

Atama problemi için yeni bir çözüm yaklaşımı

Adalet ÖNER*, Füsun ÜLENGİN

İTÜ İşletme Fakültesi, Endüstri Mühendisliği Bölümü, 34357, Maçka, İstanbul

Özet

Bu çalışmanın amacı klasik atama problemi için özgün bir çözüm yöntemini tanıtmaktır. Atama probleminin çözümünde en çok bilinen yöntem Macar yöntemidir. Bu yöntemde maliyet matrisi her seferinde sistematik bir şekilde yeni bir indirgenmiş matrise dönüştürülerek çözüme gidilmektedir. Yöntem gereği indirgenmiş maliyet matrisindeki sıfır elemanlar en az sayıda çizgi ile kapatılmakta ve buna göre matris üzerinde işlem yapılmaktadır. Ancak problemin büyüklüğü arttıkça ve indirgenmiş maliyet matrisinde sıfır eleman sayısı çoğaldıkça, matristeki sıfır elemanlarını kapatmak üzere gereken en az sayıda çizgi sayısı ve bu çizgilerin nasıl çizilmesi gerektiği sorunu ortaya çıkar. Bu çalışmada Macar yöntemindeki bu boşluğu doldurmak üzere özgün bir yöntem tanıtılmaktadır.

Anahtar Kelimeler: Atama problemi, Macar yöntemi.

A new solution approach for assignment problem

Abstract

The purpose of this study is to present a new solution approach for the assignment problem. In assignment problem, there are (m) individuals are to be assigned to (m) jobs. If the individual (i) assigned to job (j), the cost incurred will be (c_{ij}) , and accordingly the cost matrix is denoted by C. It is desired to find the minimal cost assignment or a one-to-one matching of individuals to jobs. There are many solution methods for this problem, but the simplicity and robustness of the Hungarian method makes it the best known method among all others. The Hungarian method solves the problem by converting the cost matrix into a reduced matrix systematically at each iteration. A part of this process is finding fewest number of lines to cover all zero elements in the reduced matrix. When the size of the problem increases and reduced matrix contains many zeros, it is a tedious task to find minimum number of lines and the way of drawing them. The Hungarian method has an ambiguity at this point. A solution method is presented in this study to eliminate this ambiguity. A systematic and simple procedure is defined to find the fewest number of lines to cover all zero elements in the reduced matrix.

Keywords: Assignment problem, Hungarian method.

*Yazışmaların yapılacağı yazar: Adalet ÖNER. aoner@dho.edu.tr; Tel: (216) 395 26 30 dahili: 3660.

Bu makale, birinci yazar tarafından İTÜ İşletme Fakültesi'nde tamamlanmış "Deniz Harp Okulu'nda çizelgeleme/ atama problemleri ve çözüm yöntemleri" adlı doktora tezinden hazırlanmıştır. Makale metni 27.03.2002 tarihinde dergiye ulaşmış, 25.12.2002 tarihinde basım kararı alınmıştır. Makale ile ilgili tartışmalar 31.05.2003 tarihine kadar dergiye gönderilmelidir.

Giriş

Yöneylem Araştırması'nda en çok tanınan problemlerden biri "Atama" (Assignment) problemidir. Atama probleminde yapılması gereken (m) adet görev vardır. Bu görevleri yapmaları için de (m) ayrı kişi bulunmaktadır. Herhangi bir (i) kişinin (j) görevine verilmesi durumunda (c_{ij}) maliyeti doğar. Her göreve mutlaka bir kişinin verilmesi ve bir kişinin sadece tek bir göreve atanması koşuluyla en küçük toplam maliyetle bir görevlendirme planı yapılması istenir. Diğer bir deyişle, en küçük toplam maliyeti doğuracak birebir kişi-görev eşleşmesinin bulunması istenmektedir. Bu problemin matematiksel modeli:

$$x_{ij} = \begin{cases} 1 & \text{Eğer } i. \text{ kişi } j. \text{ göreve atanacaksa} \\ 0 & \text{Eğer } i. \text{ kişi } j. \text{ göreve atanmayacaksa} \end{cases}$$

$$\text{Min. } \sum_{i=1}^m \sum_{j=1}^m c_{ij} x_{ij} \quad (1)$$

$$\text{s.t. } \sum_{j=1}^m x_{ij} = 1 \quad \forall i=1,2,\dots,m \quad (2)$$

$$\sum_{i=1}^m x_{ij} = 1 \quad \forall j=1,2,\dots,m \quad (3)$$

$$x_{ij} = 0 \text{ veya } 1 \quad \forall i=1,2,\dots,m \\ \forall j=1,2,\dots,m \quad (4)$$

