgili cezaların minimizasyonu veya
- Bu amaçların kombinasyonlarıdır [16].

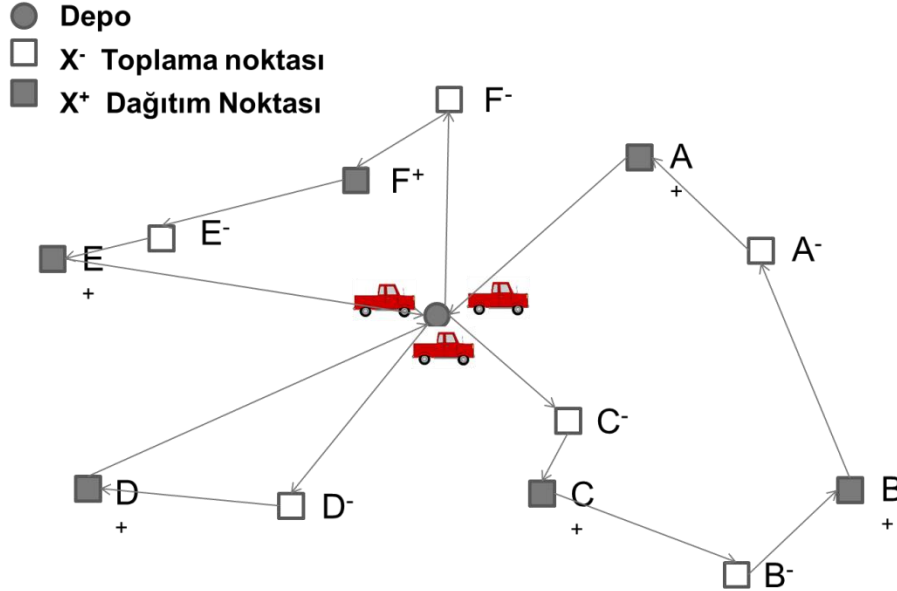
Literatürde ARP problemleri birçok temel türe göre sınıflandırılmaktadır. En önemlileri; kapasite kısıtlı, mesafe kısıtlı, zaman pencereli, geri toplamalı, dağıtım ve toplamalı araç rotalama problemleridir. Her bir temel türün, ilave kısıtlara ve farklı özelliklere sahip alt türleri de literatürde yer almaktadır [16].

Bu çalışmada incelenecek olan ARP türü, Zaman Pencereli Dağıtım ve Toplamalı Araç Rotalama Problemidir (ZPDT-ARP veya PDPTW-Pickup and Delivery with Time Windows). Devam eden bölümde bu problem hakkında kısa bilgi verilecektir.

3. Zaman Pencereli Çok Araçlı Dağıtım Toplamalı Rotalama Problemi (ZP-ÇDTRP)

Bu problem insanların veya eşyaların kaynak veya hedef noktalar arasında taşınmak zorunda olduğu araç rotalama problemlerinin en önemli sınıfını oluşturur [18]. Bu problem türünde, taşıma taleplerini karşılayacak şekilde rotalar oluşturulmakta ve bu rotalara filo içindeki araçlara atanmaktadır. Her aracın sahip olduğu özellikler farklı olabilmekte ve bu yüzden kapasite kısıtları ortaya çıkmaktadır. Her bir taşıma için talep, taşınacak yük miktarı, yükün alınacağı (toplama-pick up) kaynak noktası ve yükün teslim edileceği (dağıtılacağı) hedef nokta belirlenir. Bu problemde her bir yük başka bir konumda aktarma yapılmaksızın kendi kaynağından kendi hedefine sadece bir tek araç ile taşınmalıdır [19]. Bir aracın rotası genellikle merkezi bir depoda başlar ve aynı şekilde bir depoda biter. Bir taşıma talebi, ilgili teslimat noktasına götürülmek üzere toplama noktasından alınmalıdır. Toplama ve dağıtım çiftine aynı araç hizmet vermelidir ve toplama dağıtımdan önce gelmelidir. Her bir taşıma talebine önceden belirlenmiş zaman penceresi içinde hizmet

verilmelidir (bu kısıt zaman penceresi olarak isimlendirilmektedir). Problemin çözümü taşıma taleplerinin araçlara atanmasını ve toplam maliyeti minimize eden her araç için rotanın bulunmasını gerektirir [20]. Dağıtım ve toplama problemlerinde tek araçlı ve çok araçlı olmak üzere de bir ayrım vardır [21]. Bu çalışmada çok araçlı model incelenecektir. Problemin temsili gösterimi Şekil 1'de yer almaktadır.



