Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi 10 / 1 (2008). 97 - 116

SABİT MALİYETLİ ULAŞTIRMA PROBLEMİ İÇİN BİR GENETİK ALGORİTMA

Tuba YAKICI AYAN*

Öz:

Bu makalede klasik ulaştırma probleminin (UP) bir uzantısı olan sabit maliyetli ulaştırma problemini (SMUP) çözmek için bir genetik algoritma (GA) sunulmaktadır. SMUP nin temel özelliği, taşınan miktara orantılı bir değişken maliyetin yanı sıra taşınan miktardan bağımsız bir sabit maliyetin de söz konusu olmasıdır. Matematiksel formülasyonu bir tamsayılı programlama problemi olmakla birlikte amaç fonksiyonundaki kesikli yapı nedeni ile problemin bilinen optimizasyon algoritmaları ile çözülmesi mümkün değildir. Doğal seleksiyona dayanan bir evrimsel süreç olarak GA, SMUP nin çözümünde son derece etkin bir araçtır. Bu çalışmada önerilen GA için başlangıç kitlesinin oluşturulması, çaprazlama ve tamir süreçlerine dair algoritma parçaları açıklandıktan sonra DELPHI de kodlanmakta ve bir örnek problem çözülerek işlerliği ve etkinliği ortaya konulmaktadır.

Anahtar Kelimeler: Genetik algoritma, Sabit maliyetli ulaştırma problemi, Tamsayılı programlama, Çaprazlama.

A GENETIC ALGORITHM FOR FIXED CHARGE TRANSPORTATION PROBLEM

Abstract:

In this paper, a genetic algorithm (GA) for fixed charge transportation problem (FCTP) which is an extension of classical transportation problem (TP) is presented. In FCTP, a fixed cost is incurred, independent of the amount transported, along with a variable cost that is proportional to the amount shipped .The mathematical formulation of the problem indicates that the model is an integer linear programming problem but that cost structure results in the objective function being a step function. Therefore FCTP can not been solved by known optimization algorithms. GA is an efficient procedure that finds solution to FCTP by an evolutionary process based on natural selection. In order to solve the problem, a GA with initial population, crossover and repair sub

* Yrd.Doç.Dr., Karadeniz Teknik Üniversitesi, İktisadi İdari Bilimler Fakültesi, Ekonometri Bölümü, ayan@ktu.edu.tr

algorithm is presented. The proposed algorithm is coded with DELPHI language and illustrated with a numerical example. Numerical experimental results show that the proposed algorithm is efficient for solving FCTP.

Keywords: Genetic algorithm, Fixed charge transportation problem, Integer linear programming, Crossover.

GİRİŞ

Sabit maliyetli ulaştırma problemi (SMUP), değişken giderlerin yanı sıra sabit giderlerin de söz konusu olduğu bir kısıtlı optimizasyon problemidir. Problemin tesis yerleştirme ve ulaştırma literatüründe ele alınmış çeşitli uygulama alanlarından en yaygın ikisi şunlardır: (1) Tesis açma giderinin mevcut olduğu fabrika ya da depo açma kararlarının verilmesi ve (2) arz- talep noktaları arasında mal taşımada sabit giderlerin de mevcut olduğu ulaştırma problemi. Bir bakış açısına göre SMUP, Hitchcock (1941) tarafından geliştirilmiş ve en tanınmış kısıtlı optimizasyon problemlerinden biri olan klasik ulaştırma probleminin (UP) bir uzantısı olarak ifade edilebilir. Klasik ulaştırma problemi, fabrikalar ya da depolar kümesinden müşteriler kümesine bir tek malı taşıma durumudur. Burada amaç, taşınan mal miktarına göre doğrusal olan taşıma maliyetleri ile toplam ulaştırma maliyetini en küçük yapmaktır. Açıkça görülmektedir ki, UP bütün sabit maliyetlerin sıfır olduğu bir SMUPdir. Ancak UP literatürde çok fazla incelenmiş iken SMUP pek fazla ele alınmamıştır. Oysa uygulamada çoğu zaman bir arz merkezi ile bir müşteri arasında mal taşımak için bir rota oluşturulduğunda sabit maliyetler ortaya çıkar. Bunun sonucu olarak ulaştırma maliyeti, taşınan mal miktarından bağımsız sabit maliyetlerden ve taşınan mal miktarına orantılı değişken maliyetlerden oluşmaktadır. Sabit maliyetler iki şekilde ortaya çıkabilirler. Bir ya da daha fazla müşterinin taleplerini karşılamak için bir fabrika ya da depo kurmak sabit maliyetlere yol açmaktadır. Ayrıca taşımacılık şirketlerinin neredeyse hepsi, taşınan mal miktarına orantılı değişken maliyetlere ilaveten bedeli bölgeden bölgeye, uzaklığa veya taşınan malın cinsine göre değişen nakliye ücretleri (navlun) kullanırlar. Hirsch ve Dantzing (1968), de de ifade edilmiş olduğu gibi SMUP nin, üretim-dağıtım sektöründe UP den çok daha gerçekçi bir problem olduğuna kuşku yoktur. UP, doğrusal programlama, tamsayılı programlama veya dal-sınır metodu gibi yöntemlerle uygun sürede çözülebilirken SMUP için durum farklıdır. Probleme sabit maliyetlerin dahil edilmesi *mn* boyutlu bir problemde değişken sayısını 2 katına çıkartmakta ve *2mn* adet ilave kısıta neden olmaktadır. Ayrıca küçük boyutlu problemlerin bile yukarıda sözü geçen kesin çözüm algoritmaları ile çözülebilmeleri çok fazla hesaplama ve zaman gerektirmektedir (Steinberg, 1970). Bu nedenle, büyük boyutlu gerçek yaşam problemleri için uygun sürede optimale yakın çözümler üretecek sezgisel metotlara ihtiyaç duyulmaktadır. En fazla önerilen sezgisel metotlar UP' ye dayanmakla birlikte, literatürde sezgisel veya kesin sonuç veren çeşitli çözüm yöntemleri önerilmektedir.

Sezgisel çözüm yöntemleri içinde özel bir yere sahip olan genetik algoritmalar (GA), Darwin' in evrim ilkelerinden yola çıkarak optimizasyon problemlerine doğal seleksiyon ve genetik mekanizmalara dayalı bir evrimsel süreç aracılığı ile iyi çözümler bulmaya çalışmaktadır. GA nın dayandığı temel bileşenler, problem çözümlerinin genetik bir gösterimi, başlangıç çözümlerinden oluşan kitlenin oluşturulma tarzı, herhangi bir çözümün kalitesini ölçmek için bir değerlendirme fonksiyonu, yeni kitle üretme sırasında aday çözümlerin kombinasyonunu etkileyen genetik operatörler ve evrimleşme için kullanılacak parametre değerleri olarak özetlenebilir. Kısıtsız optimizasyon problemleri GA ile etkin bir şekilde çözülebilirken kısıtlı problemlerde durum farklıdır. Bu tür problemlerin GA ile çözülmelerinde kullanılan başlıca yaklaşımlar, ceza metodu, tamir algoritmaları veya mümkün olması halinde kısıtları otomatikman sağlayan çözümler üretecek genetik operatörlerin kullanılmasıdır.

Bu makalede yüksek derecede kısıtlı olan sabit maliyetli ulaştırma problemi ele alınmakta ve en iyi sezgisel çözümü bulmak için bir genetik algoritma önerilerek oldukça ayrıntılı bir şekilde açıklanmaya çalışılmaktadır. Çalışmanın ana vurgusu kısıtlı problemlerde GA nın kullanımı olduğu için probleme özgü matris kodlama yerine genel vektör kodlama tercih edilmiştir. Çaprazlama ve tamir operatörleri ise tamamen ilgili probleme özgü olmak zorundadır. Çünkü her problem kendi özel kısıtlarına sahiptir ve bu nedenle kısıtları otomatikman tatmin eden operatörlerin genelleştirilmesine olanak bulunmamaktadır. Ancak söz konusu operatörler UP nin amaç fonksiyonu doğrusal olmayan, değişken olan veya birden fazla amaç fonksiyonuna sahip olan farklı versiyonları için değiştirilmeden ya da çok az değişiklikle kullanılabilecektir. Bu makalenin kalan kısmı şöyle organize edilmiştir: Bölüm I de SMUP için bir literatür araştırması sunulmaktadır. Bölüm II de problemin matematiksel formülasyonu verilmektedir. Bölüm III, GA nın temel kavramları ve genel adımlarına ayrılmıştır. Bölüm IV de problemi çözmek için kullanılan yaklaşım ayrıntılı olarak anlatılmaktadır. Bölüm V de önerilen yöntemin işlerliğini göstermek için sayısal bir örnek verilmektedir. Son bölüm, bazı sonuçları ve gelecek araştırmalar için bazı yönlendirmeleri içermektedir.

