T. C.
ANADOLU ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DARBOĞAZ ATAMA MODELİNİN BİTİRME ÖDEVLERİ DAĞITIMI İŞLEMİNE UYGULANMASI

YÜKSEK LİSANS TEZİ

DANIŞMAN : DOÇ. DR. İMDAT KARA

MUZAFFER KAPANOĞLU

ESKİŞEHİR 1985

Ö Z E T

Atama Problemleri, endüstriyel ve sosyal kuruluşların sık sık karşılaştıkları başlıca problem alanlarından birisidir. Atama modellerinin belirli kıstaslara göre sınıflamasında yer alan *"Darboğaz Atama Modelleri"*, oldukça ilginç problemlere uygulanabilecek özel bir atama modeli olarak görülmektedir.

Bu çalışmada, atama problemleri genel olarak tanıtılıp, sınıflaması yapıldıktan sonra, darboğaz atama problemi ayrıntılarıyla ele alınmış, çözüm algoritmaları üzerinde durulmuş ve uygulanabilecek alanlara değinilmiştir. Yanısıra, bitirme ödevleri dağıtımı işlemleri öğrencilerin tercihlerini de gözönüne alan bir darboğaz atama problemi olarak ele alınıp, atama algoritması geliştirilmiş ve ilgili bilgisayar yazılımı ile algoritmanın uygulaması yapılmıştır.

A B S T R A C T

Assignment Problems can be considered as one of the main operational problem area in the industrial and social organizations. When the models constructed for these problems, classified with respect to certain criteria, *"The Bottleneck Assignment Models"* has the opportunity of application to many special assignment problems, that appears to be one of the interesting working area.

In this study, assignment problems are generally introduced and classified, then the Bottleneck Assignment Problem and its solution algorithms are studied in details. Meantime, the distribution of the B.Sc. thesis and projects, considering the students preferences and choises is taken as a Bottleneck Assignment Problem. A solution algorithm and a related computer program is developed and tested for this problem.

İÇİNDEKİLER

ÖZET i

ABSTRACT ii

ŞEKİLLER VE TABLOLAR LİSTESİ vi

GİRİŞ 1

B İ R İ N C İ B Ö L Ü M

ATAMA PROBLEMLERİNİN GENEL YAPISI

I.1 ATAMA PROBLEMLERİ 4

I.2 ATAMA PROBLEMİNİN MODELLENMESİ 7

I.3 ATAMA MODELLERİNİN SINIFLANDIRILMASI 11

I.3.1 Atama Modellerinin Kısıtlarına Göre Sınıflandırılması 12

i) Ağırlıklı Atama Modeli 13

ii) Üst Sınırlı veya Kapasite Kısıtlı Atama Modeli 14

iii) Klasik (Geleneksel) Atama Modeli 15

I.3.2 Atama Modellerinin Amaç Fonksiyonlarına Göre Sınıflandırılması 16

i) Doğrusal Atama Modeli 16

ii) Kareli Atama Modeli 17

iii) Darboğaz Atama Modeli 19

I.4 ATAMA MODELLERİNİN İLİŞKİLİ OLDUĞU BAZI MODELLER 21

İ K İ N C İ   B Ö L Ü M

DARBOĞAZ ATAMA PROBLEMİ

II.1 DARBOĞAZ ATAMA PROBLEMİNİN OLUŞUMU 22

II.2 DARBOĞAZ ATAMA PROBLEMİNİNİN MODELLENMESİ 24

II.3 DARBOĞAZ ATAMA MODELİNİN ÇÖZÜM YÖNTEMLERİ 28

II.3.1 Asal (Primal) Algoritma 30

II.3.2 Eşik (Threshold) Algoritması 31

II.4 ÖZEL DARBOĞAZ ATAMA MODELLERİ VE ÇÖZÜM YÖNTEMLERİ ... 34

II.4.1 Klasik Darboğaz Atama Modeli 34

i) Gross Algoritması 36

ii) Bhatia Algoritması 37

II.4.2 Kapasite Kısıtlı ve Ağırlıklı Darboğaz Atama Modelleri 39

II.2 DARBOĞAZ ATAMA MODELİNİN UYGULAMA ALANLARI 41

Ü Ç Ü N C Ü   B Ö L Ü M

BİTİRME ÖDEVLERİ DAĞITIMI İŞLEMİNİN DARBOĞAZ ATAMA MODELİYLE YAPILMASI

III.1 BİTİRME ÖDEVLERİNİ ATAMA PROBLEMİ 43

III.2 BİTİRME ÖDEVLERİNİ ÖĞRENCİLERE ATAMA PROBLEMİNE ÇÖZÜM YAKLAŞIMLARI 49

III.3 EN İYİ TERCİH SAYISININ BULUNMASI 59

III.4 GELİŞTİRİLEN ALGORİTMANIN UYGULANMASI 63

SONUÇ 71

YARARLANILAN KAYNAKLAR 75

EK - I 79

EK - II 80

EK - III 85

ŞEKİLLER ve TABLOLAR LİSTESİ

ŞEKİL I.1	Atama Modelinin Şebekesi	10
ŞEKİL II.1	Darboğaz Atama Modelinin Şebekesi	27
ŞEKİL III.1	Geliştirilen Algoritmanın Akış Şeması	58
ŞEKİL III.2 (a - b)	Tercih Sayısına Bağlı Olarak Öğrencilerin Tercih Etmedikleri Ödevlerden Birinde Olma Olasılığının Değişimi	61
ŞEKİL III.3	Tek Tercih İçin 'Ödev Sayısı/Öğrenci Sayısı' Göstergesine Bağlı Olarak Olasılığın Değişimi	62
ŞEKİL III.4	'Ödev Sayısı/Öğrenci Sayısı' Oranına Bağlı Olarak Eniyi Tercih Sayısının Değişimi	62
TABLO III.1	'Bitirme Ödevi Sayısı/Öğrenci Sayısı' Oranına Bağlı Olarak Önerilen Başlangıç Tercih Sayıları	68

G İ R İ Ş

İkinci Dünya Savaşını izleyen yıllarda en çok ilgi gören programlama tekniklerinden biri doğrusal programlama olmuştur. Kendisine bu ünü kazandıran, belki de en önemli etmenler tekniğin anlaşılmasının ve kullanılmasının kolay, çözüm yöntemlerinin nispeten hızlı olmasıdır. Bu yönleriyle doğrusal programlama, Yöneylem Araştırması ve Yönetimde Niceliksel Yöntemler kitaplarının en önemli kısımlarından birini oluştururken, iki alt başlığı genellikle hep yanında taşımıştır. Sözkonusu başlıklar, doğrusal programlamanın kullanım ve çözüm yöntemleri ile önemli bir parçası olup birbirlerinden yaygın kullanım alanlarına göre ismen ayrılmış olan, ancak temelde aynı modele sahip, ulaştırma (transportation) ve atama (assignment) modelleridir.

Atama problemi ve onun modelleri, taşıdığı özelliklerle zamanla büyük oranda doğrusal programlamanın sınırlarını aşarak tamsayılı programlama içinde yer almışlardır. Doğrusal olmayan amaç fonksiyonlarının nedenini oluşturduğu ve tamsayılı karar değişkenlerinin beslediği bu durum, atama modellerinin uygulama alanını genişletmiş, çözüm yaklaşımlarını geliştirmiştir.

Darboğaz atama modeli de doğrusal olmayan amaç fonksiyonuna sahip atama modellerinden biridir. Diğer atama modellerinden daha az ele alınıp incelenmiş olmakla birlikte bazı ilgi çekici uygulamalara sahiptir. İlgili tüm kaynaklarda ara alt başlıklar ya da makale düzeyinde yer alan Darboğaz Atama Probleminin ilişkin kavram, teknik, model ve çözüm yöntemlerini kendi içinde anlamlı bir bütün haline getirmek, yanısıra yapısı itibariyle Darboğaz Atama Problemi olarak karşılaşılan gerçek yaşamdan alınan bir probleme çözüm getirmek bu araştırmada amaç edinilmiştir.

Yukarıda belirtilen amaçlar doğrultusunda çalışma üç bölümden oluşmuştur. Birinci bölümde belirgin (deterministic) ve durağan (static) atama modelleri sınıflanarak tanıtılmış ve ilgili bazı modellere değinilmiştir.

İkinci bölümde Darboğaz Atama Problemi ve belirgin - durağan Darboğaz Atama Modelleri ayrıntılı olarak ele alınmış ve çözüm algoritmaları tanıtılmıştır.

Üniversitelerin fakülte veya dört yıllık bölümlerinde ders olarak önemli bir yeri olan bitirme ödevleri dersi, yapısı ve amacı yönüyle öğrencilerin belli konulara atanmasını öngörür. Bunun en geçerli biçimi, öğrencileri tercihlerine göre atamak olmakla birlikte, sözkonusu tercihlerin belli konularda yığılmalarını uygun biçimde aşmayı gerektirmektedir.

Çalışmanın üçüncü bölümünde, öğrencilerin bitirme ödevlerine atanmaları amacıyla darboğaz atama modeline dayalı bir algoritma geliştirilmiş, ilgili programlar yazılmış ve uygulamasına yer verilmiştir.

B İ R İ N C İ B Ö L Ü M

ATAMA PROBLEMLERİNİN GENEL YAPISI

I.1 ATAMA PROBLEMLERİ

En genel anlamda, bölünemez birimlerin oluşturduğu sonlu bir kümenin belirlenen kısıtlar altında, tamamının veya bir kısminin yine sonlu bir kümenin öğeleriyle eşlendirilmesine ilişkin problemler atama problemi olarak isimlendirilir. Çoğunlukla karşılaşılan biçimleriyle bu kümelerden biri işgören, araç, makina, ekip (mürettebat), satıcı vb. olabilen öğelerden, diğeri ise sırasıyla iş, yol, yerleşim yeri, sefer, bölge vb. öğelerden oluşabilir.

Atamaya esas alınan kümelerarası birebir eşleme istenebileceği gibi, ilgili kümelerin belirli alt kümelerinin eşlemesi de sözkonusu olabilir. Sözgelimi, bir montaj hattında, işlerin işlem noktalarında gruplandırılması ve gruplandırılan işleri yapacak yine birden fazla işgörenin bulunması halinde, karşılaşılan atama probleminde, alt kümeler gözönüne alınmaktadır.

Önceleri ortaya çıktığı şekliyle problem, işgörenlerin işlere, o işlerde umulan başarılarını veya en yalın haliyla, başarıyı gözetmeksizin, işgören atanmış iş sayısını enbüyüklemek olarak tanımlanmıştır [1]. Problem, atama kümelerinin öğelerindeki değişimlere dayalı olarak atama sonucu istenen durumu, farklı biçimlerde ölçümleme gereksinimleri doğrultusunda, değişik biçimlerde ortaya çıkabilmektedir.Her tür değişiklik problem parametrelerinin yanısıra ve bazen onlara da bağlı olarak problemin matematiksel gösterimine doğrudan yansımıştır.

Problemin karar değişkenleri, i.alt kümeden j.alt kümeye atanan öğelerin sayısını gösteren ve küme öğelerinin bölünemezliği sebebiyle negatif olmayan tamsayı değişkenlerdir. Üzerinde belirli bir işlev tanımlanan atama kümelerinin sonlu sayıda birimlere sahip olmalarına bağlı olarak, karar değişkenleri daima sonlu değer almak durumundadırlar.

Parametreler, kümeler arası ilişkileri yansıtan kısıtlardaki sayısal değerlere ek olarak, her bir karar değişkeninin aldığı değere göre çözüme sağlayacağı katkıyı amaç fonksiyonunda gösteren değerlerdir. Bu katkı parametreleri her bir alt küme için eşölçümlü ve doğal olarak eşbirimli

---

[1] H.W.Kuhn,"The Hungarian Method for the Assignment Problem", *Nav.Res.Log.Quart.*, Vol. 2, No. 1, 1955, s. 83-97, G.Hadley, *Linear Programming*, Addison-Wesley Publishing Inc., New York, 1963, s. 367-368.

olmalıdır. Böyle olamayan durumlarda problemi bu şekle dönüştürmek üzere ayrık alt problemler haline getirmek sözkonusu olabilir. Örneğin, aynı işyerinde farklı işler yapılıyor ise tüm bu işleri kapsayacak biçimde bir atama problemi tanımlanabilir. Bu durumda, işgörenlerin işteki etkinlikleri ortak bir ölçü (birim zamandaki eş üretim miktarı) ve ortak bir birim (parça) ile ölçülebilir olmalıdır. Örneği biraz açmak gerekirse, herhangi bir atölyede A işgöreni tornada yapılan belli bir işi C_{AT} zamanında ve frezedeki bir işi C_{AF} zamanında yapıyor ise bu parametreleri aynı problemde kullanmak yanlış sonuçlar doğurabilir. Bunu önlemek için problem, *"tornaya atama"* ve *"frezeye atama"* altproblemlerine dönüştürebilir. Ancak bu durumda elde edilecek eniyi çözümler toplamı atölye düzeyinde bir eniyi çözüme karşı gelmeyebilir. Parametrelerden birinin diğeri cinsinden yazılması veya kullanımı ile daha kolay ve aynı zamanda gerçeklere ters düşmeyen yeni bir etkinlik ölçüsü seçmek yoluna da gidilebilir. Bu son iki durum atölye düzeyinde eniyiyi bulabilmektedir.

Parametrelerle ilgili diğer önemli bir noktada problemin çözümünün, uygulanılması düşünülen zaman sürecince katkı parametreleri veya etkinlik ölçümlerinin durağan (static) ve belirgin (deterministic) göstergeler olduğudur.

Problemin temel kısıtları, atamanın yapıldığı her bir alt kümenin sahip olduğu öğeden fazlasının atama işlemi içine katılmasını önlemek ve gene her bir alt kümeye gereken miktarda atama yapılmasını sağlamaya yöneliktir. Bunun dışında problemin oluştuğu sistemin yapısından kaynaklanan özel kısıtlar bulunabilir. Kısıtlar, problemin analitik veya yordamlama çözüm yaklaşımlarından hangisi ile daha etkin çözülebileceği ve her bir yaklaşımın kapsadığı bir dizi çözüm tekniklerinden hangisinin daha uygun olduğu soruları açısından son derece önemlidir.

Problemin uygun çözümlerinin bir değerlendirmesi olan ve karar değişkenlerinin ilgili ağırlıklarını veren etkinlik ölçülerini veya katkı parametrelerini bu değişkenlerle birlikte ele alıp çözümü istenen yön ve doğrultuya sevkeden amaç fonksiyonları da kısıtlarda bahsedilen çözümsel düşünceleri, yerine göre de kaygıları çağrıştıracak konumdadır. Parametreler, amaç fonksiyonunun yapısını etkileyebileceği gibi, özellikle istenen bir amaç fonksiyonu var ise, kullanılacak parametrelerin bu özellikleri taşımalarına dikkat etmek gerekir.

Yukarıdaki açıklamalara bağlı olarak izleyen kesimde atama problemenin genel modeli verilecektir.

I.2 ATAMA PROBLEMİNİN MODELLENMESİ

Genellikle biri etken diğeri ise edilgen yapı gösteren ve aralarında eşleme sözkonusu olan kümelerin biri N ve diğeri M ile gösterilsin. Altkümelerinin öğelerinin özelliklerinden kaynaklanan bu durumu belirtmek üzere N ile gösterilen kümeye *"atama kümesi"* ve M ile gösterilene *"atanma kümesi"* denilecektir. Sözgelimi; *"Satış Elemanlarının Bölgelere Atanması"* ile *"Bölgelerin Satış Elemanlarına Atanması"* nın arasındaki fark, kümelerin etken ve edilgen olma özelliklerini göstermektedir. Bu nedenle birinci durumda satış elemanları atama kümesi olarak ve bölgeler atanma kümesi olarak ve ikinci durumda ise tersi ele alınacaktır.