Şekil 1 ZP-ÇDTRP Probleminin Temsili.

Taşınan yükün insan olması durumunda, müşteri memnuniyetsizliğini azaltmak için probleme ilave kısıtlar da eklenebilir. Özellikle yolcunun araçta geçirdiği zamanı sınırlayan seyahat süresi kısıtları [22] güncel hayatta sıklıkla karşılaşılabilecek bir durumdur.

3.1 Matematiksel Model

Taşıma taleplerinin sayısı n ile gösterilsin. ZP-ÇDTRP problemi yönlü $G = (N, A)$ çizgesi üzerinde tanımlanır. $N = \{0, 1, 2, \dots, 2n + 1\}$ düğümler kümesi ve A 'da bağlantıları içeren kümedir. 0 ve $2n + 1$ düğümleri başlangıç ve bitiş deposunu gösterirken, $P = \{1, \dots, n\}$ ve $D = \{n + 1, \dots, 2n\}$ altkümeleri sırasıyla toplama ve dağıtım düğümlerini temsil ederler. Bu yüzden her bir i taşıma talebi, toplama düğümü i ve teslim düğümü $n + i$ ile ilişkilidir. Herbir $i \in N$ düğümü, yük miktarı q_i ve negative olamayan hizmet süresi d_i ile ilgilidir. Depolara ait yük miktarları 0 (sıfır) olarak kabul edilmiş ($q_0 = q_{(2n+1)} = 0$); dağıtım düğümlerindeki yük miktarları ilgili toplama düğümlerinin negatifi olarak alınmıştır ($q_i = -q_{(n+i)}$ ($i = 1, \dots, n$)) ve depolardaki hizmet süreleri de sıfır kabul edilmiştir ($d_0 = d_{(2n+1)} = 0$). Her bir $i \in P \cup U$ düğümü ile ilgili $[a_i, b_i]$ zaman penceresi, i düğümünde hizmetin başlayabileceği a_i en erken zamanı ve b_i en geç zamanı gösterir. Depo düğümlerinin de zaman pencereleri vardır. $[a_0, b_0]$ depoyu terk etmek için ve $[a_{(2n+1)}, b_{(2n+1)}]$ depoya dönüş için en erken ve en geç zamanı gösterir. K araçlar kümesini gösterir. Araçların hepsinin özdeş ve Q kapasitesine sahip olduğu varsayılmıştır. Her bir $(i, j) \in A$ bağlantısı ile ilgili, rota maliyeti $c_{(ij)}$ ve seyahat süresi $t_{(ij)}$ vardır. Ayrıca seyahat süresi $t_{(ij)}$ 'nin i

düğümündeki hizmet süresi d_i ' yi içerdiği ve tüm rota maliyetleri ve seyahat sürelerinin üçgen eşitsizliğini (the triangle inequality) taşıdığı varsayılmıştır.

$x_{(ij)}^k$ ikili değişkeni, k aracı i düğümünden doğrudan j düğümüne gidiyor ise 1 değilse sıfırdır. B_i^k , i düğümünde k aracının hizmete başlama zamanını, Q_i^k , k aracı i düğümünü terk ettiğindeki yük miktarını gösteren değişkenlerdir. Bu değişkenler kullanılarak, PDPTW lineer olmayan karma-tamsayılı programlama modeli olarak aşağıdaki gibi gösterilebilir [23].

Amaç Denklemi:

$$\min \sum_{k \in K} \sum_{i \in N} \sum_{j \in N} c_{ij} x_{ijk} \quad (1)$$

Kısıtlar:

$$\sum_{k \in K} \sum_{j \in N} x_{ijk} = 1, \quad \forall i \in P, \quad (2)$$

$$\sum_{j \in N_k} x_{ijk} - \sum_{j \in N} x_{n+i,jk} = 0, \quad \forall k \in K, i \in P, \quad (3)$$

$$\sum_{j \in N} x_{o,j,k} = 1, \quad \forall k \in K, \quad (4)$$

$$\sum_{j \in N} x_{jik} - \sum_{j \in N} x_{ijk} = 0, \quad \forall k \in K, i \in P \cup D, \quad (5)$$