I) LİTERATÜR ARAŞTIRMASI

SMUP nin çözümü için önerilen kesin algoritmaların çoğu, performansı esasen kullanılan sınırlandırma stratejisine bağlı olan dal-sınır metotlarına dayanmaktadır. Balinski (1961), y_{ij} üzerindeki tamsayı kısıtını ihmal ederek indirgenmiş bir SMUP versiyonunun optimal çözümüne dayanan bir algoritma sunmuştur. Hirsch ve Danzig (1968), SMUP nin optimal çözümünün kısıt setinin herhangi bir uç noktasında ortaya çıkacağını ve uygun çözüm bölgesinin her uç noktasının bir yerel minimum olduğunu ortaya koymuştur. Gray (1971), problemi bir ana tamsayılı programlama ve bir ulaştırma alt programları serisine bölerek kesin çözüm sağlamaya çalışmıştır. Ayrıca Steinberg (1970), Kennington vd. (1976), Barr vd. (1981), Palekar (1986), Palekar vd.

(1990) dal-sınır metoduna dayanan kesin algoritmalar elde etmeye çalışmışlardır. Problemin çözümü için en bilinen kesin metotlardan biri, yükün geniş bir dağıtım alanını analiz etmeyi gerektiren “uç noktaları derecelendirme” metodudur. Murty (1968), başlangıç çözümü olarak problemin değişken kısmının optimal olarak çözülmesini ve sonra sabit maliyetleri her bir noktaya ilave ederek dağıtım alanını analiz etmeyi önermiştir. Burada analiz edilecek alanın büyüklüğü başlangıç çözümüne bağlıdır ve yazarın kendisi de önerilen yöntemin büyük sabit maliyetlerin bulunduğu problemler için zayıf bir başlangıç noktası sağladığını ifade etmektedir. Sadogopan ve Ravindran (1982), Murty’nin yöntemine katkı yapmışlardır. Onların önerdikleri hareket tarzı, başlangıç olarak problemin sabit maliyetli kısmını çözmek ve Murty’nin önerisinden daha iyi bir başlangıç çözümü elde edilmesi halinde çözüme oradan devam etmek şeklinde özetlenebilir. Ancak bu yöntem de büyük bir dağıtım alanını analiz etmeyi gerektirmektedir. Kowalski (2005), *mn* boyutlu bir SMUP nin optimal çözümündeki maksimum değişken sayısının m+n-1 olduğunu kanıtlamıştır. Bu çalışma problem için yeni çözüm yöntemlerinin geliştirilmesinde önemli katkı sağlamıştır. Cooper ve Drebes (1967), Drenzler (1969), Stinberg (1970), Cooper (1975), Walker (1976), Sun vd. (1998) ve Adlakha, ve Kowalski (2003) simpleks benzeri iterasyonlarla sonuca ulaşmaya çalışan sezgisel yaklaşımlar geliştirmişlerdir. Ancak bu yaklaşımlardan Adlakha ve Kowalski (2003) dışındakiler başlangıç çözümüne fazla vurgu yapmamışlardır. Ayrıca yaklaşımların hiçbirinde elde edilen çözümün yerel değil global optimum olduğunun garantisi yoktur. Sandrock (1988), sabit maliyetin rotalara değil arz noktalarına ait olduğu küçük problemler için basit bir algoritma sunmuştur. Chien (1993), taşıma maliyetlerinin kamyon başına sabit maliyet ile modellendiği bir problem için sezgisel bir yöntem önermiştir. Adlakha vd. (2007) de hem UP ve hem de SMUP için bir MFL (more for less paradoksu) çözümü bulmak için analitik sezgisel bir yöntem önerilmiştir. Bu yöntemde etkin maliyet yapısından dolayı her çözümde yer alabilecek hücreler aranmakta ve bu hücreler tam yüklenerek problemin boyutu küçültülmektedir. Ancak böyle hücreler her zaman mevcut olmayabilir. Ayrıca yöntem alternatif fırsatları dikkate almama şeklinde bir dezavantaja sahiptir. SMUP literatüründe genetik algoritmalar dışında dikkate değer diğer birkaç çalışma Kuhn ve Baumol (1962), Diaby (1991), Abdel-Wahed ve Lee (2006), Kuo ve Nichells (2007), Dorndorf vd. (2007) ve Kowalski ve Lev (2007) olarak özetlenebilir.

GA ilk olarak Holland (1975) tarafından ortaya atılmış ve daha sonra Grefenstette (1986), Goldberg (1989), Deb (1995), Michalewicz (1992, 1996), Sakawa (2002) ve diğer bazı araştırmacılar tarafından geliştirilmiştir. Pek çok zor kısıtlı optimizasyon problemini çözmek için kullanılmış olan GA çalışmaları arasında, son yıllarda, Gen ve Cheng (1997,2000),Gen ve Syarif (2003), Syarif ve Gen (2003a) dikkate değer nitelikte bulunmaktadır. Daha önce de belirtilmiş olduğu gibi SMUP nin özel bir durumu olan klasik UP ve çeşitli versiyonları için GA çalışmalarına literatürde sıkça rastlanmaktadır. Vignaux ve Michalewicz (1989),Michalewicz vd. (1990), Michalewicz vd. (1991)ve Mizunuma ve Watada (1995) ulaştırma problemlerinin en yalın hali olan doğrusal UP için GA yı kullanmışlardır. Vignaux ve Michalewicz (1991) de doğrusal UP için GA da bazı gösterim planları analiz edilmekte, çözüm adaylarını kodlamada bit katarlarının değil matris veya permütasyon kodlamanın uygun olduğu belirtilmekte ve yavru çözümlerin uygunluğunu garanti eden çaprazlama ve mutasyon operatörleri önerilmektedir. Michalewicz (1993), özel bir doğrusal olmayan UP için beş farklı çözüm değerlendirme yaklaşımını karşılaştırmaktadır.

Literatürde SMUP için, UP ve diğer versiyonlarına kıyasla çok daha az olmakla birlikte birkaç GA çalışması mevcuttur. Bunlardan biri olan Gottlieb ve Paulmann (1998) de düşük hacimli kitlelerle iyi çözümler üretebildiği iddiası ile GA kullanılmıştır. Sözkonusu çalışmanın sonuç bölümünde de ifade edilmiş olduğu gibi, önerilen algoritmanın eksik yönü, sadece temel çözümleri değil bazı temel olmayan çözümleri de araştırması nedeni ile çözüm süresinin ve işlem hacminin çok fazla artmasıdır. Çalışmada ayrıca permütasyon ve matris gösterimlerin bir karşılaştırması da yapılmıştır. Li v.d. (1998) ve Ida v.d. (2004), SMUP için diğer dikkate değer GA çalışmaları olarak gösterilebilir.

II) SABİT MALİYETLİ ULAŞTIRMA PROBLEMİ

Sabit maliyetli ulaştırma problemi (SMUP), homojen bir malın m adet arz noktasından n adet talep noktasına dağıtımı problemi olarak ifade edilebilir. Arz noktası i den talep noktası j ye bir birim mal göndermenin maliyeti birim taşıma maliyeti c_{ij} ile i-j rotasını açmaktan kaynaklanan bir sabit maliyet f_{ij} nin toplamına eşittir. Her bir arz noktası (i = 1, 2,….,m) a_i birim kapasiteye ve her bir talep noktası (j = 1, 2,….,n) t_j birimlik bir talebe sahiptir. Burada amaç, belirli arz ve talep kısıtlarına uygun olarak bütün taleplerin mümkün olduğunca karşılanması için en küçük toplam maliyet oluşturmak üzere açılması gereken rotaları ve bu rotalar üzerinden gönderilecek mal miktarlarını belirlemektir.