Matris formunda ifade edildiğinde ise:

$$\text{Min. } C X \quad (5)$$

$$\text{s.t. } A X = 1 \quad (6)$$

$$x_{ij} = 0 \text{ veya } 1 \quad \forall i, j \quad (7)$$

Klasik çözüm yöntemleri

Atama probleminin çözümü birçok değişik yolla elde edilebilir. Örnek olarak kısıtlara uyan tüm alternatifler belirlenerek aralarından en küçük maliyete sahip olan seçilirse çözüm elde edilmiş olur. Ancak dikkat edilirse, Atama probleminde

(m!) uygun geçerli çözüm bulunmaktadır. Problemin büyüklüğü (m) arttıkça, uygun geçerli çözüm sayısı çok büyük bir hızla artacaktır. Örnek olarak $m = 100$ olduğunda uygun çözüm alternatifi sayısı $100! = 9.33 \times 10^{157}$ olacaktır. Bu olağanüstü çok sayıdaki uygun alternatiflerin herbirinin teker teker belirlenmesi bilgisayarda dahi oldukça uzun bir zaman gerektirecektir.

Diğer bir çözüm yöntemi ise problemi simpleks yöntemle çözmektir. Atama probleminin kısıt katsayıları matrisi "unimoduler" bir karaktere sahiptir. Diğer bir deyişle, A kısıt katsayıları matrisinin her alt kare matrisinin determinanı ya -1 veya +1 veya 0 değerini almaktadır (Bazaraa v. diğ., 1990). Eğer bir tamsayılı programlama probleminde kısıtlara ait katsayı matrisi "unimoduler" ise ve sağ taraf vektörü tamsayılı elemanlardan oluşmuşsa, o takdirde tüm temel çözümler de tamsayılı olacaktır. Dolayısı ile böyle bir problemin tamsayı olma kısıtları kaldırılarak simpleks yöntemle çözüldüğünde bulunan en iyi sonuç, tamsayılı problemin de en iyi sonucu olacaktır. Atama problemi bu tarife uyan bir problemidir. O halde, orijinal olarak bir tamsayılı (0-1 binary) programlama problemi olan Atama probleminde değişkenlerin 0 veya 1 olma kısıtları kaldırılarak simpleks yöntemle çözüldüğünde tamsayılı çözümü de bulunmuş olur. Problemden kısıtlar "eşitlik" halinde olduğundan (2m) yapay değişken eklenerek klasik simpleks yöntemiyle çözüme gidilebilir. Problemin simpleks yöntemle çözümünden bahsederken, hemen bir başka çözüm yolu daha gündeme gelmektedir. Eğer Atama problemi bir "Ulaştırma-Transportation" problemi olarak ele alınırsa (ki aslında Atama problemi, Ulaştırma probleminin satır sayısının sütun sayısına eşit olduğu ve sağ taraf değerlerinin 1 olduğu özel bir halidir), o zaman simpleks yöntemin doğrudan maliyet matrisi üzerindeki uygulaması olan "Ulaştırma Tekniği" (Transportation Technique) ile problemi çözmek mümkündür. Bu durumda sadece (mxm) boyutlarında bir matris üzerinde işlemler yürütülecektir. Bununla beraber Ulaştırma Tekniği ile çözüm yapılırken bir sorunla karşı karşıya kalınacaktır. Bilindiği gibi