$$\sum_{i \in N} x_{i,2n+1,k} = 1, \quad \forall k \in K, \quad (6)$$

$$B_{jk} \geq (B_{ik} + t_{ij}) * x_{ijk} \quad \forall k \in K, i \in N, j \in N, \quad (7)$$

$$Q_{jk} \geq (Q_{ik} + q_j) * x_{ijk} \quad \forall k \in K, i \in N, j \in N, \quad (8)$$

$$B_{ik} + t_{i,n+i} \leq B_{n+i,k} \quad \forall i \in P, \quad (9)$$

$$a_i \leq B_{ik} \leq b_i \quad \forall i \in N, k \in K, \quad (10)$$

$$\max\{0, qi\} \leq Q_{ik} \leq \min\{Q, Q + q_i\} \quad \forall i \in N, k \in K, \quad (11)$$

$$x_{ijk} \in \{0,1\} \quad \forall i \in N, j \in N, k \in K,$$

Amaç denklemi toplam rotalama maliyetinin minimize edilmesini sağlar. Denklem 2 ve denklem 3 her bir talebin tam olarak bir kez hizmet görmesini ve toplama ve dağıtım düğümlerine aynı aracın hizmet sunmasını sağlar. Denklem 4-6 her bir k aracının rotasının depodan başlamasını ve depoda bitmesini garanti eder. Zaman ve yük değişkenlerinin tutarlılığı denklem 7 ve 8 ile sağlanır. Denklem 9 her bir i talebi için toplama düğümünün, dağıtım düğümünden önce ziyaret edilmesini sağlar. Son olarak, denklem 10 ve 11 kısıtları sırasıyla, zaman pencerelerini ve kapasite kısıtlarını yürürlüğe koyar.

4. ZP-ÇDTRP Problemi için yeni bir GA yaklaşımı

Genetik Algoritmalar (GA) çözümü zor problemler için geliştirilmiş popülasyon temelli bir metasezgiseldir [24-26]. Probleme ait değişkenler, kromozom vektörlerinin genlerinde temsil edilmektedir. Seçim, çaprazlama ve mutasyon olarak adlandırılan genetik operatörler, iterasyonlar boyunca kromozomlarda birtakım değişiklikler yapmakta ve en iyi sonucu verecek çözüm seti aranmaktadır. Seçim, daha iyi çözümlerin sonraki iterasyonlar için yaşama şansını arttıran, daha kötü sonuçları eleyen operatördür. Çaprazlamada kromozomlar arası bilgi değişimi yapılarak, daha iyi bireyler elde edilmeye çalışılır. Mutasyon ise, algoritmanın lokal optimumlara takılmasını önleyen, kromozomda çok küçük

değişiklikler yapan operatördür. Farklı problemler için kullanılan farklı genetik operatör çeşitleri bulunmaktadır [25]. Bu operatörler her iterasyonda uygulanarak global optimum aranır. Global optimum garanti edilmese de iyi bir çözümü kabul edilebilir zamanlarda bulunmaktadır.

Çalışmamızda ele aldığımız ZP-ÇDTRP problemi için daha önce yapılan çalışmalardan bir kısmı GA ile çözümü denemiştir. Bunlar farklı kodlama yapıları kullanmış ve sonuçlarını raporlamışlardır. Problem genel olarak iki alt problemin çözümü olarak düşünülebilmektedir. Bunlardan ilki müşterilerin gruplandırılması veya sınıflandırılması; ikincisi ise araçlara atanmış bu müşterilerin rotalandırılmasıdır. Bu iki problemin eşzamanlı olarak çözümünün GA ile yapılması bir zorluk içermektedir.

Bugüne kadar yapılan ilgili çalışmalarda bu zorluğu yenmek için farklı kromozom yapıları ele alınmıştır. Jorgensen ve diğerleri [27] ve Pankratz [28] çalışmalarında GA'yı problemin ilk alt problemi olan gruplama kısmı için kullanmışlardır. Rotalama kısmı için ise farklı sezgiseller, bağımsız rotalama algoritmaları kullanmışlardır. Créput ve diğerleri [29] problemin her iki alt problemini temsil edecek bir GA kodlaması ile çözüm aramışlar ancak performans açısından yeterli olamamışlardır. Hosny ve Mumford [20] çalışmalarında her iki alt problemi de temsil eden GA kodlaması kullanmış ve sonuçlarını grafik üzerinde göstermişlerdir. İlgili grafiğe göre, Hosny ve Mumford şu ana kadar ZP-ÇDTRP probleminin çözümü için GA yaklaşımı kullanılan çalışmalar içerisinde en başarılı sonuçları elde ettiklerini belirtmişlerdir. Nagata ve Kobayashi [30] yaptıkları çalışmada çaprazlama operatörünün bu tip problemlerdeki zorluğunu dikkate alarak yeni çaprazlama tipleri üzerinde durmuşlardır.