Standart SMUP formülasyonu:

$$Z_{min} = \sum_{i=1}^{m}\sum_{j=1}^{n}(c_{ij}x_{ij} + f_{ij}y_{ij}) \tag{1}$$

$$\sum_{j=1}^{n} x_{ij} = a_i \quad i = 1,2,...m \tag{2}$$

$$\sum_{i=1}^{m} x_{ij} = t_j \quad j = 1,2,....n \tag{3}$$

$$y_{ij} = \begin{cases} 0 & x_{ij} = 0 \text{ ise} \\ 1 & x_{ij} \rangle \ 0 \text{ ise} \end{cases} \tag{4}$$

$$x_{ij} \geq 0 \quad i = 1,2,...m \text{ ve } j = 1,2,...n \tag{5}$$

Yukarıdaki standart SMUP formülasyonunda kısıt (2) ve (3) den toplam arzın toplam talebe eşit olduğu ortaya çıkmaktadır. Bu eşitlik dengeli problem durumunu ifade etmekle birlikte, bu çalışmada sunulan bütün algoritmalar hem dengeli hem de dengesiz problemleri çözebilir niteliktedir. Sunulan algoritmalardaki bir diğer farklılık ise, standart ulaştırma modelindeki gibi, gönderilecek mal miktarlarının tamsayı olması zorunluluğudur. Bu zorunluluk ilave bir kısıt olarak probleme dahil edildiğinde ortaya bir tamsayılı programlama problemi çıkmaktadır. Hirsch ve Danzig (1968), SMUP nin her ekstrem noktası bir yerel minimum olan uygun çözüm alanının, konkav bir amaç fonksiyonu ile sınırlanmış bir konveks set olduğunu, ve optimal çözümün kısıt setinin ekstrem bir noktasında bulunabileceğini ortaya koymuşlardır. SMUP nin sadece değişken maliyetleri kapsayan yani UP ye indirgenmiş halinin optimal çözümü temel çözüm olarak adlandırılmakta. ve en fazla m+n-1 adet yükleme içerebileceği bilinmektedir. Bu nedenle problemin araştırma uzayının m+n-1 den fazla yükleme içermeyen aday çözümlerle sınırlandırılması, zaman ve işlem miktarı açısından büyük avantaj sağlayacaktır.

III) GA NIN TEMEL KAVRAMLARI VE GENEL ADIMLARI

Kromozom: Aday çözüm. Her bir kromozom, probleme ait özellikleri temsil eden bir diğer deyişle değişken değerlerini gösteren birçok genden oluşmaktadır.

Kromozomların gösterimi: Kromozomlar, 0–1 sayı katarı veya harf katarı (permütasyon kodlama) olarak kodlanabileceği gibi reel kodlama da kullanılabilir. Gösterim türünün seçimi çoğu zaman, problemin yapısına ve kullanılacak algoritmanın ayrıntılarına bağlı olmakla birlikte her birinin bazı üstünlük ve sakıncaları mevcuttur. 0–1 kodlama, çaprazlama ve mutasyon işlemlerine elverişliliği açısından en yaygın kullanılan kodlama türüdür. Ancak çok uzun kromozomlar oluşturması nedeni ile aritmetik işlemler açısından, güçlük yaratabilmekte, bilgisayar programında çok fazla hafıza gerektirebilmekte ve doğrudan görsel olarak yorumlanmaları zor olabilmektedir. Harf kodlama çok nadiren kullanılmakla birlikte. gezgin satıcı problemi, çeşitli araç rotalama problemleri, sıralama vb problemler için uygun bir kodlama türü olabilmektedir. Problem değişkenlerinin alabilecekleri gerçek değerlerle gösterilmeleri şeklinde açıklanabilecek olan reel kodlama ise kısıtlı optimizasyon problemlerinde çok tercih edilen bir kodlama türüdür. Çünkü çaprazlama ve mutasyon işlemleri kromozomların kısıtlara uygunluklarını çok kolayca bozabilmektedir. Bu tehlikeyi gözlemleyip kontrol altında tutabilmek ancak reel kodlama ile mümkündür.

Kitle hacmi: Arama uzayındaki toplam kromozom sayısıdır ve araştırma boyunca sabit kalır. Kitle hacmi genellikle keyfi olarak belirlenmekle birlikte araştırmalarda yaygın olarak 20–30 arası büyüklük kullanılmaktadır. Ayrıca kromozom uzunluğuna dayanan bir hesaplama yöntemi (Goldberg,1989) veya problemdeki değişken sayısına bağlı bir yöntem de mevcuttur.

Üreme: İki ebeveyn kromozomdan yavru kromozomların oluşturulmasıdır.

Değerlendirme fonksiyonu: Her bir aday çözümün amaca yakınlık düzeyini diğer bir deyişle çözüme katkısını ölçmeye yarayan ve tamamen probleme bağlı olarak belirlenen bir fonksiyondur. Kısıtsız problemlerde değerlendirme fonksiyonu olarak amaç fonksiyonu alınmaktadır. Kısıtlı problemlerde ise tercih edilen GA türüne bağlı olarak yalnız amaç fonksiyonu alınmakta ya da kısıt ihlal miktarları da amaç fonksiyonuna eklenmektedir.

Seçim: Üreme yapmak üzere rasgele olarak ebeveyn kromozomların seçilmesi işlemidir. Burada temel amaç, ortalamadan daha iyi olan çözümler üzerinde yoğunlaşmak ve "güçlü olan hayatta kalır" evrimsel ilkesine göre kötü çözümleri gelecek nesillerden dışlamaktır. Seçim işleminde en fazla kullanılan yöntemler rulet tekerleği seçimi, turnuva seçimi ve sıralama esaslı seçimdir. Yöntemlerin hepsi, kromozomların değerlendirme fonksiyonunda almış oldukları değerleri yani çözüme katkılarını dikkate almaktadır. Rulet tekerleği yönteminde, tekerlek üzerinde her bireye (kromozoma) optimal çözüme katkı yüzdesi kadar yer ayrılarak tekerlek çevrilmekte ve okun durduğu noktadaki kromozom seçilmektedir. Bu yöntemin üstünlüğü, en iyi bireylerin yeni kitleye girme şanslarını arttırması iken sakıncası ise çözüm sürecinin yerel bir optimuma takılması tehlikesidir. Turnuva seçiminde her seferinde rasgele iki kromozom karşılaştırılarak iyi olan seçilmekte ve seçilmiş bireylerin tekrar seçime girebilmeleri ile N adet birey arasından N adet bireyin seçimi sağlanmaktadır. Burada da söz konusu olan yerel optimuma takılabilme tehlikesini ortadan kaldırabilmek için uygulanan bir yöntem sıralama esaslı seçimdir. Bölüm IV de ayrıntılı olarak açıklanan yöntemde, bireylerin seçiminde doğrudan değerlendirme puanları değil, puanlamadaki sıra numaraları esas alınmaktadır. Ancak bu yaklaşımda da iyi bireyler kaçırılarak, global optimuma ulaşma süreci uzayabilirse de bütün bu sakıncalar, uygun kitle büyüklüğü ile çok önemsiz düzeye indirilebilmektedirler.

Epistatik gen: Değişmeleri halinde diğer genleri etkileyen genlerdir. Kısıtlı optimizasyon problemlerinin GA ile çözülmelerindeki zorluğun nedeni, böyle problemlerdeki kromozomların büyük ölçüde epistatik genlerden oluşmaları olarak ifade edilebilir.

Uygun çözüm: Kısıtlı optimizasyon problemlerinde, bütün kısıtları sağlayan çözüm.