Ulaştırma Tekniği'nde, önce başlangıç temel uygun çözüm bulunur. Bu amaçla Kuzey Batı Köşesi Yöntemi, Vogel Yaklaşım Yöntemi veya En Küçük Maliyet Yöntemi gibi yöntemler kullanılır. Atama problemi Ulaştırma tekniği ile çözülmeye çalışıldığında başlangıç temel uygun çözüm için hangi yöntem kullanılırsa kullanılsın dejenere bir problemle karşı karşıya kalınır ve devamında her iterasyonda temel çözümler daima dejenere kalır. Bunun sebebi atama probleminin özel yapısıdır. Kolaylıkla gösterilebilir ki rank (\mathbf{A}) = $2m-1$ 'dir. Bunun anlamı, temel çözümlerde en fazla $2m-1$ adet karar değişkeni bulunacaktır. Diğer taraftan, problemin kısıtlarının getirdiği bir zorunluluk gereği temel değişkenlerden sadece (m) tanesi sıfırdan farklı pozitif değer almalıdır. Geri kalan $(m-1)$ değişken sıfır değeri ile temel çözüm içinde görünecektir. Bu durum yüksek derecede dejenere bir problemle karşı karşıya olduğumuza işaret eder. Yüksek dereceli dejenarasyon durumu, problemi simpleks yöntemle çözmeye çalıştığımızda da karşımıza çıkacaktır. Simpleks yöntemde veya Ulaştırma Tekniğinde her ne kadar dejenarasyon hali büyük bir sorun teşkil etmese de, amaç fonksiyonunu geliştirmeyen, bir nevi boşa geçen iterasyonlarla (pivot işlemleri) ile uğraşmamız gerekeceği açıktır. Deneysel çalışmalarda rapor edilen bir saptamaya göre $m \geq 500$ olduğunda, çözüm için simpleks yöntem kullanılıyorsa, iterasyonların %90'ı dejenere pivotlarla geçmektedir (Bazaraa v. diğ., 1990).

Diğer bir çözüm yolu ise, problemi dual simpleks yöntemle çözmektir. Matris formunda dual problem aşağıdaki gibi yazılabilir:

$$\text{Max. } w^* \quad (8)$$

$$\text{s.t. } w \mathbf{A} \leq \mathbf{C} \quad (9)$$

$$w_j \text{ sınırlandırılmamış } \forall j=1,2,\dots,2m \quad (10)$$

Dual problemde değişkenler sınırlandırılmamış olduğundan simpleks yöntemi uygulayabilmek için değişkenlerde aşağıdaki dönüşümü yapmak gerekir:

$$u_i = u_i' - u_i'' \quad \forall i=1,2,\dots,m \quad (11)$$

$$v_j = v_j' - v_j'' \quad \forall j=1,2,\dots,m \quad (12)$$

Bu değişimle birlikte, kısıt sayısı kadar, diğer bir ifade ile (m^2) adet dolgu (slack) değişken ekleyerek dual problemin simpleks çözümüne geçilebilir. Yüksek dereceli dejenarasyon durumu bu çözüm yolunda da karşımıza çıkacaktır.

Buraya kadar anlatılan klasik çözüm yöntemleriyle bilgisayar çözümleri dikkate alındığında, (m) sayısı arttıkça hızla artan hafıza gereksinimi ve yüksek dejenarasyon hali nedeniyle artan işlem süresiyle karşı karşıya olduğumuz açıktır. Bu nedenle klasik yöntemlerden farklı, ancak onların özelliklerini kullanan özel çözüm yöntemleri de bulunmaktadır.

Özel çözüm yöntemleri

Barr ve diğerleri (1977) tarafından önerilen "Değişen Yol Temeli" (Alternating path basis) ismi verilen çözüm yöntemi primal simpleks yöntemine dayanan bir çalışmadır. Problemi network simpleks algoritması ile çözerken, sadece "kuvvetli olanaklı çözümler" (strongly feasible bases) araştırıldığından dejenere pivotlardan sakınılmaktadır. Böylece, işlem süresi açısından primal-simpleks yönteminin klasik uygulamasına göre işlem süresi açısından %25'e varan oranlarda tasarruf edildiği rapor edilmiştir. Primal simpleks tabanlı diğer çalışmalar ise Akgül (1993) ve Hung (1983) tarafından ortaya konmuştur.

Glover ve Klingman (1986) tarafından rapor edilen çalışmada ise "Ardışık en kısa yol" (Successive shortest path) ismi verilen çözüm yöntemi geliştirilmiştir. Problem ardı ardına çözülecek en kısa yol problemi biçiminde modellenerek çözülmektedir. Bu yeni yaklaşımın işlem zamanı açısından yukarıda bahsedilen "Değişen Yol Temeli" algoritmasından çok daha hızlı olduğu gösterilmiştir.

Balinski (1985, 1986) Atama probleminin özelliklerini ve çözüm bölgesini tarif ederek

Dual-Simpleks tabanlı bir algoritma geliştirmiş, ayrıca dual-simpleks tabanlı algoritmaların primal-simpleks tabanlı algoritmalara karşı daha uygun olduğunu göstermiştir.