Atama kümesi N, n tane karşılıklı ayrık ve bütüne tamamlayan alt kümeden oluşmaktadır. Bu alt kümelerin öğe sayısı, i alt küme indisi olmak üzere a_i ile gösterilsin. Atanma kümesi M'de m tane ayrık ve bütüne tamamlayan alt kümeden oluşmaktadır. b_j, j alt küme indisi olmak üzere alt kümelerin öğe sayısı olsun.

Bu kümeler arasındaki mümkün atamalar kümesi S ile gösterilsin. Sözkonusu her bir atamanın etkinliği (maliyet, kâr, verimlilik vb.), $s \in S$ olmak üreze C(s) olsun. Bu haliyle atama problemi, eniyi C(s)'yi veren s atamasının bulunmasıdır.

X_{ij}, i'inci alt kümeden j'inci alt kümeye atanan öğe sayısını gösteren karar değişkeni ve C_{ij} bu karar değişkenlerinin amaç fonksiyonundaki etkisini gösteren katkı parametresi olmak üzere tanımlansın.

Bir atama alt kümesinden tüm atanma alt kümelerine atanacak miktar, ilgili atama kümesinin öğe sayısını aşamaz. Bu yüzden modelde ilgili atamayı veren karar değişkenlerinin toplamları ilgili alt küme sayılarıyla (a_i) üstten sınırlanmıştır. Her bir atanma kümesinin gereksiniminin karşılanması sözkonusu olduğundan bununla ilgili karar değişkenlerinin toplamı ilgili alt küme sayılarıyla (b_j) alttan sınırlanmıştır.

Böylece genel atama modeli aşağıdaki gibi yazılır:

$$\sum_{j=1}^{m} X_{ij} \leqslant a_i \quad , \qquad i = 1, 2, \ldots , n \qquad ..(1)$$

$$\sum_{i=1}^{n} X_{ij} \geqslant b_j \quad , \qquad j = 1, 2, \ldots , m \qquad ..(2)$$

$$X_{ij} \geqslant 0 \text{ ve tamsayı} \quad , \qquad i = 1,2,\ldots,n$$
$$j = 1,2,\ldots,m \qquad ..(3)$$

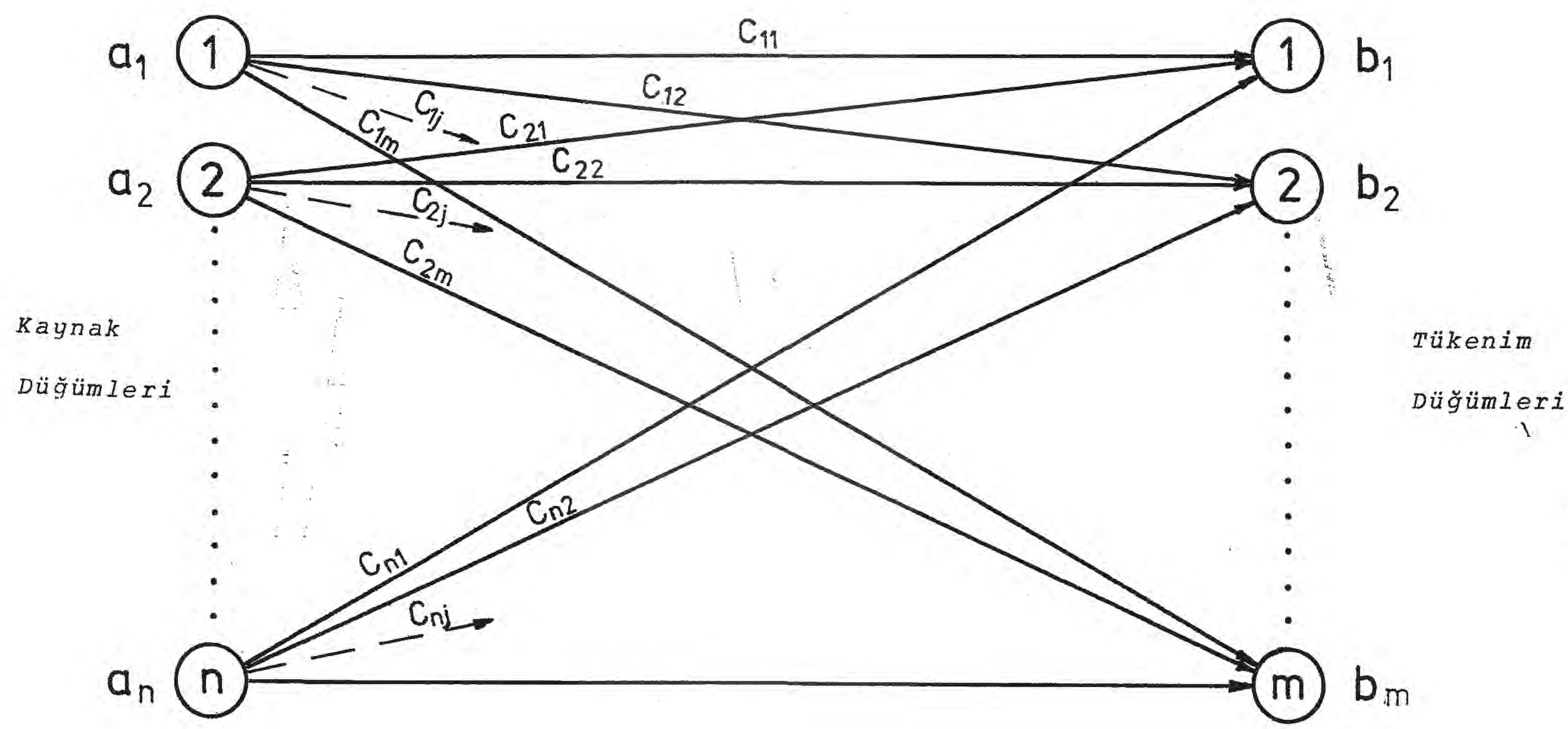
kısıtları altında,

$$\text{Eniyi } Z = f(X_{ij} \, , \, C_{ij}) \qquad ..(4)$$

Model, bu haliyle bir şebeke olarak da gösterilebilir. Şebekede kaynak (source) düğümlerine atama kümesi N'in alt kümeleri, tükenim (sink) düğümlerine de atama kümesi M'in alt kümeleri karşı gelmektedir. Şekil I.1'de genel bir atama modelinin şebekesi verilmiştir.

Şekil I.1'deki kesikli oklar noktalarla gösterilen düğümlere karşı gelen atamaları temsil etmektedir. Şekilden şebekenin iki parçalı (bipartite) olduğu ve a_i'lerin kaynak kapasitelerini, b_j'lerin tüketilecek miktarları ve C_{ij}'lerin her iki düğüm çifti arasındaki akışın katkısını verdiği görülmektedir. Karar değişkeni X_{ij}'ler i düğümünden j düğümüne olan akış miktarını gösterirler.

Atama modelleri yapıları gereği, amaç fonksiyonunun da yapısına bağlı olarak, birleşimsel eniyileme (combinatorial



ŞEKİL I.1 : *Atama Modelinin Şebekesi*

optimization) başlığı altında doğrusal programlama doğrusal olmayan programlama, doğrusal olmayan programlama, tamsayılı programlama, şebeke teknikleri ve ağ kuramının kuramsal tabanını oluşturduğu ve ulaştırma, tahsis, çizelgeleme, fabrika yerleşimi, tesis planlama, montaj veya üretim hattı problemlerinin çözümlenmesinde ilginç uygulama alanlarına sahip modeller olarak görünmektedir. Böylece atama problemlerinin tanımlanması, modellenmesi, sınıflandırılması ve çözüm tekniklerinin geliştirilmesinde bir dizi kıstas ve buradan bir dizi yaklaşım biçimi ortaya çıkabilmektedir. İzleyen kesimlerde bu noktalar ele alınıp incelenecektir.

I.3 ATAMA MODELLERİNİN SINIFLANDIRILMASI

Uygulama alanlarının gereksinimleri ve özel problemlerin yapıları sonucu, yukarıda genel yapısı açıklanan modelin kısıtlar kümesinde eklenti ve değişiklikler yapılması gerekebilirken, amaç fonksiyonun yapısı da aynı nedenlere bağlı olarak farklılık gösterebilir. O halde, özel durumlar için geliştirilen ve benzerleri için genelleştirilen modelleri özel başlıklar altında incelemekte fayda vardır.

Atama modelleri sınıflandırılırken, matematiksel modellerin sınıflandırılmasında gözönüne alınan, doğrusal, doğrusal olmayan vb. kıstaslar kullanılabileceği gibi, kısıtlar kümesinin ve amaç fonksiyonunun yapısına bağlı olarak da sınıflandırma yapılabilir.

Atama modelleri, kaynaklarda rastlanabildiği kadarıyla, kısıtlar ve amaç fonksiyonlarının çeşitli birleşimlerini kullanarak hem problem hem de model özelliklerine bağlı olarak adlandırılırlar. Bunlar, kısa isim kullanma eğiliminin da etkisiyle, ya amaç fonksiyonunun yapısını ya da kısıt kümelerinin yapısını yansıtmaktadır. Böylece, sözü edilen atama modelinin yapısını ilgili eser veya makaleyi incelemeden anlamak mümkün olmamaktadır. Benzeri sıkıntıları ve isim karmaşasını önleyecek, gelecekte türetilecek özel atama modellerine sınıflamada yer ve türetmede olanak sağlayacak, böylece konuyla ilgili akademik çalışmalar ve tartışmalarla ilgili uygulamalara açıklık kazandıracak bir sınıflamanın yapılması yerinde olacaktır.

İzleyen paragraflarda açıklanan sınıflamada yalnız durağan, belirgin yapı taşıyan atama modelleri ele alınmıştır. Parametreleri zaman içerisinde değişkenlik gösteren dinamik yapıdaki ya da parametreleri rassal yapıdaki modellerle, çok amaçlı tipler, farklı çözüm yöntem ve yaklaşımları gerektirmeleri nedeniyle,kapsam dışı bırakılmışlardır.

Sözkonusu nedenle atama modelleri, izleyen kesimde kısıtların yapılarına ve amaç fonksiyonlarının durumlarına göre iki kıstasa göre sınıflandırılmaktadır.

I.3.1 Atama Modellerinin Kısıtlarına Göre Sınıflandırılması

Önceki kesimde verilen atama probleminin genel modelinde, kısıtlar kümesini oluşturan (1) ve (2) nolu eşitsizlikler, kümeler arası atamayı salt kümelerin öğe sayılarını gözönüne alarak gerçekleştirecek yapıdadır. Oysa bir çok problemde atanma kümelerinin gereksinimlerinin karşılanması yanında, atamanın bazı özellikler taşıması beklenir. Aşağıda bu özelliklerden en yaygın olanları ve karşı gelen kısıt yazılımları incelenecektir.

i) Ağırlıklı Atama Modeli

Bu modellerde, problemden kaynaklanan bir ağırlık faktörünün tanımlanması sözkonusudur. Bu faktör, sağlanmak istenen kümeler arası ilişkinin yapısına bağlı olarak, i.alt kümeden j. alt kümeye yapılacak atamanın belli özellikler taşımasını sağlamak üzere r_{ij}, i. alt kümeden j. alt kümeye yapılacak bir birim atamanın ağırlığı ve p_j ve t_j ağırlık sınırları olup, her alt kümede benzer gereksinim olmayabileceğini de gözönüne olarak, bazı j'ler için,

$$t_j \leqslant \sum_{i=1}^{n} r_{ij} X_{ij} \leqslant p_j$$

biçiminde yazılır [2]. Sözgelimi i kümesinin öğeleri işgörenler ve j kümesininkiler de işler olduğunda j işinin en fazla p_j zamanda bitmesini sağlamak üzere r_{ij}, i.işgörenin j.iş için kullandığı süre olabilir.

---

[2] G.T.Ross, A.A.Zoltners, "A Branch and Bound Algorithm for the Generalized Assignment Problem" *Mathematical Programming*, No.8, 1975, s. 92.

Benzer biçimde t_j, j.işin birim zamandaki en az üretim miktarı iken r_{ij}, i. işgörenin j.işteki üretim miktarı şeklinde tanımlanabilir.

Yukarıdaki modelin üç veya dört indisli karar değişkenleri de kullanılarak benzer kısıt yazılımları da kaynaklarda yer almaktadır [3].

ii) Üst Sınırlı veya Kapasite Kısıtlı Atama Modeli

Bu modellerde i. alt kümeden j. altkümeye atanabilecek öğe sayısı sınırlanmış olup d_{ij} ile gösterilir ve genellikle,

$$0 \leqslant X_{ij} \leqslant d_{ij} \quad , \text{ her i ve j için}$$

biçimindeki kısıt veya kısıtlar modele eklenir [4]. Bazen bu kısıtlar bazı değişkenlerin toplamı olarak bir kısım ive j' ler için

$$\Sigma\, x_{ij} \leqslant K \qquad ..(5)$$

şeklinde de yazılabilmektedir [5].

Sözgelimi küme öğeleri, fabrikalar ve bölge depoları olduğunda i. fabrikadan j. depoya taşınabilecek mal miktarı

---

[3] G.T.Ross, AA.Zoltners, "Weighted Assignment Models and Their Applications" Management Science, Vol. 25, No.7, 1979, s. 685-690

[4] G.Hadley, A.g.k., s. 395.

[5] B.G.Dantzig *Linear Programming and Extensions,* The RAND Corporation Princeton, N.J., 1963, s. 382.

ulaşım olanakları çerçevesinde belli bir değeri aşmamak biçiminde modele yansıtılabilir. Küme öğeleri ulaşım seçenekleri ve ülkelerdeki satış merkezleri olduğunda ise belli bir ülkeye tüm ulaşım seçenekleri ile gönderilecek mal miktarının gümrük ve kotalarla belirlenmiş bir değeri aşmaması gerekebilir.

iii) Klasik (Geleneksel) Atama Modeli

Genel atama modelinin kısıtlar kümesinde (1) ve (2) nolu eşitsizlikleri her i ve j için $a_i = 1$ ve $b_j = 1$ olduğunda, modelin yazılımı,

$$\sum_{j M} X_{ij} \leq 1 \quad , \text{ her } i \text{ için}$$

$$\sum_{i N} X_{ij} \geq 1 \quad , \text{ her } j \text{ için}$$

biçiminde gerçekleşir. Model, bu haliyle (1) ve (2) nolu eşitsizlikler ile verilen genel kısıtlar kümesinin en basit şekli olup kaynaklarda oldukça ilgi çekmiş ve izleyen kesimde doğrusal alt başlığında anılacak olan amaç fonksiyonu ile birlikte, konuyla ilgili temel nitelikli ders kitaplarında bile"*atama modeli*"veya"*atama problemi*"özel başlığında ele alınmıştır. Genel atama modelinde verilen kısıt kümesinin de yukarıdaki kısıt kümeleri haline getirilebileceği ve bu kümelerin karar değişkenlerine verdiği"*0-1 değişken*" niteliğinin getirdiği bir dizi işlem kolaylıkları gözönüne alınırsa, klasik atama modeline gösterilen ilginin altında yatan nedenler daha kolay anlaşılabilir. Ayrıca bu modeli gereksinen problemlerin belirlenmesindeki kolaylıkların da bu ilgiyi beslediği söylenebilir. Bu tür kısıtlara sahip modellere amaç fonksiyonuna göre sınıflandırma altbaşlığında tekrar değinilecektir.

Genel atama modelinin (1) ve (2) nolu eşitsizliklerinde a_i ve b_j değerleri tamsayı olduğu sürece tüm uygun çözümleri tamsayı olacaktır [6]. Ancak ağırlıklı atama modeli, üst sınırlı modelin (5) nolu eşitsizliği kullanan türleri eniyi çözümde tamsayı olma özelliğini her zaman sağlamazlar [7]. Bu nedenle bu modellerin çözüm teknikleri genellikle diğerlerinden daha çok işlem gerektirirler ve daha karmaşıktırlar.