Çalışmamızda, problem için yeni bir kromozom yapısı önerilmiştir. Tablo 1'de LC101 isimli problem verilerinde de görülen bilgiler tarafımızca farklı şekilde düzenlenmiştir (<http://www.sintef.no/Projectweb/TOP/PDPTW/Li--Lim-benchmark/>). Buna göre kromozomun satırları tanımladığımız işleri temsil etmektedir. İş tanımlanırken, her bir taşıma talebi ile ilgili toplama ve dağıtım bilgileri birleştirilmiştir. Yani 1 nolu iş talebi Tablo 2'den de görüleceği üzere 11 nolu müşteriden alınan (pickup) 10 birimin 1 nolu müşteriye müşterilere ait uygun zaman pencerelerinde taşınması (delivery) olarak düşünülmüştür. ikincisi ilgili işin toplama zamanının ve üçüncüsü ise yine aynı işin dağıtım zamanını temsil etmektedir. Her satırdaki 2. Sütun (Araç) ilgili işi gerçekleştirecek aracı, 3. Sütun (Toplama (t)) , ilgili işin toplama müşterisinden alınma zamanını ve 4. Sütun (Dağıtım (t)) ise yine ilgili işin dağıtım müşterisine getirilme zamanını temsil etmektedir.

Tablo 1 Probleme Ait Orjinal Veri Seti

| CUSTOMER NO. | XCOORD. | YCOORD. | DEMAND/LOAD | EARLIEST PICKUP /DELIVERY TIME | LATEST PICKUP / DELIVERY TIME | SERVICE TIME | PICKUP[2] | DELIVERY[3] |
|--------------|---------|---------|-------------|--------------------------------|-------------------------------|--------------|-----------|-------------|
| 0 | 40 | 50 | 0 | 0 | 1236 | 0 | 0 | 0 |
| 1 | 45 | 68 | -10 | 912 | 967 | 90 | 11 | 0 |
| 2 | 45 | 70 | -20 | 825 | 870 | 90 | 6 | 0 |
| 3 | 42 | 66 | 10 | 65 | 146 | 90 | 0 | 75 |
| 4 | 42 | 68 | -10 | 727 | 782 | 90 | 9 | 0 |
| 5 | 42 | 65 | 10 | 15 | 67 | 90 | 0 | 7 |
| 6 | 40 | 69 | 20 | 621 | 702 | 90 | 0 | 2 |
| 7 | 40 | 66 | -10 | 170 | 225 | 90 | 5 | 0 |
| 8 | 38 | 68 | 20 | 255 | 324 | 90 | 0 | 10 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |

| İş | Araç | Toplama (t) | Dağıtım (t) |
|----|------|-------------|-------------|
| 1 | 1 | 201 | 858 |
| 2 | 4 | 144 | 795 |
| 3 | 3 | 229 | 495 |
| 4 | 2 | 203 | 514 |
| 5 | 5 | 101 | 930 |
| 6 | 2 | 273 | 403 |
| 7 | 2 | 204 | 455 |
| 8 | 3 | 208 | 731 |
| 9 | 4 | 224 | 697 |
| 10 | 1 | 187 | 470 |
| 11 | 5 | 282 | 686 |
| 12 | 1 | 214 | 647 |
| 13 | 3 | 156 | 964 |
| 14 | 4 | 224 | 733 |
| 15 | 5 | 187 | 886 |
| 16 | 2 | 137 | 594 |
| 17 | 1 | 133 | 588 |
| 18 | 3 | 149 | 995 |

Şekil 2 ZP-ÇDTRP Problemi İçin Geliştirilen Yeni GA Kromozomu.