Atama probleminin çözümü yönünde yukarıdaki çalışmalardan başka daha birçok özel çözüm yöntemi geliştirilmiştir. Bu çalışmalar genellikle dual veya primal simpleks yöntemi için daha akıllıca pivot seçimlerini hedef almaktadır. Diğer bir grup çalışmada ise ağ teorisi bilgileri ile problemi farklı biçimde modelleyerek çözme gayreti görülmektedir. Ancak birçoğu karmaşık işlemlere dayanmakta ve temel olarak çözüm hızı üzerinde yarışmaktadırlar. Ancak hiçbiri Macar çözüm yöntemi kadar bilinmezler. Bu yöntem neredeyse Atama problemi ile adı birlikte anılacak kadar tanınmaktadır. Bu nedenle çok bilinmesine rağmen Macar çözüm yöntemi yeniden kısaca tanıtılacaktır. Bu çalışmanın konusu olan çözüm yöntemi önerisi Macar çözüm yönteminin üzerine kurulduğundan bu kısa tanıtım gerekli görülmektedir.

Macar çözüm yöntemi

Macar çözüm yöntemi, Kuhn (1955) tarafından geliştirilmiş sade, kolayca anlaşılabilen ve son derece etkili bir çözüm yöntemidir. Macar yöntemi aşağıdaki biçimde ifade edilmektedir:

Adım-1: Maliyet matrisinin her satırı için, satırdaki en küçük değere sahip elemanı satırdaki tüm elemanlardan çıkarınız.

Adım-2: Yukarıdaki işlemten sonra maliyet matrisinin her sütunu için, sütundaki en küçük değere sahip elemanı sütundaki tüm elemanlardan çıkarınız.

Adım-3: Yukarıdaki işlemlerden sonra matrisin yeni durumuna “indirgenmiş matris” denilmektedir. İndirgenmiş matris üzerinde oluşan sıfır elemanlarını kapatmak üzere gerekli satır ve sütunların üzerine çizgi çekiniz. En az sayıda çizgi kullanılmalıdır. Eğer kullanılan en az çizgi sayısı maliyet matrisinin boyutu olan (m) sayısına eşitse o takdirde en iyi çözüm bulunmuş demektir. İşlemi durdurunuz, Aksi halde Adım-4’e gidiniz.

Adım-4 : Sıfır elemanlarını kapatmak için satır ve sütunlar üzerine çizilmiş çizgilerin kapatmadığı elemanlar arasında en küçük değere sahip olanını bulunuz. Bu değeri üzerinden çizgi geçmeyen tüm elemanlardan çıkarınız ve üzerinden iki çizgi geçen elemanlara ekleyiniz. Yeni bir indirgenmiş matris elde edilmiştir. Adım-3’e dönünüz.

Görüldüğü gibi Macar yönteminin sade anlatımı dikkat çekicidir. Simpleks yöntemin ihtiyaç duyduğu ve bilgisayar hafızası ihtiyacını arttıran yapay değişkenlerin eklenmesi ve dejenarasyon problemiyle uğraşmadan doğrudan maliyet matrisi üzerinde işlem yapmaktadır. Bununla birlikte problem büyüklüğü arttıkça, bilgisayarda çözüm gündeme geldiğinde, üçüncü adımda istenen tüm sıfır elemanları kapatmak üzere en az sayıda çizgilerin çizilmesi işi için sistematik bir yöntemin tarif edilmesi gerekmektedir.

Problemin büyüklüğü arttıkça ve matris üzerinde sıfır eleman sayısı çoğaldıkça sıfır elemanları kapatmak üzere gereken en az sayıda çizgi sayısının ne olduğu ve bu çizgilerin maliyet matrisinde hangi satır ve sütunlar üzerine çizilmesi gerektiği konusunda literatürde çeşitli çalışmalar rapor edilmiştir.