Çaprazlama yüzdesi: Kitlede çaprazlama yapacak olan kromozom oranı olup genellikle 0.75-0.90 arasında alınmaktadır. Çaprazlama yüzdesi 0 alınırsa yeni kitle tamamen eskisinin aynısı olacaktır. 1 alınması halinde ise, kitledeki bütün bireyler değişecektir. Çaprazlama yüzdesinin yüksek olması eski kitledeki iyi bireylerin kaybolmasına neden olabilir. Bunu önlemek için çaprazlamanın ardından oluşan en kötü yavru ile eski kitledeki en iyi birey değiş tokuş edilebilir. Bu işleme elitist strateji denilmektedir.

Çaprazlama: Ebeveyn kromozomlardaki genlerin çeşitli kombinasyonları ya da aritmetik işlemlere girmeleri sonucu yavru oluşturulması aşamasıdır. Tek noktalı, iki noktalı, üniform (Spears and DeJong,1991), yarı üniform ve kes-ekle çaprazlama yöntemleri mevcuttur. Ayrıca probleme özgü aritmetik çaprazlama operatörleri de kullanılabilmektedir. Çaprazlama yüzdesine dayanarak rasgele seçilen iki kromozomun benimsenen yöntem ile çaprazlanması sonucu iki yavru oluşmakta ve kitle hacmine eşit sayıda yavru elde edilinceye kadar bu işlem tekrarlanmaktadır.

Mutasyon yüzdesi: Bir kitlede mutasyona uğrayacak gen oranını göstermektedir. Mutasyon işlemi kromozomdaki bilgiyi kolayca bozabileceği için mutasyon olasılığı çaprazlama olasılığından çok daha düşük (genellikle 0,001 gibi) alınmaktadır.

Mutasyon: Bir kromozomdaki bir veya daha fazla bileşenin değişmesi işlemidir. Araştırma ilerledikçe bir noktadan sonra kromozomların birbirlerini tekrarlamaları tehlikesi mevcuttur. Mutasyon, nesildeki kromozom çeşitliliğini artırarak bu tehlikeden kaçınmaya yardımcı olmaktadır.

Elitist geçiş: Goldberg (1989) tarafından geliştirilmiş olan strateji, iyi çözümlerin gözden kaçırılmaması amacı ile en iyi bireyin bir sonraki nesile kopyalanması işlemidir. Ancak burada da mevcut olan yerel minimuma takılma tehlikesi, uygun kitle büyüklüğü ile aşılabilmektedir.

Durdurma ölçütü: Genetik araştırmaya ne zaman son verileceğini göstermektedir. En yaygın kullanılan durdurma ölçütleri, optimal çözüm, önceden belirlenmiş bir nesil sayısı, zaman ya da bütçe kısıtlaması ve ardı ardına aynı çözümün birçok kez tekrarlanması olarak özetlenebilir.

GA nın genel adımları:

Adım 1. Başlatma: Rastgele olarak N adet kromozomdan oluşan bir başlangıç kitlesi oluşturulur.

Adım 2. Değerlendirme: Başlangıç kitlesindeki kromozomlar değerlendirme fonksiyonuna göre derecelendirilir.

Adım 3. Seçim: İyi olan N adet kromozom, seçim yöntemlerinden biri ile yavrular oluşturmak üzere seçilir. Burada dikkate değer bir nokta, herhangi bir kromozomun tekrar seçilebilmesidir.

Adım 4. Çaprazlama: Çaprazlama yüzdesi ile kitle hacminin çarpılması ile bulunan r_1 değerine dayanarak rasgele olarak iki kromozom seçilir. Uygun görülen bir çaprazlama yöntemi ile iki yavru kromozom oluşturulur. N adet yavru oluşturulmuş oluncaya kadar bu adım tekrarlanır. Bu adımda da bir kromozom tekrar çaprazlama yapmak üzere seçilebilir.

Adım 5. Elitist strateji: Bu adım tercihe bağlıdır. Eğer uygulanmasına karar verilmiş ise adım 4 de oluşturulmuş olan kitledeki en kötü kromozom bir önceki kitledeki en iyi çözüm ile değiştirilir.

Adım 6. Mutasyon: Mutasyon yüzdesi ile kitle hacminin çarpılması sonucu bulunan r_2 değerine dayanarak rasgele olarak seçilen bir gene, uygun bir mutasyon yöntemi ile var olandan farklı bir değer atanır. r_2 adet gen için aynı işlem tekrarlanır. Çok küçük bir gerçekleşme olasılığına sahip olduğu için bu aşama pek çok çalışmada uygulanmayabilmektedir.

Adım 7. Tamir: Algoritmanın herhangi bir aşamasında problem kısıtlarına uygun olmama anlamında bozulan kromozomların varlığı durumunda uygulanan bir düzeltme işlemidir. Tanımından da açık olduğu üzere, ilgilenilen probleme ve kullanılan çözüm algoritmasına bağlı olarak tercihsel bir adımdır.

Adım 8: Adım 2 ye dönülür. Bir durdurma ölçütüne ulaşılıncaya kadar algoritma tekrarlanır.

Adım 9: **Sonuç:** Bulunan en iyi kromozom, çözüm olarak seçilir.

IV) KULLANILAN YAKLAŞIM

Bu bölümde SMUP nin çözümü için, bütün kısıtları sağlayarak uygun çözümler üreten bir genetik algoritmanın uygulama adımları her bir aşamaya özgü algoritma parçaları ile birlikte açıklanmaktadır.

A) Kromozom Gösterimi

Çeşitli ulaştırma problemlerinin genetik algoritmalarla çözülmelerine yönelik geçmiş çalışmalarda büyük çoğunlukla reel kodlama (doğal gösterim) tercih edilmiştir. Reel kodlamada daha çok benimsenen tarz matris gösterimi olmakla birlikte permütasyon kodlamada kullanılabilmektedir. Bu çalışmada bütün mantığı ve işleyişi matris işlemleri ile aynı olan reel sayı katarlı gösterim tercih edilmiştir. Bunun sebebi, matris gösteriminin uygun olmayabileceği problemler için çok faydalı olabilecek bazı ayrıntılar sunmaktır. Kromozom gösteriminde her bir gen, birbirine eşit ve en büyük arz veya talep miktarının basamak sayısına eşit sayıda karakterle temsil edilmektedir. Burada genler, her bir arz noktasından her bir talep noktasına gönderilecek mal miktarını ifade etmektedir İlk n adet gen birinci arz noktasına aittir. Ardından ikinci arz noktasına ait olan genler sıralanmakta ve bu şekilde devam edilmektedir. Bir kromozom genel formda aşağıdaki gibi ifade edilebilir. Basamak sayısı bs ile gösterilmek üzere;

bs= Basamak sayısı(max(a_i,t_j)) i= 1,……,m ve j= 1,….,n

Kromozom=(gen_1,gen_2,…..gen_{m*n})

Kromozomun her bir bileşenini temsil eden tamsayı (d_i) ile daha açık olarak

Kromozom=(d_1 d_2 ….d_{bs} d_{bs+1} d_{bs+2} ….d_{2bs},…….d_{bs*m*n}) olmaktadır.