Şekil 2'ye göre birinci işi temsil eden ilk satırın açılımı şu şekildedir: İlk iş birinci araç tarafından ziyaret edilecektir. Birinci işin toplama noktasındaki müşteriye toplama için 201 birim zamanında ve dağıtım noktasındaki müşteriye ise 858 birim zamanında uğranacaktır. Bu kromozom yapısı sayesinde, popülasyondaki kromozomlara ait boyutlar iterasyonlar boyunca sabit kalmakta ve probleme ait iki alt problem de aynı kromozom içerisinde temsil edilmektedir. İlk sütundaki farklı numaraların sayısı, çözüme ait araç sayısını vermektedir. Araçların rotaları ise ilgili aracın toplama ve dağıtım yaptığı müşterilerin, ikinci ve üçüncü sütunlarındaki toplama ve dağıtım zamanları ile temsil edilmektedir. Buna göre birinci aracın rotasındaki işler aşağıdaki gibi olacaktır:

17 - 10 - 1 - 12

Başlangıç popülasyonunun oluşturulması şu şekilde olmaktadır. Her müşteri için araç atanması iki farklı şekilde olabilmektedir. Eğer araç sayısı belli ise her müşteri için atanacak araç numarası 1 ile araç sayısı arasında tesadüfi olarak belirlenmektedir. Eğer araç sayısı değişkense veya amaç fonksiyonunda araç sayısının minimizasyonu söz konusu ise müşteri kadar araç atanarak çözüme başlanmaktadır. İterasyonlar boyunca GA operatörleri ile araç sayısı ilk sütun değerleri değiştirilerek azaltılmaktadır. Toplama ve dağıtım sürelerinin temsil edildiği ikinci ve üçüncü sütun değerleri ise her müşterinin toplama ve dağıtım zaman aralıkları arasında yine tesadüfi olarak belirlenmektedir. Burada, aracın bekleme yapabilmesine de olanak sağlamak ve çözüm alternatiflerini geniş tutabilmek için erken gelişler de kabul edilmektedir. Buna göre yine GA operatörleri ile hem toplama hem de dağıtım için, diğer kısıtları sağlamaları koşulu ile en erken hizmet süresinden öncesinde toplama ya da dağıtım yapmasına izin verilmektedir.

Seçim operatörü olarak rulet tekerleği seçim operatörü kullanılmıştır [31]. Çaprazlama ve mutasyon operatörleri probleme özgü geliştirilen kromozom yapısına uygun olarak tasarlanmıştır. Buna göre çaprazlama için satırlardan yapılacak şekilde bir nokta ya da iki nokta çaprazlama kullanılmıştır. Diğer kodlama biçimleri ile çalışan GA'larda bu aşamada uygunluk testi yapılmasına rağmen geliştirilen temsil tipinde buna ihtiyaç olmamaktadır. Çünkü yapılan çaprazlama ile yine tüm müşterilerin ziyaretleri gerçekleştirilmekte, yalnızca

ziyaret edecek olan araç ve ziyaret zamanları değişmektedir. Turlarda yeni üretilen bireylerde, ilk sütundaki araç numaraları ve ziyaret zamanları dikkate alınarak güncellenmektedir. Mutasyon da yine kromozom yapısına uygun olarak geliştirilmiş iki alt operatörden oluşmaktadır. İlk sütunla ilgili mutasyonda, müşterilerin ziyaretini gerçekleştiren araçlar belli bir olasılıkla değiştirilmektedir. Bu herhangi bir müşterinin aracının değiştirilmesi olabileceği gibi, bir aracın kaldırılarak mevcut başka araç ya da araçlara, ilgili müşterilerin atanması şeklinde olabilmektedir. İkinci kısım mutasyonda ise müşterilerin ziyaret zamanları yine belli bir olasılıkla değiştirilmektedir. Burada ilgili müşterinin zaman pencereleri dikkate alınarak değişiklik yapılmakta, böylelikle uygun olmayan çözümlerin oluşması engellenmektedir. Şekil 3'te geliştirilen kodlama biçimine yönelik oluşturulan çaprazlama ve mutasyon operatörleri temsilen gösterilmiştir.