Bunlardan ilki Gillett (1976) tarafından önerilen basit ve etkili bir sezgisel yöntemdir. Küçük problemler için kolay ve basitçe uygulanabilir bir yapıdadır. Bu çalışmada Macar yöntemi biraz değiştirilerek işlemler yürütülmektedir. Diğer bir yöntem ise Bazaraa ve diğerleri (1990) tarafından önerilen en yüksek ağ akış yöntemidir. Bu yaklaşımda bir $G(V,E)$ çizgesi kurulmaktadır. Maliyet matrisindeki satırları 1,2,3...,m düğümleri temsil eder. Sütunları ise $m+1, m+2, m+3, \dots, 2m$ düğümleri temsil etmektedir. Eğer indirgenmiş matriste (i, j) elemanı sıfır ise o zaman (i) düğümü ile (m + j) düğümü arasında bir bağ çizilmektedir. Bu düğümlere ek olarak bir kaynak K düğümü ile bir son S düğümü daha ağa eklenmektedir. K düğümü ile satırları temsil eden tüm düğümler arasında bağlar çizilir. Aynı şekilde S düğümü ile sütunları temsil eden tüm düğümler arasında

bağlar çizilir. Bu şekilde oluşan ağ üzerindeki tüm bağların akış kapasiteleri 1 birim olarak sınırlandırılır. Bu ağda K düğümünden S düğümüne en yüksek akış miktarı Macar yönteminde üçüncü adımda gereken en az sayıdaki çizgi sayısını vermektedir. Ayrıca bu ağ akış problemi bilinen etiketleme yöntemi ile çözülmüşse, üzerinden akış geçen düğümler maliyet matrisinde çizgilerin çekileceği satır ve sütunları vermektedir. Ancak Macar yönteminin her döngüsünde maliyet matrisini bırakıp yeni bir ağ kurmak ve bu ağ üzerinden geçecek en yüksek akış miktarını hesaplamak gerekecektir. Oysa Gillett'in yönteminde işlemler maliyet matrisi üzerinde yürütülmektedir.

Diğer bir yöntem ise Papadimitrou ve Steiglitz (1982) tarafından önerilen "ikiye ayrılabilir çizgelerde ağırlıklı eşleme" (weighted matching problem in bipartite graphs) yöntemidir. Bu yöntem oldukça karmaşık olup derinlemesine ağ bilgisi ve ağırlıklı eşleme yöntemine aşina olmayı gerektirmektedir. Bu yaklaşımda da bir ağ kurularak düğümler arasında eşleme yapılmak suretiyle Macar yöntemindeki maliyet matrisi üzerine çizilecek çizgilerin sayısı ve hangi satır/sütunların üzerine çizilmesi gerektiği bulunmaya çalışılmaktadır.

Lotfi (1989) tarafından gerçekleştirilen çalışmada Gillett yönteminin sakıncalı yönleri gösterilmiş ve bu yöntemin belirli problemler için çalışmayacağı ispat edilmiştir. Ayrıca Bazaara ve diğerleri (1990) tarafından önerilen en yüksek ağ akış yönteminin doğrudan maliyet matrisi üzerinde uygulaması olan bir yöntem önermiştir. Bu yöntemde maliyet matrisine yeni bazı sütun ve satırlar eklenerek en yüksek ağ akış problemi bu yeni eklenmiş matris üzerinde çözülmekte ve buna göre maliyet matrisi üzerinde çizilecek çizgiler bulunmaktadır. Ancak yöntem, kendisi gibi maliyet matrisi üzerinde çalışan Gillett yönteminden daha karmaşıktır.

Geliştirilen çözüm yöntemi

Bütün bu çalışmaların arasında, Macar yönteminin ruhuna da uygun olacak şekilde, maliyet matrisinin üzerinden ayrılmadan

işlemleri yürütecek daha sade ve kolay anlaşılabilir bir yöntem üzerinde çalışılmıştır. Geliştirilen çözüm yönteminde ilk iki adım Macar yöntemindeki ilk iki adımla aynıdır. Macar yönteminin üçüncü adımı ona uygun sade bir şekilde çizgilerin sistematik bir yöntemle çizilmesini sağlayacak bir yöntemle tamamlanmaktadır. Çizgilerin çekilmesi işleminin ana hatları aşağıdadır:

- Sıfır eleman sayısı 1 olan bir sütun seçerek sıfır elemanın bulunduğu satıra yatay bir çizgi çiziniz. Bu satıra çizgi çekildiğinde başka sütunlarda üzeri çizilen sıfır elemanları da olabilir. Üzeri çizilen sıfır elemanları nedeniyle ilgili sütunlardaki sıfır eleman değerlerini birer düşürünüz. Bu durumda sıfır eleman sayısı 1 olan yeni sütunlar ortaya çıkabilir. Bu işlemi sıfır eleman sayısı 1 olan hiçbir sütun kalmayıncaya kadar devam ediniz.