I.3.2 Atama Modellerinin Amaç Fonksiyonuna Göre Sınıflandırılması

Önceki kesimde verilen genel atama modelinin amaç fonksiyonu, etkinlik ölçüleri ile karar değişkenleri arasındaki ilişkinin kapalı bir gösterimi olarak verilmiştir. Böylece aşağıda tanımlanacak olan atama modellerinin [illegible] kapsamı içine alınmaktadır. Aşağıdaki sınıflama içinde modeller açık yazılımları ile verilecektir.

i) Doğrusal Atama Modeli

Bazı atama problemlerinde karar değişkenlerinin değerlerindeki değişimler amaç fonksiyonunda sabit oranda deği-

---

[6] S.R.Garfinkel, G.L. Nemhauser, *Integer Programming*, John-Wiley Sons, Inc., New York, 1972, s. 67-70.

[7] B.G.Dantzig, A.g.k., s. 380-383.

şimlere sebep olur ve bu oran birim değişimler için c_{ij}'ye eşittir. Bu problemlerde toplam katkı, her değişkenin katkıları toplamına eşittir ve modelde amaç fonksiyonu,

$$\text{Eniyi } Z = \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij}$$

şeklinde oluşur. Sözgelimi k. tür işçinin p. işteki üretim miktarı c_{kp} ise k. tür işçiden p. tür işe atananların sayısı x_{kp} ve k. işin toplam üretimdeki payı $\{c_{kp} x_{kp}\}$ olacaktır.

Atama modellerinin doğrusal programlamanın bir uzantısı olması bakımından ilk ortaya atılıp tanımlanan türü doğrusal atama modelleridir [8]. Bu model Kantarovich, Ford, Fulkerson, Koopmans ve Hitchcock tarafından bağımsız olarak incelenmiş ve Hitchcock genel simplex algoritmasına çok benzeyen ilk çözüm yöntemini önermiştir [9]. Klasik doğrusal atama modeli için ilk çözüm yöntemi ise Kuhn tarafından geliştirlmiştir [10].

ii) Kareli Atama Modeli

Bu tür atama modellerinde i'den j'ye bir öğeyi atamanın katkısı veya maliyeti, başka bir öğenin dek'dan l'ye atanmasına bağlı olarak oluşur. Şöyle ki i, k $\in$ N ve j, l $\in$ M

---

[8] H.W.Kuhn, "The Hungarian Method for the Assignment Problem", *Nav.Res.Log.Q.*, Vol. 2, No. 1,2, 1955, s. 83.

[9] L.R.Ford, D.R. Fulkerson, *Flows in Networks*, The RAND Corporation, Princeton N.J., 1974, s. 95.

[10] Kuhn, "The Hungarian ....", s. 83-96.

olsun. i'nin j'ye ve k'nın l'ye atanması halinde (j,l) atanma çiftinin etkinliği d_{jl} ve (i,k) atama çiftinin etkinliği c_{ik} ile gösterilsin. Karar değişkenleri, i'nin j'ye atanması veya atanmamasıyla ilgili 0-1 niteliğinde olmak üzere amaç fonksiyonu,

$$\text{Eniyi } Z = \sum_{\substack{i,k=1 \\ i \neq k}}^{n} \sum_{\substack{j,l=1 \\ j \neq l}}^{m} c_{ij} d_{jl} x_{ij} x_{kl}$$

biçiminde oluşturulur[11].

Modelde N ve M kümelerinin öğe sayıları eşit ve p iken, p^2 karar değişkeni çifti ve p^4 kadar etkinlik katsayısı ortaya çıkmaktadır. Sözgelimi, N kümesi fabrikaları ve M kümesi yerleşim yerlerini içersin. Bu fabrikalar arasında bir taşıma sözkonusu ise i. fabrikanın j. yerleşim yerine atanmasıile ortaya çıkacak etkinlik diğer fabrikaların da nerelere atandığına doğrudan bağlı olacaktır. Uygulama alanı itibariyle kareli atama modeli bu ve benzeri tesis planlaması problemlerinde sıkça başvurulan bir modeldir [12].

Kareli atama modeli, analitik ve yordamsal çözüm tekniklerine sahip olup, amaç fonksiyonundaki terim fazlalığı nedeniyle geniş çaplı olduklarında yordamlama teknikleri ile

---

[11] E.L. Lawler, "The Quadratic Assignment Problem", *Mgmt. Sci.*, Vol. 9, No.4, 1963, s. 586.

[12] R.L.Francis, J.A. White, *Facility Layout and Location: An Analytical Approach*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1974, s. 328-377.

iyice çözüm (suboptimization) yoluna gidilmektedir. Analitik çözüm tekniklerinde bilgisayar zamanı p'ye bağlı olarak çok büyük oranda artmaktadır [13].

Bu model ilk defa 1957'de Koopmans ve Beckmann tarafından geliştirilmiş, daha sonra 1962'de Gilmore [14], 1963' de Lawler [15] ile eniyileme ve yordamlama çözüm teknikleri geliştirilmeye başlanmış olup, çalışmalar günümüzde de sürmektedir.

iii) Darboğaz Atama Modeli

Bazı atama problemlerinde çözümün etkinliği atamanın bütününe bağlı olmayabilir. Etkinlik, atama ile elde edilen katkıların enküçüğüne veya enbüyüğüne göre oluşabilir. Daha genel olarak, modelin amaç fonksiyonu, maliyet yapılı atama problemlerinde,

$$\text{Enk } Z = \text{Enb } \{c_{ij} | x_{ij} > 0\}$$
$$i = 1, \ldots, n$$
$$j = 1, \ldots, m$$

şeklinde, gelir yapılı atama problemlerinde ise,

---

[13] R.Burkard, U.Derigs, *Assignment and Matching Problems: Solution Methods with Fortran-Programs,* Springer - Verlag, Berlin, 1980, s. 104.

[14] P.C.Gilmore, "Optimal and Suboptimal Algorithms for the Quadratic Assignment Problem", *I.Soc.Indust. Appl. Math.,* Vol. 10, No. 2, 1962, s. 305-313.

[15] Lawler "The Quadratic ......" s. [illegible]

$$\text{Enb } Z = \text{Enk } \{C_{ij} | X_{ij} > 0\}$$

$$i = 1,\ldots,n$$
$$j = 1,\ldots,m$$

biçiminde ifade edilebilir [16].

Bu tip amaç fonksiyonları, katkı parametrelerine bağlı olarak amaçlanan yönde bir eniyilemeyi gerektirirken aynı zamanda öğeler arasında *"katkı yakınlığı"* gerçekleştirirler ve bu özellikleriyle bazı problemlerde kullanılmaları kaçınılmazdır. Sözgelimi satış elemanlarının bölgelere atanmasıyla ilgili katkı parametreleri herbirinin beklenen satış miktarları ise, bu problemin katkı parametrelerinin toplamının eniyilenmesi biçiminde çözülmesi gerektiği açıktır. Oysa katkı parametrelerinin öğrencilerin dersliklerdeki uygunluk ve rahatını gösterdiği, sınıfların dersliklere atanması probleminde, sınıflar arasında bir öncelik veya tercih sözkonusu edilemeyeceğine göre, atamalarda elde edilen çözümlerin atandığı derslikte en az uygun ve rahat olan sınıfın durumuna göre değerlendirilmesi gerekir.

Darboğaz atama modeli olarak isimlendirilen yukarıdaki model, ilgili problem alanlarıyla birlikte, izleyen bölümde ayrıntılı bir biçimde ele alınacaktır.

---

[16] R.S. Garfinkel, "An Improved Algorithm for the Bottleneck Assignment Problem" *Working Paper Series*, No.7003, College of Business Administration, University of Roshester, New York, 1970, s. 1747-1753.

I.4 ATAMA MODELİNİN İLİŞKİLİ OLDUĞU BAZI MODELLER

Kaynaklarda, yapısal olarak atama modellerine çok benzemekle birlikte, özel isimler altında incelenen modeller de vardır. Ağ kuramı (graph theory) ve tamsayılı programlamanın kendi kavram ve teknikleriyle ele aldığı eşleme (matching) problemi bunlardan biridir [17]. Model, 0-1 niteliğinde karar değişkenlerine sahip klasik doğrusal atama modellerinin özel bir türüdür. Bütün değişkenlerin katkı parametreleri bir birimdir. Kaplama (covering) veya küme kaplama (set covering) problemleri de amaç fonksiyonundaki katkıları bir birim olan 0-1 nitelikli karar değişkenlerine sahip atamasal modelleri kullanırlar [18]. Gezgin satıcı problemi olarak anılan özel 0-1 tamsayılı modeller de klasik doğrusal atama modelini aynen kullanmakla birlikte, esasen bir tur belirleme problemi olduğundan alt turların oluşmasını önlemek üzere özel kısıtları içerirler [19].

Bu bölümde atama problemi kavramsal olarak tanımlanmış ve genel karar modeli geliştirilmiştir. Daha sonra, yaygın atama modelleri belli kıstaslara göre sınıflanarak tanıtılmıştır. İzleyen bölümde DAP ayrıntılarıyla ele alınacaktır.

---

[17] D.R. Plane, Jr.C.McMillan, *Discrete Optimization:Integer Programming and Network Analysis for Management Decisions*, Prentice-Hall, Inc., New Jersey, 1971, s. 30.

[18] Plane, McMillan, A.g.k., s. 31-33.

[19] Garfinkel, Newhauser, A.g.k., s. 354-356.

İ K İ N C İ   B Ö L Ü M

DARBOĞAZ ATAMA PROBLEMİ

Bu bölümün kapsamı içerisinde darboğaz atama problemi, modeli ve çözüm yöntemleri tanıtılacak ve özel kısıtların eklenmesi halinde model ve çözüm yöntemlerinde ortaya çıkan değişiklikler üzerinde durulacaktır. Darboğaz atama modellerinin her bir durumu ele alınarak tarihsel gelişiminden de söz edilecek ve bölümün sonunda uygulama alanlarına değinilecektir.

II.1 DARBOĞAZ ATAMA PROBLEMİNİN OLUŞUMU

Atama problemlerinin sahip oldukları özelliklere göre farklı amaç fonksiyonlarıyla ele alınmaları gerekliliği ve bu çerçeve içnde Darboğaz Atama Probleminin (DAP) konumunun ne olabileceği üzerine ilk bölümde bazı açıklamalara yer verilmiştir. Burada öncelikle problemi yaratan koşullar üzerinde durulacaktır.

Problem ilk defa *"Üretim Hattı Atama Problemi"* adıyla tanıtılmıştır [20]. Bu tarihten çok önce üzerinde çalışmaya başlanmış olan klasik doğrusal atama problemlerinin üretim hattı bulunan atölyeler için yeterli olmayabileceği düşüncesi darboğaz atama problemini gündeme getirmiştir. Herhangi bir üretim hattının çıktısı o hat üzerindeki en düşük üretim hızına sahip işlem noktasının çıktısını aşamaz. Bu durum hat üzerinde darboğaz yaratan işlem noktalarının daha verimli işçi veya işçilerle beslenmesini gerektirmektedir. Böylece hattın çıktısı olabildiğince arttırılabilir. Atamanın, hattın çıktısını enbüyüklemesi için sözkonusu beslemeyi yapmak üzere darboğaz atama problemi olarak ele alınması öngörülmektedir.

Bu öngörünün altında yatan, genelde darboğaz programlamada (bottleneck programming), özelde DAP'de problemin eniyi çözümünün belirlenen koşullar altında darboğaz oluşturan parametreye veya işleme bağlı olmasıdır. Sözgelimi en kısa süreli dağıtım probleminde de araçların belirlenen noktalara en kısa zamanda varış geri dönmesi istendiğinden, amaç fonksiyonunun çözümde alacağı değer en geç dönen aracın dönme süresine bağlı olacak ve bu süre darboğaz zamanı oluşturacaktır.

Bu ve benzeri örneklerle ortaya çıkan şudur ki, atamanın darboğaz yaratan öğesi veya öğeleri, atamanın tümünün etkinliğini belirlemektedir. Böyle durumlarda amaç fonksiyo-

---

[20] J.Edmonds, D.R.Fulkerson, "Bottle Neck Extrema", *Journal of Combinatorial Theory*, Vol. 8, 1970, s. 299-306.

nunun darboğaz biçiminde seçilmesi, ihtiyacı doğuran durumlara bağlıdır. Yani problemlerin DAP olarak tanımlanması ihtiyaçlar sonucu ortaya çıkmaktadır. Oysa ki bazı problemlerde atamanın bütünsel etkinliğinde darboğaz öğeye bağlanmak gerçekten bir ihtiyaç değildir. Ancak problemin DAP olarak çözülmesi uygun gürülebilir. Sözgelimi, bir fabrikada işgörenlerin tüm atölyelerde çalışma isteklilikleri yani iş tatminleri belirlenmiş olsun. Böyle bir durumda, işgörenlerin atölyelere atanması iş tatmini kıstası altında yapılacak olduğunda, atamanın en az iş tatminini elde eden işgöreni gözönüne alınarak yapılması uygun olacak gibi görünmektedir.

Herhangi bir problemin bir DAP olarak ele alınmasının veya böyle bir yapıda olduğunun düşünülmesinin altında yatan nedenler ne olursa olsun DAP, ele alınan kümelerde atama çiftlerinden en kötü katkıya sahip olanın (veya olanların) eniyilenmesine yönelik problemleri belirtir. İzleyen kesimde DAP'nin genel modeli verilerek özel DAP ve modellerine geçilmektedir.