| | | | | | | | | |
|---|-----|-----|---|-----|-----|---|-----|-----|
| 1 | 201 | 858 | 5 | 202 | 637 | 1 | 201 | 858 |
| 4 | 144 | 795 | 1 | 140 | 884 | 1 | 144 | 795 |
| 3 | 229 | 495 | 3 | 276 | 577 | 3 | 229 | 495 |
| 2 | 203 | 514 | 2 | 296 | 732 | 2 | 203 | 514 |
| 5 | 101 | 930 | 3 | 171 | 887 | 5 | 101 | 930 |
| 2 | 273 | 403 | 1 | 137 | 472 | 1 | 137 | 472 |
| 2 | 204 | 455 | 3 | 285 | 461 | 3 | 285 | 461 |
| 3 | 208 | 731 | 4 | 195 | 832 | 1 | 195 | 422 |
| 4 | 224 | 697 | 4 | 222 | 504 | 1 | 222 | 504 |
| 1 | 187 | 470 | 1 | 105 | 539 | 3 | 105 | 539 |
| 5 | 282 | 686 | 5 | 201 | 901 | 5 | 201 | 901 |
| 1 | 214 | 647 | 5 | 114 | 937 | 5 | 229 | 937 |
| 3 | 156 | 964 | 3 | 215 | 849 | 3 | 215 | 849 |
| 4 | 224 | 733 | 5 | 238 | 962 | 5 | 238 | 962 |
| 5 | 187 | 886 | 4 | 197 | 695 | 1 | 197 | 695 |
| 2 | 137 | 594 | 3 | 221 | 704 | 3 | 221 | 704 |
| 1 | 133 | 588 | 3 | 180 | 986 | 3 | 180 | 986 |
| 3 | 149 | 995 | 4 | 129 | 581 | 1 | 129 | 581 |

Şekil 3 Çaprazlama ve Mutasyon Operatörleri.

Şekil 3'e göre seçilen ebeveyn kromozomlar tek nokta çaprazlama ile 5. Satırdan çaprazlanmıştır. Oluşan yeni kromozomda, ilk ebeveyn kromozomun ilk 5 satırı ve ikinci ebeveyn kromozomun 6. satırdan itibaren kalan tüm satırları yer almıştır. İlk satırdaki araç atamaları ile ilgili olarak iki çeşit mutasyon uygulanmıştır. İlk olarak 4 numaralı araç seçilmiş ve bu aracın ziyaret ettiği şehirler 1. araca atanmıştır. İkinci olarak yine tesadüfen seçilen bir işe ait araç ataması olan 1. araç yerine 3. araç atanmıştır. Ziyaret zamanları ile ilgili yapılan mutasyonda ise 8. işin dağıtım zamanı 832'den 422'ye değiştirilmiş ve 12. işin toplama zamanı ise 114'ten 229'a değiştirilmiştir.

Bu kodlama biçimi ile uygunluğun kontrol edilmesi ile ilgili yapılan birçok işlem ortadan kaldırılmış ve değişen boyutlardaki turların saklandığı matrislerin çözümlenmesi işlemleri azaltılmıştır. Bir sonraki bölümde geliştirilen GA literatürdeki problemler ile test edilmiş ve mevcut GA çözümleri ile karşılaştırılmıştır.

5. Uygulama

Çalışmada, kullanılan yeni gerçek değer kodlamalı GA literatürde yer alan diğer GA yöntemleri ve en iyi bilinen değerleri veren çalışmalarla karşılaştırılmıştır. Karşılaştırma için kullanılan problemler <http://www.sintef.no/Projectweb/TOP/PDPTW/Li--Lim-benchmark/> adresinde de yer alan Li ve Lim'in [32] geliştirmiş oldukları problemlerden bazılarıdır. Yazarlar, LC, LR ve LRC tipi üç sınıf problem üretmişlerdir. LC'de kümelenmiş lokasyonlar, LR'de tesadüfi olarak dağıtılmış lokasyonlar ve LRC'de ise tesadüfi olarak ve kümelenmiş

lokasyonlar kullanılmıştır. Çalışmamızda LC tipi problemlerin ilk 9'u ele alınmış ve mevcut yöntemlerden ikisi ile kıyaslanmıştır.

Genetik algoritma için belirlenen parametreler: iterasyon sayısı popülasyon büyüklüğü, çaprazlama ve mutasyon olasılıkları için sırasıyla 10000, 50, 0,9 ve 0,05'tir. GA'ya ait kodlar Matlab programlama dili ile yazılmış ve Windows 7 on Intel(R) Core(TM) 2 Duo CPU, 2.10 GHz and 3 GB RAM özelliklere sahip bir bilgisayarda çalıştırılmıştır. Uygulamaya ait sonuçlar Tablo 2'de yer almaktadır.