- Yukarıdaki işlemin aynısını satırlar için yapınız. Sıfır eleman sayısı 1 olan her satırdaki sıfır elemanı hangi sütunda ise o sütuna dikey çizgi çiziniz. Bu sütuna çizgi çekildiğinde üzeri çizilen sıfır elemanları nedeniyle ilgili satırlardaki sıfır eleman değerlerini birer düşürünüz. Bu arada bir önceki adımda çizilen yatay çizgiler eğer sıfır eleman sayısı 1 olan bir satıra çizilmişse bu yatay çizgileri kaldırınız.

- Hala üzeri kapatılmamış sıfır elemanları varsa, üzerine çizgi çekilmemiş ve sıfır eleman sayısı 2 veya daha fazla olan sütun ve satır sayılarını bulunuz. Sütun sayısı fazla ise tüm sıfırları yatay çizgilerle kapatınız. Aksi takdirde tümünü dikey çizgilerle kapatınız.

Bu işlemlerden sonra tüm sıfırlar en az sayıda çizgi ile kapatılmış olur. Bundan sonra yine Macar yöntemine dönülerek çözüme devam edilir. Geliştirilen çözüm yönteminin ayrıntıları özel durumlar da dikkate alınarak aşağıda verilmiştir:

Adım-1: Maliyet matrisinin her satırı için, satırdaki en küçük değere sahip elemanı satırdaki tüm elemanlardan çıkarınız.

Adım-2: Yukarıdaki işlemten sonra maliyet matrisinin her sütunu için, sütundaki en küçük değere sahip elemanı sütundaki tüm elemanlardan çıkarınız.

Adım-3: Yukarıdaki işlemlerden sonra elde edilen indirgenmiş matriste;

Adım 3.1: Matriste her satırda ve her sütundaki sıfır eleman sayılarını bularak bu sayıları matriste satır ve sütunların karşısına yazınız.

Adım 3.2: Sütunlardaki sıfır eleman sayılarını inceleyiniz. Eğer tüm sütunlarda sıfır eleman sayısı 2 veya daha fazla ise Adım-3.3'e gidiniz. Aksi halde Adım 3.4'e gidiniz.

Adım 3.3: En az sıfır elemana sahip sütunu seçerek bu sütunda yer alan sıfır elemanların satırlarına çizgi çiziniz. Her çizgi çekildiğinde o satırın başka sütunlarında üzeri çizilen sıfır elemanları da olabilir. Üzeri çizilen sıfır elemanlarını nedeniyle ilgili sütunlardaki sıfır eleman değerlerini birer düşürünüz. Sütunlardaki sıfır eleman sayılarını yeniden inceleyiniz. Eğer tüm sütunlarda sıfır eleman sayısı 2 veya daha fazla ise Adım-3.5'e gidiniz. Aksi halde Adım 3.4'e gidiniz.

Adım 3.4: Sıfır eleman sayısı 1 olan bir sütun seçerek sıfır elemanın bulunduğu satıra yatay bir çizgi çiziniz. Bu satıra çizgi çekildiğinde başka sütunlarda üzeri çizilen sıfır elemanları da olabilir. Üzeri çizilen sıfır elemanları nedeniyle ilgili sütunlardaki sıfır eleman değerlerini birer düşürünüz. Bu durumda sıfır eleman sayısı 1 olan yeni sütunlar ortaya çıkabilir. Bu işleme sıfır eleman sayısı 1 olan hiçbir sütun kalmayınca kadar devam ediniz.

Adım 3.5: Satırlarda sıfır eleman sayılarını inceleyiniz. Eğer tüm satırlarda sıfır eleman sayısı 2 veya daha fazla ise Adım-3.6'ya gidiniz. Aksi halde Adım 3.7'ye gidiniz.

Adım 3.6: En az sıfır elemana sahip satırı seçerek bu satırda yer alan sıfır elemanların sütunlarına çizgi çiziniz. Her çizgi çekildiğinde o sütunun başka satırlarında üzeri çizilen sıfır

elemanları da olabilir. Üzeri çizilen sıfır elemanlarını nedeniyle ilgili satırlardaki sıfır eleman değerlerini birer düşürünüz. Satırlarda sıfır eleman sayılarını yeniden inceleyiniz. Eğer tüm satırlarda sıfır eleman sayısı 2 veya daha fazla ise Adım-3.8'e gidiniz. Aksi halde Adım 3.7'ye gidiniz.