II.2 DARBOĞAZ ATAMA PROBLEMİNİN MODELLENMESİ

DAP'nin n tane atama kümesi ve m tane atanma kümesi olsun. Her bir atama kümesi, i = 1, 2, ... , n olmak üzere a_i, öğeye ve her bir atanma kümesi j = 1, 2, ... , m olmak üzere b_j öğeye sahiptir. Sözgelimi, bir üretim hattı ve hattın belirli işlemler için uğradığı iş istasyonları ele alınsın. İşçiler, hat üzerindeki iş istasyonları dizisinde benzer veya farklı, herbirini bir işçinin yapacağı gibi bölünmüş işlerden bazılarını yapabilir özelliktedirler. İş istasyonları, yapılarına göre, bir veya birden fazla işçiye gereksinim duymaktadır. n farklı nitelikte işçi olup, m iş istasyonu vardır. Herhangi bir iş istasyonunun işçi gereksinimi b_j birim olup, üretim için karşılanmak zorundadır. Benzer biçimde i niteliğine sahip işçi sayısı a_i olup, sınırlı sayıda ve kısa dönemde arttırılamaz olduğundan aşılmamalıdır. Kısaca, tüm iş istasyonlarının i nitelikli işçi gereksinimi en fazla a_i birim olmak zorundadır. Ayrıca her bir iş istasyonunun işçi gereksinimi b_j birim olup, o iş istasyonunda bunun daha altında bir sayıda işçiyle üretim yapılması sözkonusu değildir. İşçiler sadece atandıkları işte çalışacaklardır. İşgücünün iş istasyonları arasında kaydırılması veya aynı istasyonda iki ayrı işe atanması kabul edilmeyecektir.

i niteliğindeki işçilerin j istasyonundaki üretim kaybı c_{ij} ile gösterilmiş olsun. Bu durumda hattın üretimi, üretim kaybının büyüklüğüne ve dolayısıyla hat üzerinde üretim kaybının en fazla olduğu iş istasyonuna bağlı olacaktır. Üretimin daha fazla olması için atamanın üretim kayıplarından en büyüğünü gözönüne alarak yapılması ve gene buna göre iyileştirilmesi mümkündür. Çünkü iş istasyonlarından birindeki tıkanıklık veya gecikme sonraki istasyonları ve hatta hattın yapısına göre önceki istasyonları da durmaya zorlayacaktır. Böylece hattın üretimi gecikmenin en fazla olduğu iş istasyonunun çıktısına bağlı kalmaktadır. O halde model üretim kaybı en fazla olan iş istasyonunun kaybını azaltmak üzere tasarlanmalıdır. Bu koşulları sağlayan DAP genel modeli, kısıtlar kümesi ve parametre kavramları genel atama modeli ile aynı olup, aşağıdaki gibidir.

$$\sum_{j=1}^{m} X_{ij} \leqslant a_i \quad , \quad i = 1, 2, \ldots , n$$

$$\sum_{i=1}^{n} X_{ij} \geqslant b_j \quad , \quad j = 1, 2, \ldots , m$$

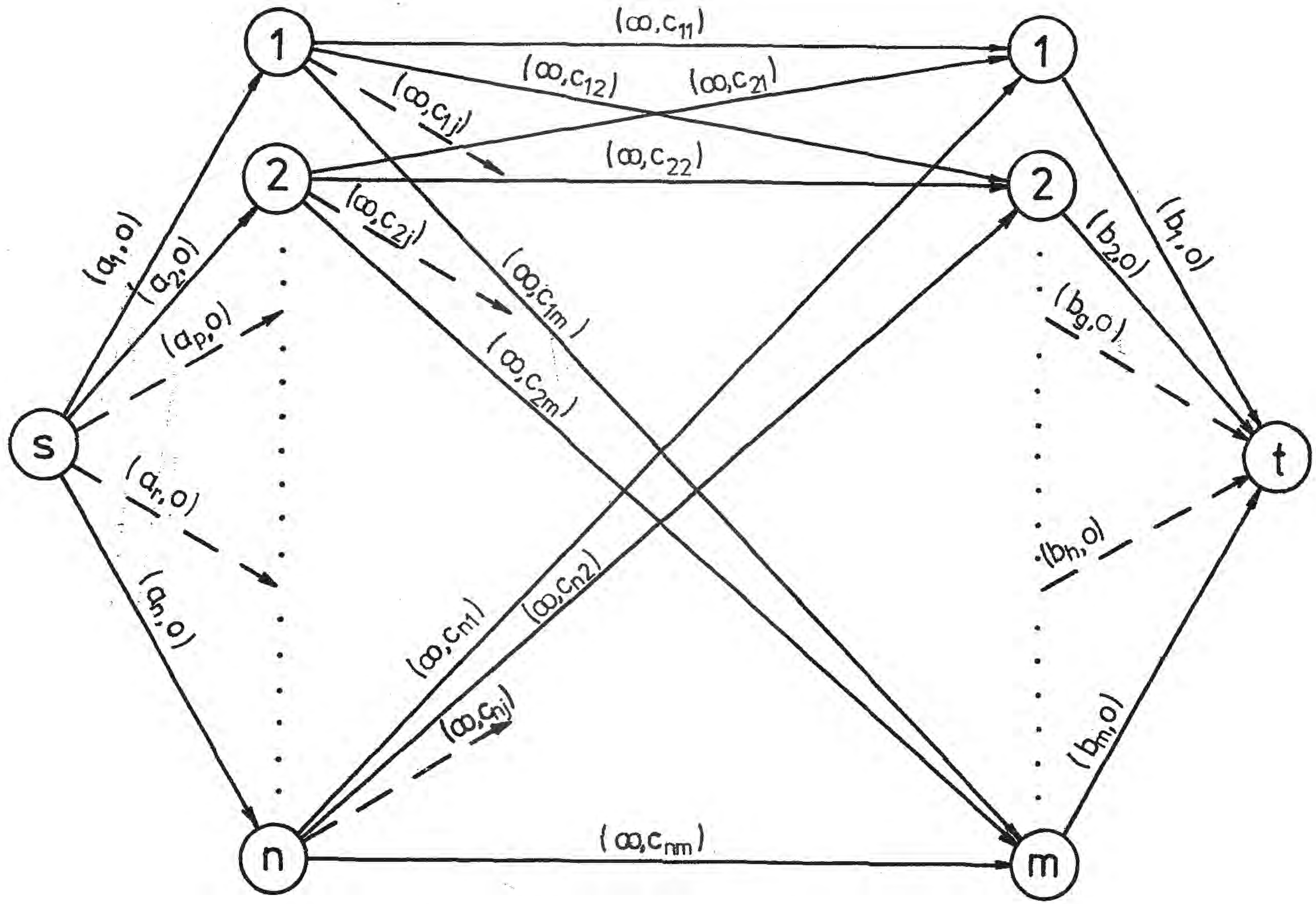
$$X_{ij} \geqslant 0 \text{ ve tamsayı} \quad , \quad \text{her } i \text{ ve } j \text{ için}$$

kısıtları altında

$$\text{EnkZ} = \underset{\substack{i=1,..,n \\ j=1,..,m}}{\text{Enb}} \{c_{ij} | X_{ij} > 0\}$$

DAP'nin yukarıda verilen modeli n atama düğümü sayısı ve m atanma düğümü sayısı olmak üzere Şekil II.1'deki gibi bir şebeke biçiminde gösterilebilir. Bu şebekede S atama kaynağı düğümü ve t atanma yeri düğümü olup, problem bu şekliyle S'den t'ye enbüyük maliyetli akışın enküçüklenmesidir. Şekilde gösterildiği gibi, S'den atama düğümlerine olan her bir okun akış kapasitesi, $i \in n$ olmak üzere a_i birim ve atanma düğümlerinden t atanma yeri düğümüne olan her bir okun kapasitesi $j \in m$ olmak üzere b_j birim olarak alınmaktadır. Herhangi bir $(i \to j)$ okundan bir birim akışın katkısı c_{ij} olup, bu ok üzerinden sağlanan akış miktarı X_{ij} birimdir.

S atama kaynağı düğümünden atama düğümlerine birim akışın katkısı (veya maliyeti) sıfır olup, her bir ok boyunca



ŞEKİL II.1 : Darboğaz Atama Modelinin Şebekesi akışın üst sınırı a_i birimdir. Atanma düğümlerinden t atanma yeri düğümüne de her birim akışın katkısı sıfır olup, her bir ok boyunca akışın alt sınırı b_j'dir. Sözkonusu (i,j) çiftleri arasında akışın herhangi bir üst sınırı yoktur ve sonsuz alınmıştır. S'den atama düğümlerine ve t'ye atanma düğümlerinden gelen toplam akışın w birim olduğu durumlar kümesi matematiksel modeldeki uygun çözüm uzayını vermektedir. Burada problem, üzerinden en az bir birim akışın geçtiği ve enbüyük c_{ij}'ye sahip $(i \rightarrow j)$ okundaki akışın enküçüklenmesidir.

DAP'nin verilen modeli maliyet yapılı olup, izleyen kesimdeki algoritmalar bu durum için verilmişlerdir. Katkı yapılı problemler için modelin düzenlenmesi basit işlemleri gerektirir.

II.3 DARBOĞAZ ATAMA MODELİNİN ÇÖZÜM YÖNTEMLERİ

Darboğaz Atama Modeli (DAM), önceki kesimde değinildiği gibi, iki ayrı kavram ve teknikle modellenebilmektedir. Bunun nedeni ve sonucu olarak modelin çözüm yöntemleri bu iki tekniğe dayanmaktadır. Biri diğerini açıklayabilir nitelikteki bu teknikler daha iyi çözüm yöntemlerine olanak yaratmışlardır.

DAM'ne ilk çözüm algoritması 1959'da Barsow tarafından geliştirilmiştir [21]. Szwarc'ın Doğu Avrupa'daki bu konuyla ilgili çalışmalardan da söz ettiği makalesinde Barsow'

---

[21] W.Szwarc, "Some Remarks On The Time Transportation Problem", *Nav.Res.Log.Quart.*, Vol. 18, 1971,s. 473

dan sonra geliştirilen ve Barsow'unki gibi uluslararası yayınlarda ve dillerde yayınlanmamış bir kaç çalışmadan bahsedilmektedir. Batılı araştırmacı ve yazarlar Zukhovitskıy ve Avdeyeva'nınki dışında bu çalışmalardan söz etmemişlerdir. Uluslararası kaynaklarda rastlanan ilk iki çalışma Szwarc ve Hammer [22]'ın birbirinden pek farklı olmayan algoritmalarıdır [23]. Daha sonra 1971'de Garfinkel ve Rao yaygın olarak kabul gören asal (primal) ve eşik (threshold) algoritmaları [24]; 1975'de Frieze [25]; 1976'da Srinivasan ve Thompson [26]; 1984'de Isermann'ın çalışmaları görülmektedir [27]. Uluslararası kaynaklara girmemiş olmakla birlikte Grabowski'

---

[22] P.L.Hammer. "Communications On 'The Bottleneck Transportation Problem' and 'Some Remarks On the Time Transportation Problem' ". *Nav.Res.Log.Quart.*, Vol. 18, 1971, s. 487-490.

[23] V.Srinivasan, G.L. Thompson, "Algorithms for Minimizing Total Cost, Bottleneck Time and Bottleneck Shipment in Transportation Problems",*Nav.Res.Log.Quart.*, Vol. 23, 1976, s. 568.

[24] R.S.Garfinkel, M.R. Rao, "The Bottleneck Transportation Problem", *Nav.Res.Log.Quart.*, Vol. 18, 1971, s. 465-472.

[25] A.M.Frieze, "The Bottleneck Linear Programming",*Operational Research Quart.*, Vol. 26, No. 4, 1975, s. 871-874.

[26] V.Srinivasan, G.L.Thompson, "Algorithms for... s.567-595.

[27] H.Isermann, "Linear Bottleneck Transportation Problems", *Asia-Pacific Journal of Operational Research*, Vol.1,1984, s. 38-52.

nin algoritması da dikkate değerdir [28].

Aşağıda, darboğaz programlama algoritmalarına esin kaynagı olması ve yaygın olarak benimsenmesinden dolayı Garfinkel ve Rao'nun asal ve eşik algoritmaları verilmiştir.

II.3.1 Asal (Primal) Algoritma

Yöntem, 1971 yılında Garfinkel ve Rao tarafından tanıtılmış olup, başlangıç bir uygun çözümden hareketle eniyi çözümün araştırılması esasına dayanır [29].Yöntem uyarınca iki ayrı modelin çözümü araştırılır. Bunlardan biri bölümün başında verilen genel DAM, diğeri ise; Z^* bu modelin bir uygun çözümündeki darboğaz değer olmak üzere,

$$\text{Enk} \quad U = \sum_{\{(i,j) \mid C_{ij} = Z^*\}} X_{ij}$$

işleminin tanımlanmasıdır. Burada, ilk modelin her bir uygun çözümü için yukarıdaki enküçükleme işlemi yapılır. Bu işlemleri içeren asal algoritma aşağıdaki gibi verilebilir.

A.1 Modele herhangi bir başlangıç uygun çözüm bul. Bu $\hat{X} = (\hat{X}_{ij})$ olsun.

---

[28] W.Grabowski, "Problem of Tranportation in Minimim Time", *Bulletin De L'Académie Polonaise Des Sciences*, Vol. 12, No. 2, 1964, s. 107-108.

[29] R.S.Garfinkel, M.R.Rao, "The Bottleneck ...,"s.465-468.

A.2

$$\hat{Z} = \underset{(i,j)}{\text{Enb}} \{c_{ij} \mid \hat{x}_{ij} > 0\}$$

ve

$$t_{ij} = \begin{cases} 1 & c_{ij} = \hat{Z} \text{ ise} \\ 0 & c_{ij} < \hat{Z} \text{ ise} \\ M & \text{(yeterince büyük)} \ c_{ij} > \hat{Z} \end{cases}$$

olmak üzere $T = [t_{ij}]$ matrisini oluştur.

A.3 $T = \{t_{ij}\}$ leri maliyet matrisi olarak alıp doğrusal atama modelini kur ve çöz. Elde edilen çözüm $\hat{X}$ olsun.

A.4 Eğer amaç fonksiyonunun değeri sıfır ise 2. adıma git. Değilse $\hat{X}$, DAM'nın eniyi çözümüdür.

İkinci adımda, $\hat{Z}$ değerinin azalacağı açıktır. Bu ise yukarıdaki algoritmanın sonlu sayıda adımla eniyi çözüme erişebileceğini gösterir.

II.3.2. Eşik (Threshold) Algoritması

Bu algoritma da 1971'de Garfinkel ve Rao tarafından tanıtılmıştır [30]. Algoritmanın temeli 1968'de Edmands ve Fulkerson tarafından oluşturulmuştur [31]. Garfinkel, 1970 de klasik doğrusal atama modeli için bu yöntemi kullanmıştır [32].

---

[30] R.S.Garfinkel, M.R.Rao, "The Bottleneck ..."S. 468-470.

[31] J.Edmands, D.R.Fulkerson,"The Bettleneck..."s. 300-306.

[32] R.S.Garfinkel, "AnImproved Algorithm ... " s. 1747-1753.

Yöntemin esası amaç fonksiyonunun alt sınır değerlerini türeterek çözüm aramak olduğundan, alt sınırlara atfen eşik algoritması veya yöntemi olarak anılır.

Herhangi bir i için, P_i, $\{1,2,\ldots,n\}$ nin bir birleşimi olmak üzere aşağıdaki koşulu sağlasın.

$$C_{i,p_i(1)} \leqslant C_{i,p_i(2)} \leqslant \ldots\ldots \leqslant C_{i,p_i(n)}$$

Ayrıca $S_{ij} = \sum_{j=1}^{k} b_{p_i(j)}$ ve $S_{ik^*} = \underset{k}{\text{Enk}}\{S_{ik} \mid S_{ik} > a_i\}$

olarak tanımlandığında $r_i = C_{i,p_i(k^*)}$, Z^* için bir alt sınır olmaktadır. Çünkü i. kısıt r_i'den daha küçük C_{ij}'ye sahip X_{ij} lere atama yaparak sağlanamaz.

Benzer biçimde herhangi bir j için q_j,

$$C_{q_j(1),j} \leqslant C_{q_j(2),j} \leqslant \ldots \leqslant C_{q_j(m),j}$$

özelliğini sağlayan $\{1,2,\ldots,n\}$ in bir birleşimidir.

Bundan başka,

$$S_{kj} = \sum_{i=1}^{k} a_{q_j(i)}$$

ve

$$S_{k^*j} = \text{Enk}\ \{S_{kj} \mid S_{kj} > b_j\}$$

olarak tanımlandığında $V_j = C_{q_j(k^*),j}$'nin de Z^* için bir alt sınır olacaktır. Nihayet aşağıdaki diziden Z^* ın eniyi alt sınırı bulunur.

$$\underline{Z}^{o} = \text{Enb}\,\{r_i,\ v_j\}\ ,\quad i = 1, \ldots, n,\ j = 1, \ldots . m$$

Bu açıklamalara bağlı olarak algoritma aşağıdaki gibi verilebilir.

A.1 F, s'den t'ye olan akış miktarı (atama miktarı) olmak üzere başlangıç olarak ($-\infty$) değerini alsın. Z'nin alt sınırı $\underline{Z} = \underline{Z}^{o}$ olsun. Yani $F = -\infty$ ve $\underline{Z} = \underline{Z}^{o}$ 'dır.

A.2 $C_{ij} \leqslant \underline{Z}$ olan tüm ($i \to j$) okları (veya X_{ij}'ler) atama için kabul edilebilirdir. Bu okların oluşturduğu şebekede $\hat{F}$ enbüyük akışını bul. $W = \sum_i a_i$ iken $\hat{F} = W$ ise $Z^* = \underline{Z}$ amaç fonksiyonunun eniyi değeridir. Eğer $\hat{F} < W$ ise üçüncü adıma git. Değilse dördüncü adıma git.