B) Başlangıç Kitlesinin Oluşturulması

Aşağıda, tamamen uygun bireylerden oluşan bir başlangıç kitlesi yaratmak için etkin bir algoritma adımları ile birlikte sunulmaktadır. Burada uygunluk, her yüklemede henüz yüklenmemiş arz ve talep miktarları dikkate alınarak başarılmaktadır. Aşağıda görülen [ ] simgesi tamsayı bölmeyi ifade etmek üzere öncelikle arz noktası sayısını (m), talep noktası sayısını (n), arz miktarlarını (a_i, i=1,…,m) , talep miktarlarını(t_j, j=1,…,n) ve en büyük basamak sayısını (bs) belirlenir.

i = 1 j= 1 s= 1

A1. $z = \min(a_i, t_j)$

(0-z) arasında rasgele sayı (rs) çek

l=1

A2. $d_l = \left[\dfrac{rs}{10^{bs-l}}\right](\text{mod}10)$

$l = l + 1$

Eğer $l \leq bs$ *ise A2 ye git, aksi halde*

$gen_s = \sum_{k=1}^{bs} d_k 10^{bs-k}$

$a_i = a_i - gen_s$ ve $t_i = t_i - gen_s$

s = s + 1

*Eğer s>m*n ise DUR "başlangıç kitlesi oluşmuştur". Aksi halde eğer s (modn) = 1 ise i=i+1, j=1 ve A1 e git. Aksi halde j=j+1 ve A1 e git.*

C) Değerlendirme Fonksiyonu

Potansiyel çözümlerin ne ölçüde iyi olduklarını belirlemek amacı ile (1) deki maliyet fonksiyonu kullanılmaktadır.

$$DF = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij} gen_{n(i-1)+j} + f_{ij} y_{ij} \tag{6}$$

$$y_{ij} = \begin{cases} 0, & gen_{n(i-1)+j} = 0 \text{ ise} \\ 1, & gen_{n(i-1)+j} > 0 \text{ ise} \end{cases} \tag{7}$$

D) Seçim Süreci

İyi olan ebeveynlerin seçimi için sıralama esaslı seçim yaklaşımı kullanılmaktadır. Seçim sürecinin adımları aşağıda özetlenmektedir.

A1. Kromozomları değerlendirme puanlarına göre büyükten küçüğe sırala

A2. Her bir kromozom için bir seçilme olasılığı hesapla,

$$P(kr_s) = \frac{2(N+1-s)}{N(N+1)} \qquad s = 1,...,N \tag{8}$$

A3. Her bir kromozom için birikimli olasılıkları hesapla,

$$q(kr_s) = \sum_{j=1}^{s} P(Kr_S) \qquad s = 1,...,N \tag{9}$$

A4. (0,1) arasında rasgele sayı çek (rs)

A5. rs den büyük en küçük $q(kr_s)$ değerine sahip olan bireyi seç

A6. N sayıda birey seçilmiş oluncaya kadarA4 ve A5 i tekrarla.

E) Çaprazlama

Kitledeki bireylerden yeni yavrular üretme aşamasında çeşitli çaprazlama operatörleri mevcut olmakla birlikte, yavruların uygunluğunu doğrudan garanti edebilecek tek yol aşağıda ayrıntıları sunulan aritmetik çaprazlamadır. Diğer yöntemler, uygunluk için ayrıca uzun mutasyon süreçleri gerektirdiklerinden dolayı bu bölümde sadece, daha önce benzer çalışmalarda kullanılmış olanlardan esinlenerek oluşturulmuş olan aritmetik çaprazlama operatöründen söz edilmektedir. Burada önerilen yaklaşım esasen, çaprazlamanın hemen ardından bir tür tamir algoritması uygulaması gibi de görülebilir. Ancak en doğru tanım, çaprazlama ile oluşan arızalı tek yavrunun doğumun hemen ardından tamir edilerek sağlam iki yavruya dönüştürülmesi olacaktır. Söz konusu çaprazlama operatörü temel olarak, seçilen iki kromozomun aritmetik ortalamasına dayanmaktadır. Aritmetik ortalamalar sonucu oluşan kromozomlar arz ve talep kısıtlarını doğal olarak sağlamakta ancak bütün gen ortalamalarının tamsayı olmaları garanti edilememektedir. Tamsayı kısıtının sağlanması için aritmetik ortalamaların kesirli kısımlarının atılması bir çözüm olarak görülmekle birlikte bu yöntem, arz ve talep kısıtlarının atılan kesirli kısımların toplamı kadar ihlal edilmesine yol açmaktadır. Çaprazlama sürecinin bu aşamasında bir tür tamir algoritması olarak da adlandırılabilecek bir yöntem, her bir arz noktasına ait genlerden atılan kesirli kısımların toplamı kadar bir miktarın o arz merkezine ait genlere tamsayı olarak tekrar eklenmesidir. Aynı işlemin her bir talep merkezine de uygulanması durumunda, bütün kısıt ihlalleri ortadan kalkmakta ve farklı dağıtım kombinasyonları ile iki yavru kromozom oluşturulabilmektedir. Burada dikkat çekici bir nokta arz ve talep merkezlerine dağıtılacak miktarların toplamlarının tamsayı olması gerekliliğidir ki bu, doğal olarak gerçekleşen bir durumdur. Çaprazlama operatöründe üzerinde durulması gereken bir başka önemli nokta, atılmış olan kesirli kısımların uygun şekilde dağıtılmasının nasıl sağlanabileceğidir. Bu aşamada genlerin epistatik olduklarını ve bu nedenle arz ve talep kısıtlarının birbirleri ile sıkı etkileşim halinde olduklarını hatırlatmak yerinde olacaktır. Genlerin bu özel yapıları nedeni ile başlı başına bir problem olarak karşımıza çıkan, eksik yüklemelerin arz ve talep koşullarına uygun olarak tamamlanması işlemi için çeşitli sezgisel (heuristic) yöntemler geliştirilebilir. Ancak bu çalışmada hızla seçenekli optimal çözümler sunan bir 0-1 programlama algoritması tercih edilmektedir. x_i, rasgele seçilmiş iki kromozomda aritmetik ortalamaları tamsayı olmayan genleri, b_i her bir arz merkezi için ve c_i ise her bir talep merkezi için ortalaması tamsayı olmayan genlerin sayısını göstermek üzere,

$$Z_{\text{max veya min}} = x_1 + x_2 + \ldots + x_n + x_{n+1} + x_{n+2} + \ldots + x_{2n} + x_{(m-1)n+1} + x_{(m-1)n+2} + \ldots + x_{mn} \qquad (10)$$

$$x_1 + x_2 + \ldots + x_n = b_1/2$$

$$x_{n+1} + x_{n+2} + \ldots + x_{2n} = b_2/2$$

$$x_{(m-1)n+1} + x_{(m-1)n+2} + \ldots + x_{mn} = b_m/2 \qquad (11)$$

$$x_1 + x_{n+1} + \ldots + x_{(m-1)n+1} = c_1/2$$

$$x_2 + x_{n+2} + \ldots + x_{(m-1)n+2} = c_2/2$$

$$\ldots$$

$$x_n + \ldots + x_{mn} = c_n/2 \qquad (12)$$

$$x_i = 0 \text{ veya } 1$$

Yukarıdaki problemin birbirinin tam tersi olan iki optimal çözümü, birbirinden farklı iki uygun kromozom oluşturmaktadır. Aşağıda çaprazlama sürecinde kullanılan algoritmanın adımları verilmektedir.

A1. r_1= çaprazlama yüzdesi*kitle hacmi ile yeni kitlede çaprazlama ile oluşacak kromozom sayısını hesapla

A2. Her bir kromozom çifti için 2/N(N–1) formülü ile birbirine eşit bir seçilme olasılığı hesapla

A3. (0,1) arasında bir rasgele sayı çek ve birikimli olasılıkları kullanarak bir kromozom çifti seç(kr_1, kr_2)

A4. Seçilen kromozomların aritmetik ortalamasını hesapla, tamsayı olanları 0, tamsayı olmayanları 1 olarak kaydet (kr'), kesirli kısımları at ve sonuçları başlangıç kitlesindeki basamak sayısına uygun şekilde kaydet (kr'').

gen_{1i} kr_1 de, gen_{2i} kr_2 de, gen_i' kr' de ve gen_i'' kr'' deki i no.lu geni göstermek üzere,

$$gen_i' = (gen_{1i} + gen_{2i}) \pmod 2$$

$$gen_i'' = \left[\frac{gen_{1i} + gen_{2i}}{2} \right]$$

Basamak sayısı standardizasyonu için,

$$d_l = \left[\frac{gen_i''}{10^{bs-l}} \right] (\mathrm{mod}10)$$

$l = l + 1$

Eğer $l \leq bs$ *ise A2 ye git, aksi halde*

$$gen_i'' = \sum_{k=1}^{bs} d_k 10^{bs-k}$$

A5. Değişken katsayılarını ve kısıtların sağ taraflarını kr$'$den alarak 0 – 1 programlama problemi (10-12) yi çöz.