Tablo 2 Uygulama Sonuçları

| Problem | Bilinen en iyi | Li & Lim [32] | Pankratz (GGA) [32] | RV-GA |
|---------|----------------|------------------|---------------------------|----------------|
| lc101 | 828.94 | 828.94 | 828.94 | 828.94 |
| lc102 | 828.94 | 828.94 | 828.94 | 828.94 |
| lc103 | 827.86 | 827.86 | 827.86 | 842.06 |
| lc104 | 818.60 | 861.95 | 818.60 | 842.06 |
| lc105 | 828.94 | 828.94 | 828.94 | 828.47* |
| lc106 | 828.94 | 828.94 | 828.94 | 828.94 |
| lc107 | 828.94 | 828.94 | 828.94 | 828.94 |

*lc105 probleminin rotaları EK-1'de verilmiştir.

Sonuçlara bakıldığında genel olarak yeni kodlama biçimiyle GA'nın iyi sonuçlar verdiği ve iyi bir alternatif çözüm yöntemi olabileceği söylenebilir. Problemlerin dördünde bilinen en iyi sonuca ulaşılmış, ikisinde yaklaşılmış ve birinde (lc105) daha iyi sonuç üretilmiştir.

6. Sonuçlar

Çalışmada, ZP-ÇDTRP problemi için Gerçek Değerli GA yaklaşımı önerilmiştir. Sonuçlar incelendiğinde önerilen kodlama biçimi ile GA'nın iyi bir alternatif olabileceği söylenebilir.

Önerilen model ile gerçek değerli kodlamanın genetik algoritmanın yapısını kolaylaştırdığı ve farklı operatörlerin daha hızlı şekilde uygulanmasına imkan verdiği görülmektedir. Algoritmanın daha hızlı çalışabilmesi için mümkün çözüm aralığına çok daha hızlı ulaşabileceği başlangıç çözümlerinin diğer klasik yöntemlerden elde edilebilir. Test edilen veri setindeki çeşitli zorluktaki diğer problemlere de genetik algoritma yaklaşımı uygulanabilir. Çalışmada uygulanan genetik algoritma sonuçlarının geliştirilmesi için iyileştirmeler yapılabilir. Yine farklı tekniklerle melez bir model problemin çözümü için kullanılabilir. Ayrıca paralel arama algoritmalarının çok daha etkin kullanılması ile CPU işlem zamanının düşürülmesi hedeflenebilir.

Kaynakça

- [1] D. Simchi-Levi, X. Chen, J. Bramel, *The Logic of Logistics: Theory, Algorithms and Applications for Logistics Management*, Springer, 2005.
- [2] O. Bräysy, M. Gendreau, Vehicle routing problem with time windows, Part II: Metaheuristics. *Transportation Science*, 39, 1, 119-139 (2005).
- [3] T.G. Crainic, G. Laporte, *Fleet Management and Logistics*, Springer, 1998.
- [4] J.F. Cordeau, G. Laporte, M.W.P. Savelsbergh, D. Vigo, Vehicle Routing. *Transportation, Handbooks in Operations Research and Management Science*, 14, 367-428 (2007).

- [5] M.M. Solomon, J. Desrosiers, Survey Paper-Time Window Constrained Routing and Scheduling Problems. *Transportation Science*, 22, 1-13 (1988).
- [6] M.M. Solomon, Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations research*, 35, 2, 254-265 (1987).
- [7] B. Funke, T. Grünert, S. Irnich, Local search for vehicle routing and scheduling problems: Review and conceptual integration. *Journal of Heuristics*, 11, 4, 267-306 (2005).
- [8] L.D. Bodin, Twenty years of routing and scheduling. *Operations Research*, 38, 4, 571-579 (1990).
- [9] G.B. Dantzig, J.H. Ramser, The Truck Dispatching Problem. *Management Science*, 6, 1, 80-91 (1959).
- [10] G. Laporte, Fifty Years of Vehicle Routing. *Transportation Science*, 43, 4, 408-16 (2009).
- [11] B. Eksioğlu, A.V. Vural, A. Reisman, The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering*, 57, 1472-1483 (2009).
- [12] B.L. Golden, A.A. Assad, E.A. Wasil, Routing vehicles in the real world: applications in the solid waste, beverage, food, dairy, and newspaper industries. *The vehicle routing problem*, 9, 245-286 (2002).
- [13] Partyka, Janice G., and Randolph W. Hall, On the Road to Service. *OR/MS Today* 27, 4, 26-35 (2000).
- [14] R.H. Ballou, *Business Logistics and Supply Chain Management*, Pearson Prentice Hall, 2004.
- [15] P. Brandimarte, G. Zotteri, *Introduction to Distribution Logistics*, Wiley-Interscience. 2007.
- [16] P. Toth, D. Vigo, *The Vehicle Routing Problem*, SIAM, 2002.
- [17] G. Desaulniers, J. Desrosiers, M.M. Solomon, *Column Generation*, Springer, 2005.
- [18] G. Berbeglia, J.-F. Cordeau, I. Gribkovskaia, G. Laporte, Static pickup and delivery problems: A classification schema and survey. *Top*, 15, 1-31 (2007).
- [19] M.W.P Savelsbergh, M. Sol, The General Pickup and Delivery Problem. *Transportation Science*, 29, 1, 17-29 (1995).
- [20] M. I. Hosny, C. L. Mumford, Investigating Genetic Algorithms for Solving the Multiple Vehicle Pickup and Delivery Problem with Time Windows. *MIC2009, Metaheuristic International Conference*, Hamburg, Germany. 2009.
- [21] M. I. Hosny, C. L. Mumford, The single vehicle pickup and delivery problem with time windows: intelligent operators for heuristic and metaheuristic algorithms. *Journal Heuristics*, 16, 417-39 (2010).