A.3 $F = \hat{F}$ ve $\underline{Z} = \text{Enk}\,\{C_{ij} \mid$ i'ye akış geliyor, j'ye gelmiyor$\}$ olsun. Bu durumda şebekede daha fazla akış sağlanacağı açıktır. İkinci adıma git.

A.4 İkinci adımda akış elde edilen oklar $\hat{X} = (\hat{X}_{ij})$ olsun. Eğer

$$\hat{U} = \underset{\{(i,j) \mid C_{ij} = Z^*\}}{\sum \hat{X}_{ij}}$$

[illegible]

[illegible] ise beşinci adıma git.

A.5 Tüm kabul edilebilir oklar için,

$$t_{ij} = \begin{cases} 1 & \text{eğer } C_{ij} = Z^* \text{ ise} \\ 0 & \text{eğer } C_{ij} < Z^* \text{ ise} \end{cases}$$

oluşturulur. Kabul edilebilir okları ve $T = \{t_{ij}\}$ maliyet matrisini kullanarak doğrusal atama modelini oluştur ve çöz.

Dördüncü adımda $\hat{U}$ darboğaz ok boyunca oluşan akışı göstermektedir. Kabul edilebilir oklar kümesi her adımdan sonra artmakta böylece istenen atama miktarı kadar akış elde edilene dek işlemlere devam edilmektedir.

II.4 ÖZEL DARBOĞAZ ATAMA MODELLERİ VE ÇÖZÜM YÖNTEMLERİ

Bu kesimde, birinci bölümde yapılan sınıflamaya uygun olarak, Klasik Darboğaz Atama Modeli (KDAM), Kapasite Kısıtlı veya Üst Sınırlı Darboğaz Atama Modeli (ÜDAM) ve Ağırlıklı Darboğaz Atama Modeli (ADAM) üzerinde durularak son ikisi birlikte ele alınacaktır.

II.4.1 Klasik Darboğaz Atama Modeli

Atama modelini çalışma alanı yapmış olan araştırmacılar genellikle $a_i = b_j = 1$ biçiminde toplu atamaya izin vermeyen modelleri atama modeli başlığına almış ve $a_i > 1$ ve $b_j > 1$ şeklinde toplu atamaya izin verenleri ulaştırma modeli başlığına terketmiş ve DAM başlığında bu kesimin konu edendiği KDAM 'ni ele alıp çözüm yöntemleri önermişlerdir. Bu çözüm yöntemlerinin ilki 1953 yılında Fulkerson, Glicksberg ve Gross tarafından önerilmiş olup etkin değildir [33]. İlk etkili çözüm yöntemi 1959'da Gross tarafından geliştirilmiştir. Daha sonra Porsching (1963), Page (1963), Edmonds ve Fulkerson (1970), Garfinkel (1971), Bhatia (1977). Derigs ve Zimmermann (1978), Carpaneto ve Toth (1981) konuyla ilgili çalışmaları ötelemişlerdir.

Sözü edilen çalışmalardan Page ve Bhatia'nınki Macar Algoritmasından yararlanmayı hedefler ve bu amaçla birtakım dönüşümler tanımlar. Edmands ve Fulkerson 1970'de yayınlanan makaleleriyle *"eşik"* mantığını gündeme getirmişlerdir. Aynı yıl Garfinkel eşik algoritmasını geliştirmiştir. Bu algoritma önceki kesimde DAM için geliştirilen algoritmanın temelini oluşturur. Derigs ve Zimmermann 1978'de Dijkstra'nın en kısa yol algoritmasının bir düzenlemesinden ve Tomizawa'nın konuyla ilgili çalışmalarından yararlanarak yörünge türetme yöntemini (Augmenting Path Method) geliştirmişler ve bir bilgisayar programı vermişlerdir [34]. Carpaneto ve Toth, 1981'de Derigs ve Zimmermann'ın algoritmasında bir takım düzenlemeler yaparak genellikle daha etkin olan yeni bir bilgisayar programı yazmışlardır [35].

---

[33] J.Edmonds, D.R.Fulkerson, "The Bottleneck ... " s. 299.

[34] U.Derigs,U.Zimmermann, "An Augmenting Path Method for Solving Linear Bottleneck Assignment Problems", *Computing 19*, 1978, s. 285-295.

[35] G.Carpaneto,P.Toth, "Algorithm for the Solution of the Bottleneck Assignment Problem",*Computing 27*,1981, s.179-187.

Şu anda KDAP için en etkin çözüm algoritmasının bu olduğu söylenebilir.

Genel olarak KDAP aşağıdaki gibi modellenmektedir.

$$\sum_{j=1}^{m} x_{ij} \leqslant 1 \quad , \quad i = 1, \ldots , n$$

$$\sum_{i=1}^{n} x_{ij} \geqslant 1 \quad , \quad j = 1, \ldots , m$$

$$x_{ij} \geqslant 0 \text{ ve tamsayı}$$

kısıtları altında,

$$\text{Enk } Z = \underset{\substack{i=1,\ldots,n \\ j=1,\ldots,m}}{\text{Enb}} \{c_{ij} | x_{ij} > 0\}$$

Modelin kısıtlar kümesi basit işlemlerle eşitlik haline getirilebildiğinden çözüm yöntemleri bu durum için geliştirilmişlerdir. Bu durumda m = n olmaktadır.

i) Gross Algoritması

Yöntem, Gross tarafından 1958 yılında geliştirilmiş olup, 1959'da yayınlanmıştır [36]. Herhangi bir başlangıç uygun atamadan hareketle Z değerinin adım adım eniyilenmesi temelindedir. Algoritmanın adamları aşağıdaki gibidir.

A.1 Herhangi bir başlangıç uygun çözüm bul.

---

[36] J.Edmonds, D.R.Fulkerson, "The Bottleneck...." S. 299.

A.2 $Z = \text{Enb}\ \{C_{ij} | X_{ij} > 0\}$

$i=1, \ldots, n$

$j=1, \ldots, m$

A.3 Z değerini veren X_{ij}'ye ait yörüngeyi oluştur. Yörünge oluşturulamıyorsa mevcut çözüm eniyidir, durulur. Değilse dördüncü adıma gidilir.

A.4 Yörünge üzerinde yeni atama yapılır ve ikinci adıma gidilir.

Yukarıdaki algoritmanın üçüncü adımında sözü edilen yörünge oluşturma işlemi şöyledir, Z değerini verin X_{ij} yörüngenin ilk ve son öğesi olacaktır. Bu öğenin bulunduğu j sütununda $C_{ij} < Z$ olan atama yapılmamış bir öğe bulunur. Bu öğenin bulunduğu satırda atama yapılmış olan öğeye gidilir ve işlemler bu şekilde sürdürülür. İşlemler esnasında eğer atama yapılmış öğe olarak X_{ij}'e gelinirse yörünge tamamlanmıştır.

Bu algoritma, en uç durumda her bir karar değişkeni için bir yörünge araştıracağından en fazla n^2 yörüngede eniyi çözümü bulacaktır.

ii) Bhatia Algoritması

KDAM'ni, Klasik Doğrusal Atama Modelinin etkin çözüm tekniği olan Macar Algoritması ile çözülebilir hale getirmeyi ve bu algoritmayı ardışık olarak kullanmayı temel alır [37].

---

[37] H.L.Bhatia,"Time Minimizing Assignment Problem",*SCIMA*, Vol. 6, No. 3, 1977, s. 75-83.

Algoritma aşağıdaki adımlardan oluşmaktadır.

A.1 $X_1 = |X_{ij}|$ herhangi bir başlangıç uygun çözüm iken karşı gelen darboğaz değer Z_1 şöyle bulunur.

$$Z_1 = \underset{(i,j)}{\text{Enb}} \{C_{ij} \; X_{ij} = 1\}$$

A.2 Z_1 değerine karşı gelen klasik doğrusal atama modeli aşağıdaki gibi oluşturulur.

$$t'_{ij} = \begin{cases} 0 & \text{eğer} \quad C_{ij} < Z_1 \quad \text{ise} \\ 1 & \text{eğer} \quad C_{ij} = Z_1 \quad \text{ise} \\ \infty & \text{eğer} \quad C_{ij} > Z_1 \quad \text{ise} \end{cases}$$

t'_{ij}, modelin yeni katkı parametreleridir. Amaç fonksiyonu ise,

$$\text{Enk } T_1 = \sum_{i=1}^{n} \sum_{j=1}^{n} t'_{ij} X_{ij}$$

biçiminde oluşturulur. Bu model Macar Algoritması ile çözülür.

A.3 i) Eğer ikinci adımda bulunan eniyi çözüm sonlu ve sıfırdan büyük ise Z_1'in eniyi değeri mevcut olandır. Azaltılması mümkün değildir. Algoritma burada durur.

ii) Eğer ikinci adımda bulunan eniyi çözüm sıfır ise yeni amaç fonksiyonu Z_2'ye veren yeni uygun çözüm $X_2 = |X^2_{ij}|$ bulunur. İkinci adımda Z_2 değeri için tekrarlanır.

İşlemler T_i değeri sıfırdan farklı oluncaya kadar sürdürülür. İkinci adımdan dolayı algoritmanın sonlu olacağı açıktır. Algoritmanın son adımında (k'ıncı adım olsun), $T_k = p$ ise darboğaz değere sahip olup da atamada yer alan öğe sayısı, yani $Z_k = C_{ij}$ olan sıfırdan büyük X_{ij} sayısı p'dir.

Atama yapılması istenmeyen (i,j) çiftleri için enbüyük C_{ij}'den daha büyük olan herhangi bir değerin verilmesi yeterlidir. Ayni işlem Gross algoritması için de geçerlidir.

DAM kolaylıkla KDAM haline getirilebilir. Ancak bu durumda kısıt sayısı oldukça fazlalaşacaktır [38]. Bu modellerin ilk örgün ve etkin çözüm tekniğinin Gross'un yukarıda verilen algoritması olduğu söylenebilir.

II.4.2 Kapasite Kısıtlı ve Ağırlıklı Darboğaz Atama Modelleri

Daha önce verilen kısaltmalarıyla ÜDAM ve ADAM şu ana kadar özgün çözüm tekniklerine sahip olmayan modellerdir. Kapasite kısıtlı (veya üst sınırlı) darboğaz atama modeli,

$$\sum_{j=1}^{m} X_{ij} \leqslant a_i \quad , \ i = 1, \ldots , m$$

$$\sum_{i=1}^{n} X_{ij} \geqslant b_j \quad , \ j = 1, \ldots , n$$

$$0 \leqslant X_{ij} \leqslant d_{ij} \quad , \text{ her } i \text{ ve } j \text{ için}$$

---

[38] R.S. Garfinkel, M.R.Rao, "The Bottleneck ...", s. 465-466.

kısıtları altında,

$$\text{Eniyi } Z = \text{Enb } \{C_{ij} | X_{ij} > 0\}$$

ve ağırlıklı darboğaz atama modeli,

$$\sum_{j=1}^{m} X_{ij} \leqslant a_i \quad , \; i = 1, \ldots , m$$

$$\sum_{i=1}^{n} X_{ij} \geqslant b_j \quad , \; j = 1, \ldots , m$$

$$t_j \leqslant \sum_{i=1}^{n} r_{ij} X_{ij} \leqslant p_j \quad \text{bazı j'ler için}$$

$$X_{ij} \geqslant 0$$

kısıtları altında,

$$\text{Eniyi } Z = \text{Enb } \{C_{ij} | X_{ij} > 0\}$$

biçimindedir. Her iki modelin yapısı mevcut DAM yöntemlerini doğrudan kullanmaya yatkın değildir. Bu durumda modellerin çözümü için iki yol önerilebilir. Birincisi 0-1 biçimindeki karar değişkenlerinden oluşan modeller için geçerlidir. Garfinkel ve Nemhauser'in 0-1 tamsayılı modeller için önerdikleri iki yöntem vardır [39]. Birincisi uygun çözümden hareketle eniyi çözümü araştıran *"asal algoritma"* diğeri DAM'nın çözüm yöntemlerinde tanıtılan algoritmaya benzeyen *"eşik algoritması"* dır.

---

[39] R.S. Garfinkel, G.L. Nemhauser, A.g.k., s. 351-354

Karar değişkenlerinin 0-1 değişken özelliği sağlamadığı modellerin çözümünde Frieze'in 1975'te önerdiği ve karar değişkenlerinin sıfırdan büyük olma koşulunu gözönüne alan *"asal"* ve *"eşik"* algoritmalarından yararlanılabilir [40]. Bu durumda elde edilen çözümler tamsayı olma özelliğini sağlamaz ise uygunluğu bozmayacak biçimde değerler yuvarlatılabilir. Veya aynı amacı sağlamak üzere Frieze'in algoritması Simplex algoritmasının yerine kullanılarak Dal ve Sınır tekniğinin bir düzenlemesi kullanılabilir. Bu öneri tüm tamsayılı darboğaz programlama problemlerine genellenebilir görünmektedir.

II.5 DARBOĞAZ ATAMA MODELİNİN UYGULAMA ALANLARI

DAM'lerin ilk kullanım alanı üretim hattı atama problemleridir [41]. Daha sonra Page, *"i. sınıfın j.derslikteki uygunluk ve rahatını gözönüne alarak sınıfların dersliklere atanması"* problemi ile *"bir ekipmanın parçalarını tamir edecek teknisyenlerin hangisinin hangi parçayı tamir etmesinin uygun olacağı"* problemlerinin DAM ile çözülmesi gereğine değinmiştir [42].

---

[40] A.M.Frieze, "Bottleneck ....." s. 871-872.

[41] M.Klein, H.Takamori, "Parallel Line Assignment Problems", Mgmt. Sci., Vol. 19, no. 1, 1972, s. 1-2.

[42] E.S.Page, "A Note on Assignment Problems", *Comp.Journal* Vol.16, No. 3, 1963, s. 241.

DAM'lerinin potansiyel uygulama alanlarını kestirmek, tüm endüstriyel ve sosyal sistemlerin bugünkü ve gelecekteki yapılarına hakim bir bilgi birikimini gerektirir. Ancak, sözkonusu alanlarla ilgili problemlerden bazıları şu şekilde verilebilir.

- Ara stoklama yerlerinde kapasite tespitleri.
- Birden fazla işi yapabilecek yetenekteki, işgörenlerin iş tatminleri gözönüne alınarak işlere atanması.
- Bir işyerinde üst kademe yöneticilerinin vardiyalı çalışma düzeni içinde kendi tercihlerine göre aylık görev programlarının yapılması.
- N sayıdaki satış elemanının belli sayıdaki bölgelerde tercihlerine göre görevlendirilmesi.

Birinci bölümde atama problemleri genel olarak tanıtılmış ve ilgili modellere değinilmiştir. Bu bölümde ise, birinci bölümde sözedilen darboğaz atama modeli, problem alanları ve çözüm yöntemleriyle ele alınmıştır. İzleyen bölümde, "*bitirme ödevlerinin öğrencilere atanması*" probleminin DAM ile çözümü ve model parametrelerinin elde edilmesi ile ilgili özellikler verilecek ve uygulamanın sonuçlarına değinilecektir.

Ü Ç Ü N C Ü   B Ö L Ü M

BİTİRME ÖDEVLERİ DAĞITIMI İŞLEMİNİN DARBOĞAZ ATAMA MODELİYLE YAPILMASI

III.1 BİTİRME ÖDEVLERİNİ ATAMA PROBLEMİ

Üniversitelerimizin dört yıllık fakülte veya yüksek okullarının genellikle son sınıf ikinci dönemindeki bitirme ödevi dersi öngörülmüştür. Bitirme projesi, bitirme tezi, lisans tezi gibi isimlerle de anılan bitirme ödevleri lisans öğrenimi boyunca alınmış olan meslek derslerinin uygulamada kullanılma yeteneğini geliştirmek ve ölçmek amacını gütmektedir. Bu amacın yerine getirilmesi, sözkonusu dersle ilgili bir takım ara hedeflerin belirlenip gerçekleştirilmesine bağlıdır. Bitirme ödevlerini atama problemi bu ara hedeflerle yakın ilgili olup, kendisi de gerçekte bunlardan biridir.