$b_i = gen'_{(i-1)n+1} + gen'_{(i-1)n+2} + \ldots.. + gen'_{in}$

$c_i = gen'_i + gen'_{n+i} + \ldots.. + gen'_{(m-1)n+i}$

$Z_{max\ veya\ min} = gen'_1 x_1 + gen'_2 x_2 + \ldots.. + gen'_n x_n + gen'_{n+1} x_{n+1} + gen'_{n+2} x_{n+2} + gen'_{2n} x_{2n}$
$+ gen'_{(m-1)n+1} x_{(m-1)n+1} + gen'_{(m-1)n+2} x_{(m-1)n+2} + \ldots.. + gen'_{mn} x_{mn}$

$gen'_1 x_1 + gen'_2 x_2 + \ldots.. + gen'_n x_n = b_1/2$

$gen'_{n+1} x_{n+1} + gen'_{n+2} x_{n+2} + \ldots + gen'_{2n} x_{2n} = b_2/2$

$\ldots$

$gen'_{(m-1)n+1} x_{(m-1)n+1} + gen'_{(m-1)n+2} x_{(m-1)n+2} + \ldots.. + gen'_{mn} x_{mn} = b_m/2$

$gen'_1 x_1 + gen'_{n+1} x_{n+1} + \ldots. + gen'_{(m-1)n+1} x_{(m-1)n+1} = c_1/2$

$gen'_2 x_2 + gen'_{n+2} x_{n+2} + \ldots. + gen'_{(m-1)n+2} x_{(m-1)n+2} = c_2/2$

$\ldots.$

$gen'_n x_n + gen'_{2n} x_{2n} + \ldots.. + gen'_{mn} x_{mn} = c_n/2$

$x_i = 0$ *veya* 1

A6. Çözümde olan genlere 1 ekle, birinci uygun yavru kromozomu (kr_{y1}) oluştur.

Eğer $x_i = 1$ *ise* $gen_i^{y1} = gen_i'' + 1$ *aksi halde* $gen_i^{y1} = gen_i''$

$kr_{y1} = (gen_1^{y1} \ldots gen_n^{y1}\ gen_{n+1}^{y1} \ldots gen_{2n}^{y1}\ gen_{(m-1)n+1}^{y1} \ldots gen_{mn}^{y1})$

A7. Çözümde olmayan genlere 1 ekle, ikinci uygun yavru kromozomu (kr_{y2}) oluştur.

Eğer $x_i = 0$ *ise* $gen_i^{y2} = gen_i'' + 1$ *aksi halde* $gen_i^{y2} = gen_i''$

$kr_{y2} = (gen_1^{y2} \ldots gen_n^{y2}\ gen_{n+1}^{y2} \ldots gen_{2n}^{y2}\ gen_{(m-1)n+1}^{y2} \ldots gen_{mn}^{y2})$

A8. A3-A7 arasındaki adımları r/2 kere tekrarlayarak r adet yeni kromozom oluştur.

F) Elitist Strateji

Oluşan yeni kitledeki bütün kromozomların değerlendirme puanları hesaplanır. En düşük puana sahip olan kromozom kitleden çıkartılır ve bir önceki kitledeki en yüksek puanlı kromozom kitleye dahil edilir.

G) Tamir İşlemi

Bu aşamada, çözümlerin optimal adayı olabilmeleri için bir gereklilik koşulu olarak en fazla m+n–1 adet yüklemeli dağıtım planları oluşturulmaya çalışılmaktadır. Bu amaçla çaprazlama süreci ile elde edilmiş olan çözümler kontrol edilerek ve m+n–1 den daha fazla sıfırdan farklı gen içeren kromozomlar tamir edilmektedir. Geleneksel olarak bir tek gen üzerinden yapılan mutasyon burada mümkün değildir. Çünkü bir tek gendeki değişim, sıkı arz ve talep kısıtlamaları nedeni ile o genle aynı arz veya talep merkezlerine ait olan bütün genleri doğrudan etkileyecektir. Buna karşın aşağıda ayrıntıları verilmekte olan tamir işlemi dört gen üzerinden işleyen bir tür mutasyon gibi de görülebileceği için ayrıca mutasyon uygulamaya gerek yoktur. Bir kromozomun tamir sürecinin aşamaları:

A1. Kromozomdaki sıfırdan farklı genleri say (S). Eğer $S \leq m+n-1$ ise kromozom optimal çözüme adaydır. Aksi halde tamir işlemine başla

A2. Kromozomdan rasgele olarak sıfırdan farklı bir gen seç (gen_i).

A3. gen_i ile aynı arz merkezine ait olan genler arasından rasgele olarak sıfırdan farklı bir gen seç (gen_j).

gen_{i-1}, gen_{i-2},...,gen_{i-l} ,((i-l) (modn) = 1) ve gen_{i+1}, gen_{i+2},.. ,gen_{i+k}, ((i+k) (modn) = 0) arasından rasgele bir gen seç(gen_j)

Eğer $gen_j = 0$ ise A2 ye git. Yeni bir gen seçerek süreci yeniden başlat. Aksi halde

A4. gen_i ile aynı talep merkezine ait olan genler arasından rasgele olarak sıfırdan farklı bir gen seç (gen_k).

gen_k ,(k = 1,..,N; k (modn) = i (modn) ve $k \neq i$)

Eğer $gen_k = 0$ ise A2 ye git. Yeni bir gen seçerek süreci yeniden başlat. Aksi halde

A5. gen_i ile aynı talep ve gen_k ile aynı arz merkezine sahip olan gen_r (r = i+k-j) için,

Eğer $gen_r = 0$ ise A2 ye git. Yeni bir gen seçerek süreci yeniden başlat.Aksi halde

A6. gen_i, gen_j, gen_k ve gen_r arasından en küçüğünü seç (gen_{min}). En küçüğü sıfır yap, diğer genleri arz ve talep kısıtlarına uygun olarak düzelt.

Eğer $gen_{min} = gen_i$ ise, $gen_i = 0$, $S = S-1$, $gen_k = gen_k - gen_{min}$, eğer $gen_k = 0$ ise $S = S-1$, $gen_j = gen_j + gen_{min}$, $gen_r = gen_r + gen_{min}$. Aksi halde

Eğer $gen_{min} = gen_k$ ise yukarıdaki eşitliklerde sadece i ile k indisleri karşılıklı yer değiştirirler.Aksi halde

Eğer $gen_{min} = gen_j$ ise $gen_j = 0$, $S = S–1$, $gen_r = gen_r$-gen_{min}, eğer $gen_r = 0$ ise $S = S–1$, $gen_i = gen_i+gen_{min}$, $gen_k = gen_k+gen_{min}$. Aksi halde

Eğer $gen_{min} = gen_r$ ise yukarıdaki eşitliklerde sadece j ile r indisleri karşılıklı yer değiştirirler.

A7. S≤m+n–1 oluncaya kadar tamir sürecini tekrarla.

V) SAYISAL BİR ÖRNEK

Bu bölümde, önerilen GA nın çaprazlama ve tamir aşamalarını açıklamak üzere Sadagopan ve Ravindran (1982) den Tablo 1 de sunulan ve optimal çözümü bilinmekte olan problem ele alınmaktadır. Problemin çözümü için, yukarıda açıklanmış olan GA, Delphi programlama dilinde kodlanmış ve Mobil AMD Turion 64 (ML34), 1,79 GHz, 512 MB RAM li bilgisayarda 3 kere koşturulmuştur. GA nın parametreleri: kitle hacmi = 10, çaprazlama oranı = 0,40, maksimum nesil sayısı = 250 olarak alınmıştır. Koşum sayısının az olması problem boyutunun küçüklüğünden kaynaklanmaktadır.