- [22] S. Ropke, J.F. Cordeau, G. Laporte, Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *NETWORKS*, 258–72 (2007).
- [23] S. Ropke, J.-F. Cordeau, Branch-and-cut-and price for the pickup and delivery problem with time windows. *Transportation Science*, 43, 3, 267–86 (2009).
- [24] D.E. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*, Addison Wesley Publishing Company, USA, 1989.
- [25] Z. Michalewicz, *Genetic Algorithms + Data Structure = Evolution Programs*, Springer-Verlag, Berlin, 1992.
- [26] C.R. Reeves, *Modern Heuristic Techniques for Combinatorial Problems*, McGraw-Hill Book Company Inc., Europe, 1995.
- [27] R. M. Jorgensen, J. Larsen, K. B. Bergvinsdottir, Solving the dial-a-ride problem using genetic algorithms. *Journal of the Operational Research*, 58, 10, 1321–31 (2007).
- [28] G. Pankratz, A grouping genetic algorithm for the pickup and delivery problem with time windows. *OR Spectrum*, 27, 21–41 (2005).
- [29] J.-C. Créput, A. Koukam, J. Kozlak, J. Lukasik, An Evolutionary Approach to Pickup and Delivery Problem with Time Windows. *In Computational Science-ICCS 2004*, 1102–8 (2004).
- [30] Y. Nagata, S. Kobayashi, A Memetic Algorithm for the Pickup and Delivery Problem with Time Windows Using Selective Route Exchange Crossover, *In Parallel Problem Solving from Nature PPSN XI*, 6238, 536–545 (2010).
- [31] D. E. Goldberg, K. Deb, A Comparative Analysis of Selection Schemes Used in Genetic Algorithms, *Urbana*, 51, 61801–996 (1991).
- [32] H. Li, A. Lim, A Metaheuristic for the Pickup and Delivery Problem with Time Windows, *In Tools with Artificial Intelligence, Proceedings of the 13th International Conference on*, 160–67 (2001).

EK-1 Ic105 Probleminin Rotaları

| | | | | | | | | | | | | | |
|----|----|-----|----|----|----|----|-----|-----|-----|----|-----|----|----|
| 20 | 24 | 32 | 33 | 31 | 35 | 37 | 38 | 39 | 36 | 34 | 101 | | |
| 43 | 42 | 41 | 40 | 44 | 46 | 45 | 48 | 51 | 50 | 52 | 104 | 49 | 47 |
| 98 | 96 | 95 | 94 | 92 | 93 | 97 | 100 | 99 | 105 | | | | |
| 67 | 65 | 103 | 63 | 62 | 74 | 72 | 61 | 64 | 68 | 66 | 69 | | |
| 57 | 55 | 54 | 53 | 56 | 58 | 60 | 59 | | | | | | |
| 90 | 87 | 86 | 83 | 82 | 84 | 85 | 88 | 89 | 91 | | | | |
| 25 | 27 | 29 | 30 | 28 | 26 | 23 | 106 | 22 | 21 | | | | |
| 13 | 17 | 18 | 19 | 15 | 16 | 14 | 12 | | | | | | |
| 81 | 78 | 76 | 71 | 70 | 73 | 77 | 79 | 102 | 80 | | | | |
| 5 | 3 | 7 | 8 | 10 | 11 | 9 | 6 | 4 | 2 | 1 | 75 | | |