Bitirme ödevi dersinin yürütülmesi, ilgili birimin (bölümün) başkanlıklarının denetim ve kontrolunda öğrencilerin atandıkları danışman (yürütücü) öğretim elemanlarınca yapılır. Genellikle danışmanlık görevini üstlenen kişilerin beklentileri *"iyi öğrenciler"*dir. Bunun yanısıra öğrenciler de *"iyi konu"* ve *"iyi öğretim elemanı"* ile çalışmak istemektedirler. Bu kavramların kişiden kişiye değişebilir nitelikleri olmakla birlikte, gerek öğretim üyeleri gerekse öğrencilerin iletişimi ve etkileşimi fazla olan gruplar olduğu düşünülürse sözkonusu beklentiler iki taraflı olarak belli noktalarda yoğunlaşabilmektedir. Sözkonusu beklentilerdeki yoğunluk; konuyla ilgili kararları daha önemli bir hale getirmektir.

Bitirme ödevlerinin yürütülmesi aşağıdaki işlemleri gerektirir.

i) Bitirme ödevi dersini alacak öğrencilerin belirlenmesi : Bu dersi alacak öğrencilerin bir önceki dönemden hiç bir derslerinin kalmaması gerekmektedir. Yani, öğrenci bitirme ödevi dersini aldığı dönem mezun olma olanağına sahip olmalıdır.

ii) Bitirme ödevi dersini yürütecek danışmanların belirlenmesi : Danışmanlık yapacak kişilerin öğretim elemanı olması gerekmektedir. Bitirme ödevi alacak öğrencilerin sayısının 3 veya daha az olması halinde bunların tek bir öğretim elemanında toplanması istenebilir.

iii) Bitirme ödevi konularının belirlenmesi : Konuların belirlenmesi, yürütme görevini yapacak danışmanlar tarafından veya bölüm başkanlıkları tarafından yapılmaktadır. Bölüm başkanlıkları genellikle alt alanları -ki bunlar her sene aynı kalmaktadır- belirlemektedir. Bu alanların birinde uzmanlaşmış öğretim elemanı ise öğrencilere çalışacakları konuları vermektedir. Nadiren, danışmanlar bölüm başkanlığı denetiminde özgün konu başlıklarını ödevlerin öğrencilere dağıtımından önce belirler. Bu yüzden öğrenci çoğunlukla seçtiği alanda hangi konuda çalışacağından habersiz olarak belirli bir alana yönelmek durumundadır.

iv) Bitirme ödevi konularının dağıtılması : Bu işlem çoğunlukla konuların değil, danışmanlara öğrencilerin atanması biçiminde yürütülmektedir. Böylece bitirme ödevi konularının belirlenmesi işlemi danışmanlara öğrencilerin atanması sonrasının bir alt eylemi haline gelmektedir. Konuların önceden belirlendiği durumlarda da bunların öğrencilere dağıtımı sözkonusu olup, tatmin edici bir yöntem bulmak güçtür.

v) Bitirme ödevlerinin yürütülmesi ve değerlendirilmesi : Sözkonusu ders özgün amacının yanısıra, diğer derslerden farklı yürütme ve değerlenmeye sahiptir. Ders büyük oranda öğrencinin çabasına dayalı olup, danışmanın yönlendiriciliği ve denetimi önemli rol oynamaktadır. Değerlendirme ise öğrencinin yıl içi çalışmalarının ve dönem sonunda ortaya koyduğu bitirme ödevinin ve bazen de onu sunuş biçiminin bir birleşimidir.

Meslek yaşamının okuldaki bir uygulaması niteliğindeki bu dersin başarılı olması yukarıdaki adımların yürütülme biçimiyle ilgili olacaktır. Bu noktada bir dönem üzerinde çalışacağı ve öncül bir meslek deneyimi yaşayacağı konuyu belirlemede öğrencinin payı ve rolü olmalıdır. Konuların belirlenmesinin, danışmanların belerlenmesinin bir fonksiyonu olması ve bununda genellikle bir seçim işlemi içermemesi, öğrencilerin eğilimlerini gözönüne almayı, bitirme ödevlerinin dağıtılması aşamasına bırakmaktadır.

Bu dağıtım işleminde öğrencilerin konular arasında tercihlerine yer vermek, onların bu dersteki başarılarını arttırıcı rol oynayabilecektir. Böylece ileride çalışacakları meslek alanını seçmek ve o alandaki problemlere yakınlık sağlamak gibi gözardı edilmeyecek bir fayda elde edilebilir. Bitirme ödevlerinin yürütülmesi aşamasında karşılaşılacak güçlükler için öğrenci, bu tercihlerde güdülenmiş olacaktır. Ancak bu durum tercihlerinden birine girmiş öğrenciler için sözkonusudur. Tercihlerinden birine girmeyen öğrencilerde ise, ters bir güdüleme veya küskünlük sözkonusu olabilmektedir. Böyle bir öğrenci kendisinin *"istenmediği"* duygusuna kapılabilir ve atandığı bitirme ödevinde istenen başarıyı elde edemeyebilir.

Önerilen konulardan bir kısmının tercih edilmesine yönelik bir dağıtım beraberinde *"nasıl bir dağıtım"* sorusunu gündeme getirecektir. Bitirme ödevleri öğrencilere öyle biçimde atanmalıdır ki tercihlerle -ki genellikle yığılmalardan ötürü kaçınılmazdır- ortaya çıkacak hoşnutsuzluklar enaz olsun.

Konuyu İngiliz Üniversiteleri için inceleyen L.G.Proll da benzeri sakıncaları ortaya koymakta ve aşağıdaki iki nedenden ötürü danışmanların öğrenci tercihi yapmalarını uygun görmemektedir [43].

i) Çoğu durumlarda, danışmanların tercihlerini yapabilecekleri objektif bir kıstas olmayacaktır. Yani bitirme ödevlerinin atanacağı öğrencilerle bazı danışmanlar arasında hiçbir ilişki olmayabilir veya çok az olabilir. Böyle durumlarda danışmanlar tercihlerini eski sınav sonuçlarına dayandırmaya teşvik edilmiş olur. Oysa sınavlar bir öğrencinin projeleri yapma ve farklı inceleme alanlarını ilişkilendirmeden çok, farklı hücrelerdeki yeteneğini ortaya koyabilir.

ii) Geçmişte, üniversite yönetiminin her seviyesinde, öğrenci ilgisinde ortaya çıkan genel görünüş, danışmanlara tercih hakkı vermenin sakıncalı olabileceğini göstermektedir. Çünkü bu tercih öğrenci kayırmaya kadar varabilmektedir.

Böylece aşağıdaki sonuçlara varılmaktadır.

i) Öğrencilere, bitirme ödevi yapacakları konular üzerinde seçme olanağı vermek yararlıdır.

ii) Danışmanlara aynı olanağın verilmesi sakıncalıdır.

Bu durumda bitirme ödevlerinin öğrencilere dağıtılması problemi şöyle tanımlanabilir. Belirlenmiş bitirme ödevi ko-

[43] L.G.Proll, "A Simple Method of Assigning Projects to Students", *Opr.Res.Quart.*, Vol. 23, 1972, s. 195-196.

nularından bir kısmını, tercih eden öğrencilere kabul edilebilir bir kıstas altında dağıtmaktır. Böylece problem öğrenci - danışman ikilisinden ayrıştıralarak, öğrenci-bitirme ödevi ikilisi arasında bırakılarak, elde edilecek çözümlerin öğrenciyi rahatsız etme olasılığı azaltılmaktadır. Konuyu öneren danışmandan habersiz kılınan öğrenci, olayın daha başında dersin sorumluluğunun üzerinde olduğunu kabullenmek durumunda kalmaktadır. Öğrenci, bu durumda biçimsel olmayan ilişkilerinin kuvvetli olduğu veya kendisine daha fazla yardım ve kolaylık göstereceğine inandığı ve belki de bunu kendisine vaadeden danışmanı seçebilme olanağından yoksun bırakılmaktadır. Bunun yanısıra konularla ilgili ek bilgi ve ilgili olduğu dersler konularla beraber öğrenciye verilebilir.

Danışman, belki kendi kıstaslarına göre istediği bir öğrenciyle çalışma olanağına değil ama, önermiş olduğu konuyu benimseyen ve o konuda çalışmaya -biçimsel olmayan ilişkilere güvenmeyerek- hazır bir öğrenciyle karşı karşıya kalmaktadır.

Konuların önceden belirlenmiş olması, danışmanlar arasında ve danışmanlarla yönetim arasında biçimsel bilgi akışına olanak verir. Böylece aynı veya çok benzeyen veya geçmiş senelerdeki çalışmaların tam bir tekrarı niteliğindeki konu önerilerinin gözden geçirilmesi, sözkonusu olabilir. Ayrıca öğrenciye göre konu belirleme olanağı danışmanlara verilmeyerek, olabildiğince eşit seviyede öneriler elde edilebilmektedir. Bu bilgi akışı yönetime basit de olsa, bir denetim şansı vermektedir.

Tüm bu açıklamalar gerçekte temel bir varsayım içeri ler. Bu varsayım, bütün öğrencilerin tercih ettikleri bitir- me ödevlerine atandıklarıdır. Oysa pratikte bunu sağlamak, yığılmalardan ötürü, mümkün değildir. Burada problem, bitir- me ödevi konularından bir kısmını tercih etmiş olan öğrenci- lerin olabildiğince daha çok istedikleri konularda çalışmal rını sağlayacak biçimde, öğrencileri bitirme ödevlerine ata mak şeklinde ele alınmalıdır. İzleyen kesimde, önceki bölüm de tanıtılmış olan darboğaz atama modelinden yararlanılarak bu problemin çözümü araştırılacaktır.

III.2 BİTİRME ÖDEVLERİNİ ÖĞRENCİLERE ATAMA PROBLEMİ ÇÖZÜM YAKLAŞIMLARI

Bitirme ödevlerini öğrencilere atama problemi le ilgili koşullar, aşağıdaki gibi varsayılmıştır.

i) Her öğrenciye bir bitirme ödevi verilecektir.

ii) Grup halinde bitirme ödevi yapılması sözkonusu d ğildir.

iii) Her danışman yürütebileceği kadar konu önerebili

iv) Önerilen konular toplamı öğrenci sayısından az o lamaz.

v) Öğrenciler ilan edilen konulardan belirlenen say dakini öncelik sırasına göre tercih ederler.

Konuyla ilgili niceliksel bir çözüm yaklaşımını Pro geliştirmiştir [44]. Öğrenci tercihlerini gözönüne alan yöntemde mümkün bitirme ödevlerinin sıralanması öğrencilerden istenir. Bununla beraber, çok sayıda bitirme ödevi ve öğrenci sözkonusu olduğunda, atanacak öğrencilerin tüm ödevleri anlamlı biçimde sıralamalarını beklemek gerçekçi olmayacağından her öğrenciden sadece K tercihini sıralaması istenmektedir. Proll'un yöntemi aşağıdaki adımları içerir [45].

i) N bitirme ödevi bulunan liste ilan edilir ve bilinçli tercih yapmak için öğrencilere gerekli olduğu düşünülen bazı açıklayıcı bilgi ilave edilir.

ii) Öğrencilerden N bitirme ödevinden K tanesini seçmesi ve önceliklerine göre sıralaması istenir.

iii) Bir süre sonra bu tercihler alınır ve

$$r_{ij} = \begin{cases} L & \text{, i. öğrenci j. ödevi L. tercihine koymuştur.} \\ K+1 & \text{, i. öğrenci j. ödevi L tercihine dahil etmemiştir.} \\ & (i=1,2,\ldots,M \text{ , } j=1,2,\ldots,N) \end{cases}$$

M öğrenci sayısı ve N proje sayısı iken,

$$r_{ij} = 1 \quad , \quad (i = M+1,\ldots,N \text{ , } j = 1,2,\ldots,N)$$

olmak üzere, $R = [r_{ij}]$ matrisi yazılır.

---

[44] L.G. Proll, "A Simple ....", s. 195-201

[45] L.G. Proll, "A Simple ....", s. 196-198.

iv) i. öğrenciye j. konu verilirse $X_{ij} = 1$, diğer durumlarda $X_{ij} = 0$ olmak üzere;

$$\sum_{j=1}^{N} X_{ij} = 1 \qquad (i = 1, 2, \ldots , N)$$

$$\sum_{i=1}^{N} X_{ij} = 1 \qquad (j = 1, 2, \ldots , N)$$

$$X_{ij} = 0, 1$$

kısıtları altında,

$$\text{Enk } Z = \underset{1 \leqslant i \leqslant N}{\text{Enb}} \{r_{ij} | X_{ij} = 1\}$$

darboğaz atama modeli çözülür.

v) Z^* darboğaz atama modelinin eniyi çözümü olmak üzere $D = [d_{ij}]$ matrisi aşağıdaki gibi oluşturulur.

$$d_{ij} = \begin{cases} r_{ij} & \text{eğer} \quad r_{ij} \leqslant Z^* \quad , \ i = 1,2,\ldots,N \\ \infty & \text{eğer} \quad r_{ij} > Z^* \quad , \ j = 1,2,\ldots,N \end{cases}$$

Daha sonra d_{ij}'lere göre doğrusal atama modeli aşağıdaki gibi yazılır ve çözülür.

$$\sum_{j=1}^{N} X_{ij} = 1 \qquad (i = 1, \ldots , N)$$

$$\sum_{i=1}^{N} X_{ij} = 1 \qquad (j = 1, \ldots, N)$$

$$X_{ij} = 0 \text{ veya } 1$$

kısıtları altında,

$$\text{Enk } Z = \sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij} X_{ij}$$

Öğrenci ve ödevlerin birebir eşlenmesi gerektiğinden, klasik darboğaz atama modeline özgü kısıtlar yazılmıştır. Bununla beraber, eniyileme problemi için amaç fonksiyonunun tercihi böylesine açık değildir ve bazı açıklamaları gerektirir.

Bir amaç fonksiyonunun oluşturulması problemine her öğrenciye ilk tercihindeki bitirme ödevinin atanmasıyla mümkün olabilecek mükemmel atama incelenerek yaklaşılabilir. Özellikle bitirme ödevleri (veya onların yürütücüleri) nadiren aynı rağbette olduğundan, uygulamada fiilen buna erişmek olanaksızdır. Bu durum, atamanın etkinliğini ilk tercihine atanmış öğrencilerin sayısı ile ölçmeye teşvik etmektedir. Bu ölçüyü en büyüklemeye çalışmak, gizli olarak, bir öğrencinin ilk tercihi ile diğerleri arasında bir fark olmadığını varsayar ve bu yönüyle savunulamazdır.

Altı ödevi sıralamaları istenmiş olan öğrencilerin sıralamadıkları veya altıncı tercihlerindeki ödevlerine atanmış olmaktan incineceğini varsaymak mantıklı olacaktır. Bununla birlikte incinme duygusu aynı durumda bir kaç öğrenci varsa azalır. Bu, atamanın sıralanmış ödevlerin (K+1)'inci tercih ve (L = 1, ... , K+1) olmak üzere, L tercihli ödevini elde eden öğrenci sayısını gösteren bir frekans diyagramı halinde düşünülmesi ve farklı atamaların ilgili frekans diyagramındaki dağılım ölçüsü ile kıyaslanması gerektiğini akla getirir. Daha önce işaret edildiği gibi, ilk tercihli ödevini elde eden öğrenci sayısı ile de ilgilenilir, ancak çaba, dağılımın en ucunu daraltmakta yoğunlaştırılır. Diğer birdeyişle, darboğaz amaç fonksiyonu, en kötü durumdaki öğrenciler, tercihleri uyarınca olabildiğince bu durumdan uzaklaştırmayı garanti edecek biçimde seçilir.