Tablo : 1
Maliyet Matrisi

	Değişken maliyet c_{ij}				Sabit maliyet f_{ij}				
	t_1	t_2	t_3	t_4	t_1	t_2	t_3	t_4	Arz
a_1	760	71	283	594	900	1000	700	800	50
a_2	594	64	170	564	900	300	700	600	15
a_3	594	69	79	202	600	200	400	0	5
Talep	25	20	15	10	25	20	15	10	

Elde edilen en iyi çözüm 30350 maliyet değeri ile, (25 20 00 05 00 00 15 00 00 00 00 05) dir. Bu çözüm Sadagopan ve Ravindran (1982) deki optimal çözümün aynısıdır. Aşağıda çözümün elde edilmesi sırasında GA nın çaprazlama ve tamir işlemleri için birer açıklayıcı örnek sunulmaktadır. Başlangıç kitlesinden çaprazlanmak üzere iki birey (kr_1 ve kr_2) seçilmiş olsun. İki basamak bir yüklemeyi göstermek üzere; kr_1 : (25 05 15 05 00 15 00 00 00 00 00 05) kr_2 : (25 20 00 05 00 00 15 00 00 00 00 05)

$b_1 = 2$, $b_2 = 2$, $b_3 = 0$, $c_1 = 0$, $c_2 = 2$, $c_3 = 2$, $c_4 = 0$ ile ortaya çıkan 0-1 programlama probleminin iki seçenekli çözümünden ($x_2 = 0$, $x_3 = 1$, $x_6 = 1$, $x_7 = 0$ veya $x_2 = 1$, $x_3 = 0$, $x_6 = 0$, $x_7 = 1$) aşağıdaki iki yavru kromozom oluşmaktadır.

kr_{y1}: (25 12 08 05 00 08 07 00 00 00 00 05) kr_{y2}: (25 13 07 05 00 07 08 00 00 00 00 05)

Çaprazlama işlemi sonucunda oluşan yeni bireyler optimal çözüm adayı olma kriterini taşımamaları durumunda bir tamir sürecinden geçmektedirler. Yukarıdaki problem için söz konusu kriter basitçe, 6 adetten fazla yüklemenin olmamasıdır. Yeni birey olarak 8 adet sıfırdan farklı gene sahip olan bir kromozom seçilmiş olsun. kr:(10 17 15 08 12 03 00 00 03 00 00 02). Sıfırdan farklı genler arasından rastgele olarak 2. gen seçilsin.bu gen ile aynı arz merkezine ait olan sıfırdan farklı genler arasından 1. gen ve ayrıca aynı talep merkezine ait olan sıfırdan farklı genler arasından 6. gen rastgele seçilmiş olsun. Bu durumda 1., 2., 6. ve zorunlu olarak 5. genler arasında yeniden dağıtım ile yedi adet sıfırdan farklı gen içeren yeni bir kromozom (07 20 15 08 15 00 00 00 03 00 00 02) oluşmaktadır. Aynı süreç tekrarlanarak optimal çözüm adayı olabilecek bir kromozom (05 20 15 10 15 00 00 00 05 00 00 00) elde edilebilmektedir.

SONUÇLAR

Bu makalede değişken maliyetlerin yanı sıra sabit maliyetleri de içeren ve 0-1 programlama problemi olarak formüle edilebilen ulaştırma problemini (SMUP) çözmek için reel kodlu bir genetik algoritma sunulmuş ve DELPHI lisanında kodlanarak koşturulmuştur. Hesaplama sonuçları söz konusu algoritmanın bu problemi çözmek için uygun olduğunu ortaya koymaktadır. Çalışmada, probleme özgü bütün bilgi algoritmaya dahil edilerek problem kısıtlarını otomatikman sağlayan bir çaprazlama operatörü önerilmektedir.

GA lar yapıları gereği uzun işlem süreleri gerektirmektedirler ve standart doğrusal programlama algoritmasına dayanan özel optimizasyon teknikleri ile rekabet edemezler. Ancak standart ulaştırma metodlarının kullanılamayacağı sabit maliyetli ya da doğrusal olmayan ulaştırma problemlerini çözmek için çok gerekli tekniklerdir. GA lar yavaş çalıştıkları için, bir GA nın performansını belirleme de optimal çözüme yakınlığın yanı sıra çözümün elde edilme süresi de önemli rol oynamaktadır. SMUP nin temel çözümleri, kısıtlara uygun olarak en fazla m+n-1 adet yükleme içeren çözümlerdir. Bu çalışmada çözüm süresinin kısaltılması amacı ile araştırmayı uygun (feasible) çözüm uzayında değil sadece temel çözümler üzerinde sürdürmek için çaprazlamanın ardından bir tamir operatörü önerilmektedir.

Önerilen çaprazlama ve tamir operatörlerinin işleyiş mantığı elbette tamamen orijinal değildir. Ancak çözümlerin gösterim tarzı ve bunun gereği olarak kodlama, çaprazlama ve tamir süreçleri için sunulan algoritma parçaları tamamen bu çalışmaya özgü yeniliklerdir. Ulaştırma problemleri için bir çözümün en doğal gösterimi iki boyutlu bir yapı olmasına rağmen, bu çalışmada vektör gösterimi tercih edilmiştir. Bunun nedeni, çalışmanın temel amaçlarından birinin, çok daha yaygın olan tek boyutlu kısıtlı optimizasyon problemlerini çözmede de kullanılabilecek esnek yapıda algoritmalar sunmak olmasıdır.

Kısıtlı optimizasyon problemlerinin GA ile çözülmelerinde karşılaşılan zorluklardan biri, karar değişkenlerinin alabilecekleri değerler üzerindeki sınırların algoritmaya dahil edilmesidir. Bu çalışmada bunu başarmak için başlangıç kitlesi oluşturulurken karar değişkenleri rasgele seçilmekte ve seçilen değişkene ilişkin arz ve talep değerlerinden küçük olanı değişkene atanmaktadır.

Ulaştırma problemlerini çözmek için kullanılan standart metodların hepsi arz-talep dengesini dikkate almasına karşın, önerilen GA, dengeli ve dengesiz problemler arasında herhangi bir ayrım yapmaksızın her iki tür problemi de çözebilecek niteliktedir. Ayrıca önerilen algoritma çok az değişiklikle diğer UP türlerine de uyarlanabilmektedir. Örneğin; standart ulaştırma problemini çözmek için değerlendirme fonksiyonunu uygun şekilde değiştirmek ve tamir sürecini ya tamamen kaldırmak ya da basit bir mutasyon süreci ilave etmek yeterli olacaktır.

Çok önemli bir gelecek araştırma konusu, otomatikman m+n-1 veya daha az yüklemeli bireyler üretebilen bir çaprazlama operatörünün geliştirilmesine çalışmaktır. Bunun başarılabilmesi durumunda tamir sürecine olan gereksinim tamamen ortadan kalkacak ve sonuçta optimal çözüme ulaşma süresi önemli ölçüde azalacaktır. Bir diğer gelecek araştırma konusu ise, değişken arz ve/veya talep durumunun söz konusu olduğu UP nin GA ile çözülmesidir. Ayrıca, araştırmacılar, miktar indirimi ve/veya kullanılan taşıma araçlarının tam kapasite ya da eksik yüklenmelerine göre birim taşıma maliyetlerinin değişmesi durumunda ortaya çıkan kesikli amaç fonksiyonuna sahip problemlerin GA ile çözülmelerine yönelik algoritmalar geliştirebilirler.

KAYNAKÇA

ABD EL-WAHED, W F. and LEE, S M., (2006), "*Interactive Fuzzy Goal Programming for Multi-Objective Transportation Problems*", Omega-The International Journal of Management Science, Vol. 34, No: 2, pp. 158-166.

ADLAKHA, V. and KOWALSKI, K., (2003), "*A Simple Heuristic for Solving Small Fixed-Charge Transportation Problems*", Omega-The International Journal of Management Science, Vol. 31, pp. 205-211.

ADLAKHA, V.; KOWALSKI, K.; VEMUGANTI, R.R. and LEV, B., (2007), "*More-for-Less Algorithm for Fixed-Charge Transportation Problems*", Omega-The International Journal of Management Science, Vol. 35, pp. 116-127.

BALINSKI, Michel, L., (1961), "*Fixed Cost Transportation Problems*", Naval Research Logistic Quarterly, Vol. 8, No: 1, pp. 41-54.