Adım 3.7: Sıfır eleman sayısı 1 olan her satırdaki sıfır elemanı hangi sütunda ise o sütuna dikey çizgi çiziniz. Bu sütuna çizgi çekildiğinde üzeri çizilen sıfır elemanları nedeniyle ilgili satırlardaki sıfır eleman değerlerini birer düşürünüz. Bu arada daha önce çizilen adımda çizilen yatay çizgiler eğer sıfır eleman sayısı 1 olan bir satıra çizilmişse bu yatay çizgileri kaldırınız.

Adım.3.8: Üzerine çizgi çekilmeyen satır ve sütunlarda sıfır elemanları varsa, sıfır eleman sayısı 2 ve daha fazla olan ve üzerine çizgi çekilmemiş sütun ve satır sayılarını bulunuz. Sütun sayısı fazla ise tüm sıfırları yatay çizgilerle kapatınız. Aksi takdirde tümünü dikey çizgilerle kapatınız. Tüm sıfırlar en az çizgi ile kapatılmıştır. Eğer çizgi sayısı (m) ise işlemi durdurunuz. Aksi takdirde Adım-4'e gidiniz.

Adım-4: Sıfır elemanlarını kapatmak için satır ve sütunlar üzerine çizilmiş çizgilerin üzerinden kapatmadığı elemanlar arasından en küçük değere sahip olanını bulunuz. Bu değeri üzerinden çizgi geçmeyen tüm elemanlardan çıkarınız, ve üzerinden iki çizgi geçen elemanlara ekleyiniz. Yeni bir indirgenmiş matris elde edilmiştir. Adım-3'e dönünüz.

Sonuç

Geliştirilen bu yöntem ile Macar yönteminin üçüncü adımındaki boşluk daha sade ve kolay anlaşılabilir bir şekilde doldurulmaktadır. Diğer tüm yöntemler önerilen bu yöntemden daha karmaşık işlemleri içermektedir ve okuyucuya anlatması daha zordur. Oysa bu yöntemde basit komutlar ile işlem yürütülmektedir. Sadelik konusunda da Gillett yöntemine karşı hayli iddialıdır, ayrıca onun belirli problemlerde girdiği sonsuz döngülere düşmemektedir. Bu haliyle eğitim maksatlarıyla olduğu kadar

bilgisayar ortamında çözümler için programlama açısından da kullanışlıdır.

Yöntem ile birçok değişik boyutlu test problemi çözülerek sonuçlarının doğruluğu kontrol edilmiştir. Değişik boyutlarda ayrı ayrı olmak üzere (100*100) boyutuna kadar olan test problemlerinin çözümünde, önerilen yöntemin çözüm süreleri açısından Gillet yöntemine göre daha yavaş olduğu, Lotfi yöntemi ile hemen hemen eşit, fakat az farkla ondan daha hızlı çalıştığı görülmüştür.

Kaynaklar

- Akgül, M. (1993). A Genuinely Polynomial Primal Simplex Algorithm for the Assignment Problem, *Discrete Applied Mathematics*, **45**, 93-115.
- Balinski, M. L., (1985). Signature Methods for the Assignment Problems. *Operations Research*, **33**, 527-536.
- Balinski, M. L., (1986). A Competitive (Dual) Simplex Method for the Assignment Problem. *Mathematical Programming*, **34**, 125-141.
- Barr, R., Glover, F., ve Klingman D., (1977). The Alternating Basis Algorithm for Assignment Problem. *Mathematical Programming*, **13**, 1-13.
- Bazaraa, M., Jarvis, J., ve Sherali, H., (1990). *Linear Programming and Network Flows*, 2nd Ed., John Wiley & Sons, New York, USA.
- Gillett, B. E., (1976). *Introduction to Operations Research: A Computer-Oriented Algorithmic Approach*, McGraw-Hill, New York, USA.
- Glover, F., ve Klingman, D., (1986). Threshold Assignment Algorithm. *Mathematical Programming Study*, **26**, 12-37.
- Hung, L., (1983). A Polynomial Simplex Method for the Assignment Problem. *Operations Research*, **31**, 595-600.
- Kuhn, H. W., (1955). The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, **2**, 1-2.
- Lotfi, V., (1989). A Labeling Algorithm to Solve the Assignment Problem. *Computers and Operations Research*, **16**, 397-408.
- Papadimitrou, C. H., ve Steiglitz, K., (1982). *Combinatorial Optimization, Algorithms and Complexity*. Printice-Hall, Englewood Cliffs, N.J, USA.