R matrisinin yapısı, genellikle birden fazla eniyi çözüm üretecek gibidir. Bu özellikten, diğer bazı kıstaslar sebebiyle eniyi çözümlerden birini seçerek atamayı daha da geliştirme olanağı elde edilebilir. Aslında yöntem tarafından nispeten kötü davranılmış öğrenciler üzerinde daha da yoğunlaşarak, frekans dağılımının ortalaması geliştirilmeye çalışılabilir. i. öğrenci kendisinin r_{ij} tercihli ödevine atandığından frekans dağılım ortalaması,

$$\left[\sum_{i=1}^{M} \sum_{j=1}^{N} r_{ij} x_{ij} \right]$$

ile orantılıdır. Buradan yararlanılarak doğrusal atama modeline başvurulmuştur.

Proll'un algoritması öngördüğü ilk tercihine atanan öğrenci sayısına bağlı etkinliği ile FIFO (ilk defa başvuran birinci tercihine atanır) yöntemine göre daha başarılıdır.

FIFO yöntem hiç olmazsa bir öğrenciyi sıralamaya almadığı ödeve atar. Halbuki Proll'un algoritması eğer tercih sayısı (0.2 x N) ve 5'dan daha geniş mertebeden ise bunu sağlamaktadır. Tercih sayısı bu mertebeden olduğunda deneyler atamanın modunun 1, ortalamasının 1.9 ve üst limiti 4 veya 5 olan bir frekans dağılımına sahip olması beklendiğini gösterir.

Yöntemin uygulanmasında karşılaşılan bir problem, bazı öğrencilerin L'den az ödevi işaretlemiş olmalarıdır. Bunun ılımlı bir miktarına yukarıdaki konuçları değiştirmeksizin izin verilebilir ve buradan kısa vadede ciddi bir problem olmayacağı beklenerek sadece küçük bir grup ödev sıralaması gerekir. Ardısıra gelen sınıflar arasında geri bildirimi varsayarak -gerekli olandan daha az sıralama yapmanın daha iyi sonuç verdiğine dair- uzun vadede problem daha baskın olur. Tercihlerini tam yapmayanlara karşı itaatsizlik cezası biçiminde bir önlem düşünülebilir.

Algoritma, darboğaz eniyi çözümü elde ettikten sonra darboğaz tercih sayısındaki öğrenci sayısının üzerinde durmamakta, dikkati dağılımın modunu iyileştirmeye yöneltmektedir. Bu yaklaşım aslında Proll'un kendisinin de eleştirdiği ilk tercihle diğerleri arasında bir fark olmadığı varsayımını gizli olarak içermektedir. Tercihler anlamlı yapıldığı veya öyle yapıldığı varsayıldığı sürece, ikinci tercihinde olabilecek bir öğrencinin bir takım etkinlik ölçüleriyle üçüncü tercihinde kalmasına göz yumulması mantıklı görünmemektedir.

Bu mantık benimsendiği takdirde etkinlik ölçüsü ilk tercihindeki öğrenci sayısı olmaktan çıkarılmalıdır. Öğrencilerden bir veya bir kaçının istemedikleri ödevlere atanmaları kadar, beş tercihten üç veya dördüncü tercihine atanmaları da yukarıdaki düşünüş içinde, önemlidir. Çünkü öğrenci kendi tercihlerindeki yerini diğer öğrencilerin durumuna bakarak değerlendirecektir. Ama bu durum en uzak tercihindeki öğrencinin değerlendirmesini rahatlatmak üzere bu tercihteki öğrenci sayısının arttırılmasını gerektirmez. Yani her öğrenci bulunabileceği eniyi tercihinde olmalıdır. Bu istek, öğrencilerin tüm tercihlerini anlamlı yapmalarını gerektirir veya varsayar.

Oysa tercih sayılarının belirlenmesi başlı başına bir sorundur. Proll tercih sayılarını 0.2N ve 5 olarak belirler. Halbuki tercih sayısı iki ayrı orana bağlıdır. Bunlardan biri *"bitirme ödevi sayısı / öğrenci sayısı"* dır ki Proll bunu hiç gözönüne almaz. Öyle ki 10 ödev 2 öğrenci olduğunda Proll tercih sayısını 5 olarak öngörür. Öğrenci başına düşen ödev sayısının artması seçim işleminde belli tercihlerdeki yığılmaları azaltacağından tercih sayısını düşünebilecektir. Tercih sayısının azaltılması daha anlamlı tercih yapılmasına olanak vereceğinden tercih sayısının abartılması olumsuz olacaktır. Şu halde tercih sayılarının *"kabaca"* bir takım oran ve rakamlarla belirlenmesi sakıncalıdır.

Tercih sayısını etkileyen diğer bir gösterge tercih sayısının ödev sayısına oranıdır ki Proll sadece bunu gözönüne alır.

Bu açıklamalar aşağıda adımları verilen yeni bir algoritmanın uygulanmasının daha uygun olacağını göstermektedir.

A.1 Bitirme ödevi alacak öğrenci sayısı ve önerilen bitirme ödevi sayısı belirlenir. Bu verilerle eniyi tercih sayısı elde edilir. Tercih sayısı belli bir güvenle tüm öğrencilerin tercih listelerindedi ödevlere girmelerini sağlayacak biçimdedir.

A.2 Tercih sayısı ve bitirme ödevleri öğrencilere ilan edilir ve belli bir süre sonra toplanır.

A.3 Toplanan tercih listeleri ile darboğaz atama modeli aşağıdaki gibi oluşturulur.

N : Önerilen bitirme ödevlerinin toplam sayısı.

N1 : Bitirme ödevi alacak toplam öğrenci sayısı.

K : Her öğrencinin yaptığı tercih sayısı.

C_{ij} : i. öğrencinin j. ödeve verdiği öncelik(1,2,..K) K tercih içinde yer almayanların önceliği K+1 'dir.

Karar değişkeni X_{ij} i. öğrencinin j. ödeve atanması halinde 1, aksi halde 0 değerini almaktadır.

Bu gösterimlerle,

$$\sum_{i=1}^{N} X_{ij} = 1$$

$$\sum_{j=1}^{N} x_{ij} = 1$$

$$x_{ij} = 0 \text{ veya } 1$$

kısıtları altında,

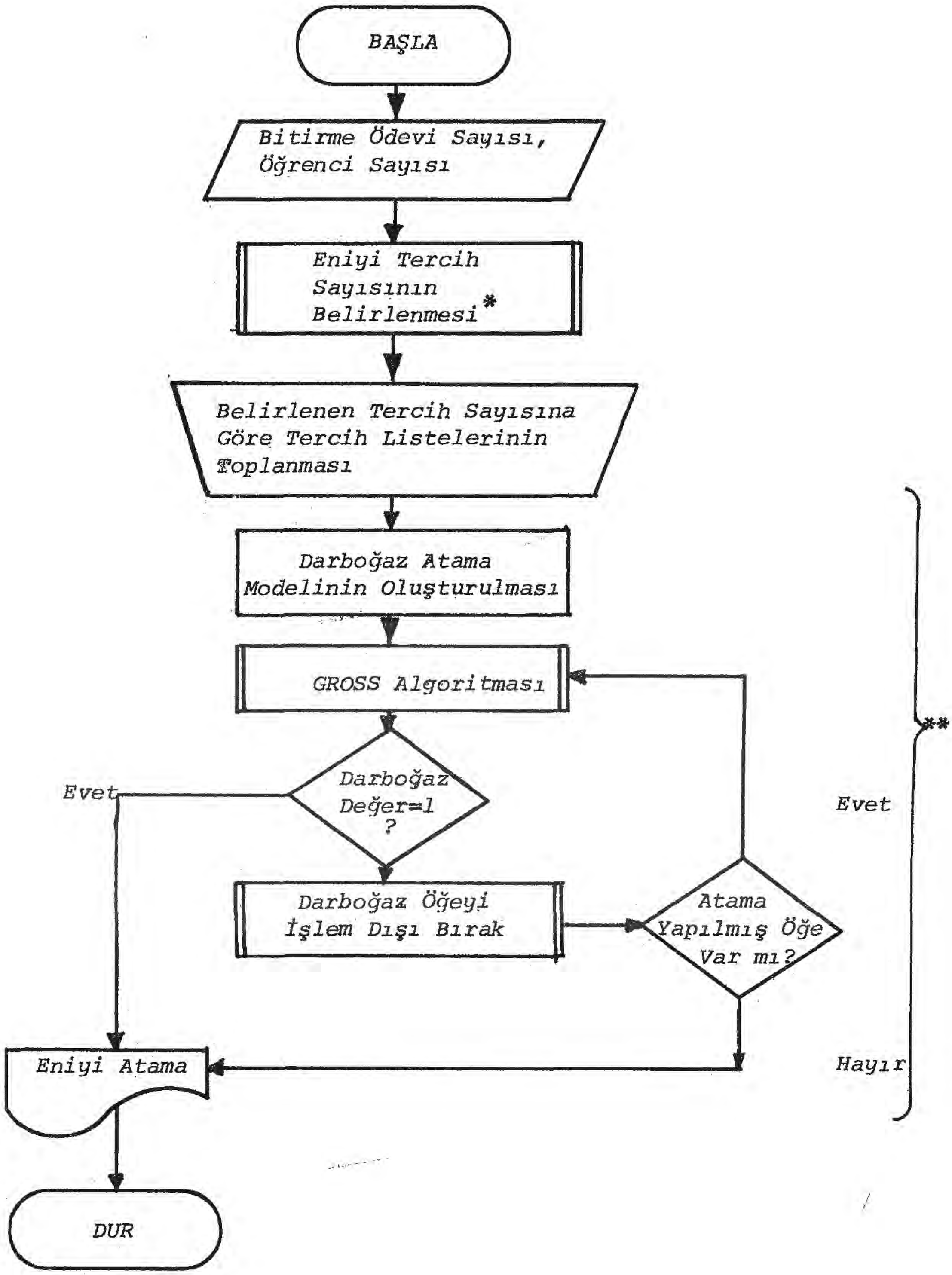
$$\text{Enk } Z = \text{Enb } \{C_{ij} | x_{ij} = 1\}$$

modeli yazılır.

A.4 Modelin eniyi çözümü bulunur. Eniyi çözüm değeri Z^* ve karşı gelen öğe x^*_{ij} olsun. Bu öğrencinin mevcut koşullar altında daha iyi bir tercihine girmesi sözkonusu değildir. Ancak bu atamada yer alan diğer öğrencilerin de artık daha iyi tercihlerine giremeyeceklerini göstermez. Böylece darboğaz öğe işlemlere katılmayarak modelin tekrar eniyi çözümü aranır. İşlemlere her seferinde darboğaz öğe işlemdışı bırakılarak devem edilir. Böylece kalan tüm öğelerin amaç fonksiyonundaki değreleri 1 oluncaya veya tüm öğeler işlem dışı oluncaya kadar sürdürülür.

Yukarıdaki algoritmanın dördüncü adımı ile etkinlik ölçüsü daha az istenen tercihlerdeki öğrenci sayısını enküçüklemek biçimine dönüştürülmüştür. Geliştirilen algoritmanın akış şeması Şekil III.1'de verilmiştir.

Algoritmanın uygulanmasında öğrencilerin bitirme ödevi tercihleri C matrisine yerleştirilir. Matrisin öğrencilere karşı gelen her satırında, K tane 1'den K'ya kadar değer ve (N - K) tane (K + 1) değeri vardır. Böylece öğrencinin ter-



Şekil III.1 : Geliştirilen Algoritmanın Akış Şeması.

* Bu altprogram ile ilgili ayrıntılı akış şeması EK-I'de verilmiştir.

** Programın bu kesimi eniyi atamanın bulunmasıyla ilgili geliştirilen algoritmaya ait olup ayrıntılı akış şeması EK-I'de verilmiştir.

cihleri arasında yer vermediği tüm bitirme ödevleri (K + 1)' inci tercih olarak ele alınmaktadır. Bitirme ödevi sayısının öğrenci sayısından fazla olduğu durumlarda C matrisi (N - N1) kadar yapay öğrenci kullanılarak (N x N)'lik bir matris haline getirilir. Bu durumda yapay öğrencilerin tüm bitirme ödevleri birinci tercihlerinde varsayılmaktadır.

Darboğaz atama modelinin çözüm yöntemi olarak önceki bölümdeki klasik darboğaz atama modeli çözüm tekniklerinden herhangi biri kullanılabilir. Buradan en kötü tercihine düşen öğrenci belirlenmektedir.

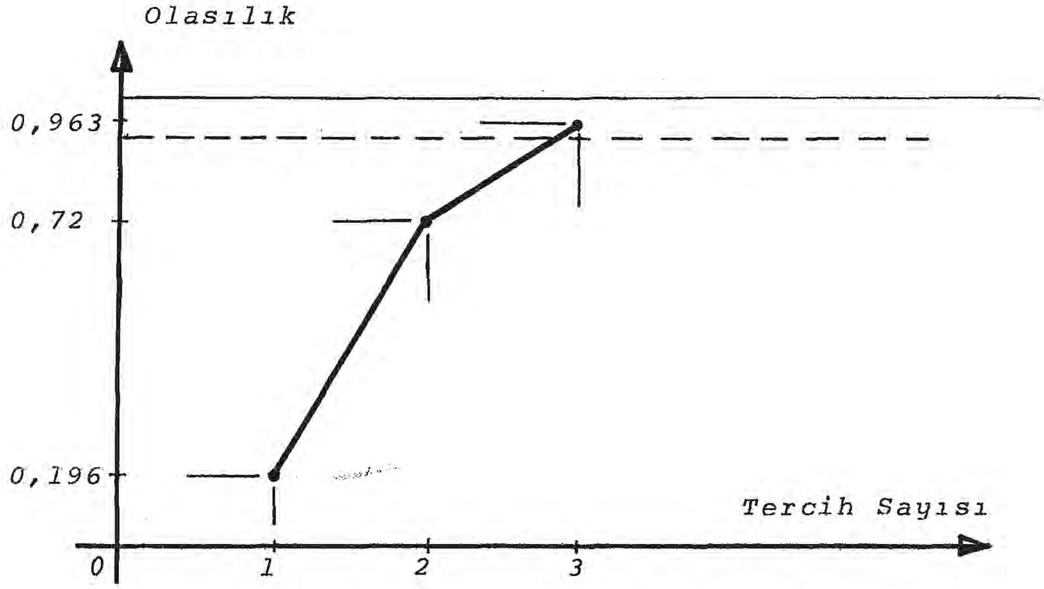
Elde edilen çözüm iki yönlü değerlendirilmektedir. Eğer çözümde darboğaz değer 1 ise tüm öğrenciler birinci tercihlerine atanmış demektir, işlemlere devam etmek gereksizdir. Algoritma burada durur. Eğer amaç fonksiyonunun değeri 1'den büyük ise bu değeri veren öğrenci atama matrisinin dışında tutulur. Böylece yeniden ikinci adıma dönülerek indirgenmiş atama matrisi ile yeni darboğaz eniyi çözüm bulunur. Bundan amaç darboğaz oluşturan öğrenci kadar diğerlerini de elde edebilecekleri eniyi tercihlerine itmektir. Önerilen algoritmanın en önemli özelliği ardışık olarak bir dizi darboğaz atama modeli oluşturup çözmesidir. Böylece her öğrenci tercihlerinin izin verdiği eniyi bitirme ödevine atanmaktadır.