BARR, R.S.; GLOVER, R.S. and KLINGMAN, D., (1981), "*A New Optimization Method for Large Scale Fixed Charge Transportation Problems*", Operations Research, Vol. 29, No: 3, pp. 448-463.

CHIEN, William, T., (1993), “*Determining Profit-Maximizing Production/Shipping Policies in a one to one Direct Shipping Stochastic Environment*”, European Journal of Operational Research, Vol. 64, pp. 83-102.

COOPER, L. and DREBES, C., (1967), “*An Approximate Algorithm for the Fixed Charge Problem*”, Naval Research Logistic Quarterly, Vol. 14, pp. 101-113.

COOPER, Leon, (1975), “*The Fixed Charge Problem-I: A New Heuristic Method”,* Computers &Mathematics with Applications, Vol. 1, pp. 89-95.

DIABY, Moustapha, (1991), “*Successive Linear Approximation Procedure for Generalized Fixed-Charge Transportation Problems”,* Journal of Operational Research Society, Vol. 42, pp. 991-1001.

DORNDORF, U.; DREXL, A.; NIKULIN, Y. and PESCH, E., (2007), “*Flight Gate Scheduling: State-of-the-art and Recent Developments”,* Omega-The International Journal of Management Science, Vol. 35, No: 3, pp. 326-334.

DRENZLER, D.R., (1969), “*An Approximate Method for the Fixed Charge Problem”,* Naval Research Logistic Quarterly, Vol. 16, pp. 411-416.

GEN, M. and CHENG, R., (2000), *Genetic Algorithms and Engineering Optimization*, New York, John Wiley &Sons, pp. 512.

GEN, M. and SYARIF, A., (2003), *Multi-Stage Chain Network by Hybrid Genetic Algorithms with Fuzzy Logic Controller,* Fuzzy Sets Based Heuristics for Optimization, New York, Springer Verlag , pp. 181-196.

GOLDBERG, David,.E., (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, *Massachusetts*., Addison Wesley Publishing,.pp. 412

GOTTLIEB, J. and PAULMANN, L., (1998), “Genetic Algorithms for the Fixed Charge Transportation Problem*”, In Proceedings of IEEE International Conference on Evolutionary Computation*, , pp. 330-335, Anchorage, USA.

GRAY, Paul, (1971), “*Exact Solutions of the Fixed Charge Transportation Problem*”, Operations Research, Vol. 19, pp. 1529-1538.

GREFENSTETTE, John, (1986), “*Optimization of Control Parameters for Genetic Algorithms”,* IEEE Transactions on Systems, Man, Cybernetics, Vol. 16, pp. 122-128.

Hirsch, W. M. and DANTZIG, G. B., (1968), “*The Fixed Charge Problem”,* Naval Research Logistic Quarterly, Vol. 15, pp. 413-424.

HITCHCOCK, Frenk L., (1941), “*The Distribution of a Product from Several Sources to Numerous Localities”,* Journal of Mathematical Physic, Vol. 20, pp. 224-230.

HOLLAND, John. H., (1992), *Adaptation in Natural and Artificial Systems,* USA, MIT Press, p. 211.

IDA, K.; TOHYAMA, H.; TERAMATSU, C. And FUTATANI, Y., (2004), “A Genetic Algorithm for Fixed Charge Transportation Problem”, *Proceedings of the fifth Asia-Pacific Industrial Engineering Management Systems*, Vol. 5, pp.3211-3215., Australia.

JIMENEZ, F. and VERDEGAY, J. L., (1996), “Interval Multiobjective Solid Transportation Problems via Genetic Algorithms”, *Proceedings of Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pp-787-792, Granada, Spain.

KENNINGTON, J. L. and UNGER, V. E., (1976), “*A New Branch and Bound Algorithm fort he Fixed Charge Transportation Problem”,* Management Science, Vol. 22, No: 10, pp. 1116-1126.

KOWALSKI, Krzysztof, (2005), “*On the Structure of the Fixed Charge Transportation Problem”,* International Journal of Mathematical Education in Science and Technology”, Vol. 36, No: 8, pp. 879-888.

KOWALSKI, K and LEV, B., (2007), “*New Approach to Fixed Charge Problems*”, International Journal of Management Science and Engineering Management, Vol. 2, No: 1, pp. 75-80.

KUHN, H.W. and BAUMOL, W.J., (1962), “*An Approximation Algorithm for the Fixed Charge Transportation Problem”,* Naval Research Logistic Quarterly, Vol. 9, pp. 1-15

KUO, C. and GILLIAN, M. N., (2007), “*A Mathematical Modelling Approach to Improving Locomotive Utilization at a Freight Railroad”,* Omega- The International Journal of Management Science, Vol. 35, No: 5, pp. 472-485.

LI, Y.; GEN, M. and IDA, K., (1998), “*Genetic Algorithms for the Fixed Charge Transportation Problem*”, Beijing Mathematics, Vol. 4, No: 2, pp. 239-249.

MICHALEWICZ, Z.; VIGNAUX, G. A. and HOBBS, M., (1991), “*A Nonstandard Genetic Algorithm for the Nonlinear Transportation Problem*” Orsa- Journal on Computing, Vol. 3, No: 4, pp. 307-316.

MICHALEWICZ, Zbigniew, (1993), “*A Hierarchy of Evolution Programs*” Evolutionary Computation, Vol. 1, No: 1, pp. 51-76.

MICHALEWICZ, Zbigniew, (1996), *Genetic Algorithms+Data Structures = Evolution Program*, New York, Springer Verlag , pp. 387.

MİZUNUMA, H. and WATADA, J., (1995), “*Fuzzy mixed integer programming based on genetic algorithm and its application to resource distribution*", *Japanese Journal of Fuzzy Theory and Systems*, Vol. 7 No: 1, pp.97-116.

MURTY, Katta, G., (1968), “*Solving the Fixed Charge Problem by Ranking the Extreme Points”*, Operations Research, Vol. 16, pp. 268-279.

PALEKAR, U. S.; KARWAN, M. H. and ZIONTS, S. A., (1990), “*A branch-and bound Method for the Fixed Charge Transportation Problem*”, Management Science, Vol. 36, pp. 1092-1105.

SADAGOPAN, S. And RAVINDRAN, A. (1982), “*A Vertex Ranking Algorithm for the Fixed-Charge Transportation Problem*”, Journal of Operational Theory and Application, Vol. 37, pp. 221-230.

SAKAWA, Masatoshi, (2002), *Genetic Algorithms and Fuzzy Multiobjective Optimization*", Boston, Kluwer Academic Publishers, pp. 304.

SANDROCK, Keith, (1988), "*A Simple Algorithm for Solving Small Fixed Charge Transportation Problems*", Journal of Operational Research Society, Vol. 39, No: 5, pp. 467-475.

SPEARS, W. M. and De JONG, K. A., "On the Virtues of Parameterized Uniform Crossover", *Proceedings of the Fourth International Conference on Genetic Algorithms*, pp. 230-236, La Jolla ,CA, USA.

STEINBERG, David, I., (1970), "*The Fixed Charge Problem*", Naval Research Logistic Quarterly, Vol. 17, pp. 217-235.

. SUN, M.; ARONSON, J. E.; McKEOWN, P. G. and DRINKA, D., (1998), "*A Tabu Search Heuristic Procedure for the Fixed Charge Transportation Problem*", European Journal of Operational Research, Vol. 106, pp. 441-456.

SYARIF, A. and GEN, M., (2003), "*Solving Exclusionary Side Constraint Transportation Problem by Using a Hybrid Spanning Tree-based Genetic Algorithm*", International Journal of Intelligent Manufacturing, Vol. 14, pp. 389-399.

VİGNAUX, G.A. and MİCHALEWİCZ, Z., (1991), "*A genetic algorithm for the linear transportation problem*", IEEE Transactions on Systems, Man and Cybernetics, Vol. 21, pp.445-452.

WALKER, Warren, .E., (1976), "*A Heuristic Adjacent Extreme Point Algorithm for the Fixed Charge Problem*", Management Science, Vol. 22, pp. 587-596.