III.3 ENİYİ TERCİH SAYISININ BULUNMASI

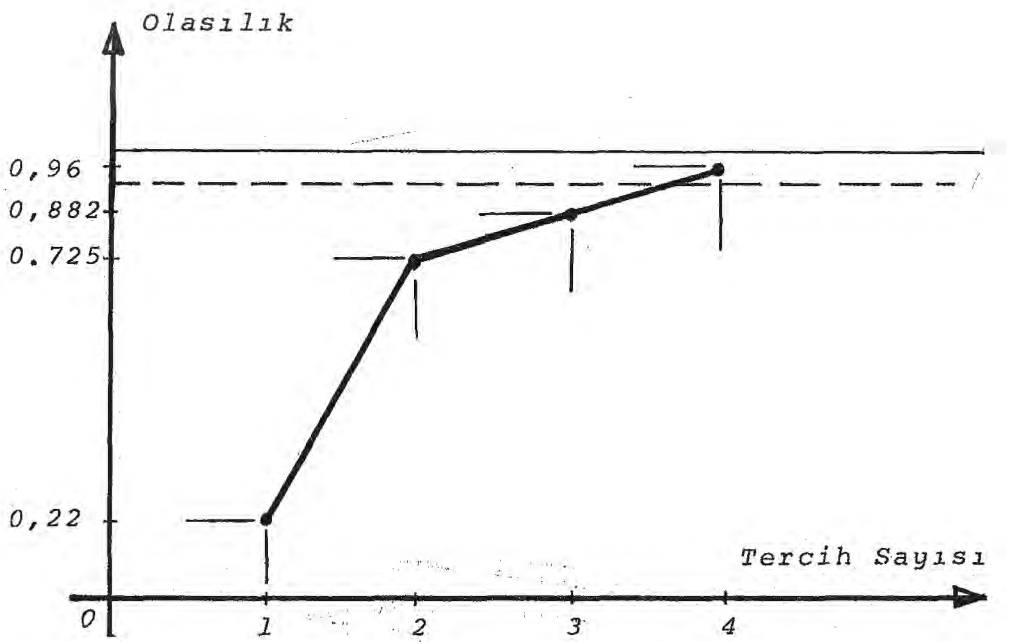
Eniyi tercih sayısının elde edilmesi için verilen bitirme ödevi ve öğrenci sayısına bağlı olarak tercihler rassal olarak yaptırılmakta ve böyle oluşturulan atama matrisi üzerinde eniyi atamalar bulunmaktadır. Bu işlemin sayısı algoritmayı kullanan kişilerin beklentilerine göre değişebilmekte olup, bu sayının arttırılması daha güvenilir bir tercih sayısının elde edilmesini ancak daha uzun bir zamanı gerektirmektedir. Güvenilirliğin bir ölçüsü de, etkinliğin ölçüsü olarak kullanılan istemediği tercihine giren öğrenci sayısının ve aynı zamanda daha az istediği tercihine giren öğrenci sayısının enküçüklenmesine bağlı olarak bu aşamada elde edilen herhangi bir öğrencinin istemediği tercihine girmesi olasılığıdır. Bu olasılık algoritmanın genel uygulamalarında 0,95 olarak alınmış olup daha yüksek veya düşük olarak seçilebilir. Doğal olarak, yüksek seçilmesi tercih sayısının artmasıyla, düşük seçilmesi tercih sayısının azalmasıyla sonuçlanacaktır.

Daha sonra tanıtılacak olan bilgisayar programı yardımıyla tercih sayılarına bağlı olarak herhangi bir öğrencinin istediği tercihlerden birine girme olasılığının değişimi incelenmiştir (Şekil III.2.a- Şekil III.2.b).

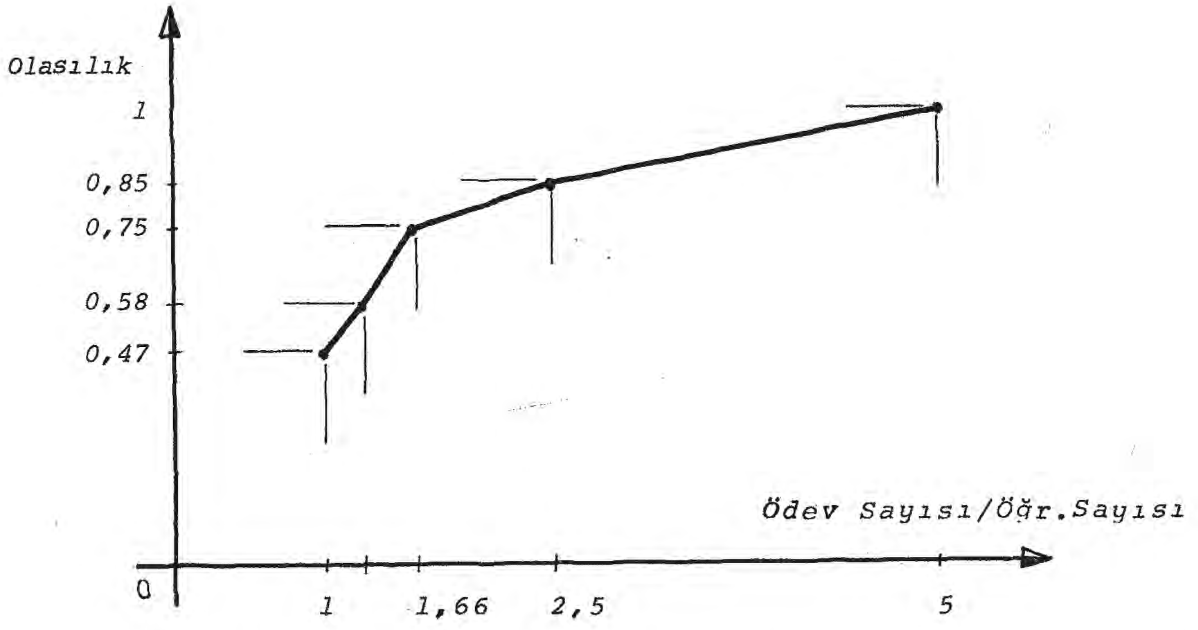
Herhangi bir öğrencinin tercihlerinden birine girmesi olasılığını tercih sayısından bağımsız olarak etkileyen bir oran da *"bitirme ödevi sayısı / öğrenci sayısı"* dır. Bu oran ile herhangi bir öğrencinin 1 tercih yaptığında bu tercihine girme olasılığının değişimi Şekil III.3'de, aynı oranın artışına bağlı olarak tercih sayısındaki değişimler Şekil III.4 de verilmiştir. Görüleceği gibi tercih sayısı ödev sayısının olduğu kadar ödev sayısının öğrenci sayısına oranının bir



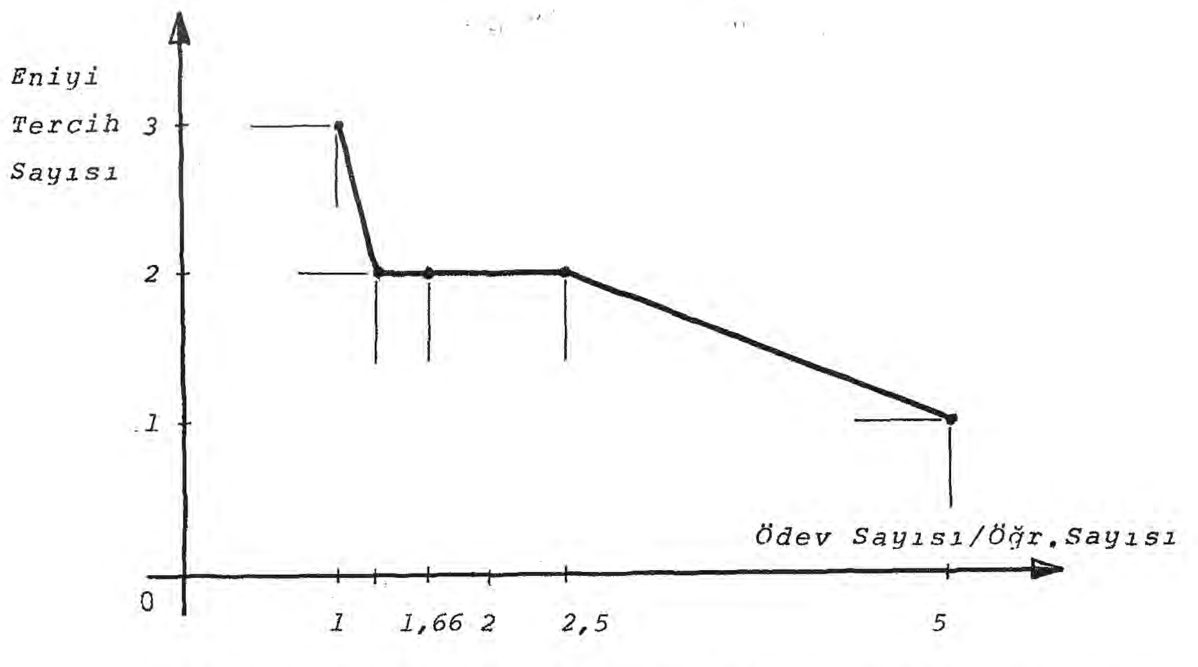
ŞEKİL-III.2(a) Tercih Sayısına Bağlı Olarak Öğrencilerin Tercih Etmedikleri Ödevlerden Birinde Olma Olasılığının Değişimi (Ödev Sayısı:30, Öğr.Sayısı:30, Güvenilirlik Düzeyi:10, Yeterlilik Düzeyi:0,95).



ŞEKİL-III.2(b) Tercih Sayısına Bağlı Olarak İlgili Olasılığın Değişimi (Ödev Say.:40, Öğr.Sayısı:40, Güv.Düz.:10, Yeter.Düz.:0,95).



ŞEKİL III.3 : Tek Tercih 'Ödev Sayısı/Öğrenci Sayısı' Göstergesine Bağlı Olarak Olasılığın Değişimi (Yeterlilik Düzeyi = 10, Ödev Sayısı = 10)



ŞEKİL III.4 : 'Bitirme Ödevi Sayısı/Öğrenci Sayısı' Oranına Bağlı Olarak Eniyi Tercih Sayısındaki Değişimler (Yeterlilik Düzeyi=10, Ödev Sayısı = 10) fonksiyonudur.

III.4 GELİŞTİRİLEN ALGORİTMANIN UYGULANMASI

Önerilen yöntemin ilk işlemi, algoritmanın birinci adımının yerine getirilerek tercih sayısının belirlenmesidir.

Algoritmanın ikinci adımı olan tercihlerin toplanması aşamasında, öğrencilerin hiçbirinin tercih listesine girmeyen ödevler, ödev sayısı öğrenci sayısından az olmayacak biçimde ayıklanabilir. Bu işlem, programın çalışma zamanını azaltacaktır.

Algoritmanın üç ve dördüncü adımları eniyi atamanın bulunmasına yöneliktir. Bu amaçla yazılan bilgisayar programında, önceki bölümde tanıtılmış olan Gross algoritması kullanılmıştır.

Geliştirilen algoritmanın uygulanması amacıyla Basic programlama diliyle bir bilgisayar programı yazılmış olup, bu kesimde tanıtımı ve uygulaması ele alınmıştır. Yazılan programın akış şeması Ek I'de verilmiştir.

Aşağıda sözkonusu programın matris ve dizileri tanımlanmıştır.

i) Tercih Matrisi : 1'den (K + 1)'e kadar olan tercih sıralarından oluşturulur. K(I,J) ile gösterilir. Satırlar öğrencilere, sütunlar ödevlere aittir. (N x N) kadar yer ayrılmıştır.

ii) Atama Matrisi : Sıfır ve birlerden ibarettir. Matriste sürekli N adet bir vardır ve her bir bulunan satır ve sütunda başkaca bir yoktur. A(I,J) ile gösterilmiş ve (N x N) kadar yer ayrılmıştır.

iii) Yörünge Dizileri : Gross algoritması gereği oluşturulacak yörünge üzerindeki öğeleri tutar. Satır, sütun ve öğenin değeri olmak üzere üç ayrı bilgi, üç ayrı dizide tutulmaktadır. Yörüngenin ilk ve son öğeleri aynı olduğundan 3 x (N + 1) kadar alan ayrılmıştır.

iv) Tercih Dizisi : Program iterasyonlar esnasında ataması yapılmış olan öğrencileri, atandıkları tercih sayısına göre bu dizide biriktirir. Böylece her tercihde kaç öğrenci olduğu belirlenir. Ayrılan alan tercih sayısı kadardır.

v) Darboğaz Öğe Dizisi : Darboğaz olup da eniyi çözümde yer alan öğeye ait satır ve sütun numarası program içinde iki dizide tutulur. Böylece ilk iterasyonda darboğaz çözüm olarak bulunan öğe ikinci iterasyonda gözönüne alınmaz. Böylece ardışık darboğaz eniyi atama gerçekleştirilmiş olur. Satır ve sütun için N'er birimlik alan ayrılmıştır.

Program, yukarıda verilen dizi ve matrislerin hepsi için $(2N^2 + 6N)$ kadar bellek alanı kullanmaktadır.

Program iki kısımdan oluşmaktadır. Birinci kısım eniyi tercih sayısının belirlenmesine yöneliktir. Bu amaç için program çalıştırıldığında aşağıdaki verilerin girilmesi gereklidir.

- Bitirme Ödevi Sayısı (N),
- Bitirme Ödevi Alacak Öğrenci Sayısı (N1),
- Güvenilirlik Düzeyi (Ds),
- Yeterlilik Düzeyi (Yd),
- Başlangıç Tercih Sayısı (Bts).

Bitirme ödevi sayısı ile bitirme ödevi alacak öğrenci sayısı arasındaki fark kadar satıra program (1) değerini atayacaktır. Öğrenci sayısı kadar satıra ise rassal bitirme ödevi tercihi yaparak yazacaktır. Bu durumda herhangi bir anda tercih matrisinde 1'den tercih sayısının bir fazlasına, yani (K + 1)'e kadar değerler bulunacaktır. Eğer programda başlangıç tercih sayısı 1 olarak girilmişse ilk rassal tercih matrisi sadece 1 ve 2'lerden oluşacaktır. Programın rassal tercih matrisi yaratmasında en önemli sorun, kullanılan rassal sayı programının iki defa aynı sayıyı üretmesidir. Böyle bir durumda aynı bitirme ödevi aynı öğrencinin iki farklı tercihinde bulunabilecektir. Üretilen sayının rassallığına herhangi bir müdahalede bulunulmamış ancak aynı öğrenci için aynı bitirme örevi seçilmiş ise bu tercihte o ödev numarası red edilerek yeni bir rassal sayı türetilmesi yoluna gidilmiştir. Bu işlemler ilgili öğrencinin tercihlerinde farklı ödevler yer alıncaya kadar sürdürülür.

Güvenilirlik düzeyi olarak girilen veri, sırasıyla herbir tercih sayısında kaç defa rassal tercih matrisi yaratılarak her seferinde kaç öğrencinin istemedikleri tercihlerinde olduğunu bulmak için istenen deney sayısıdır. Deney sayısının artması, deney sonuçlarına bağlı olarak elde edilen, herhangi bir öğrencinin o tercih sayısında istemediği ödevlerden birinde olması olasılığının güvenilirliğini arttıracaktır. Yapılan uygulamalar güvenilirlik düzeyi olarak 10 deneyin yeterli olacağını göstermektedir.

Güvenilirlik düzeyi olarak verilen deneylerle elde edilen herhangi bir öğrencinin o tercih sayısında istemediği ödevlerden birinde olması olasılığının kabul edilecek değeri, yeterlilik düzeyi olarak programa verilir. Programda her deneyde istenmeyen tercihlerdeki öğrenci sayıları toplanır ve (öğrenci sayısı x deney sayısı)'na bölünür. Elde edilen olasılık, verilen yeterlilik düzeyinden büyük veya eşit ise o andaki tercih sayısı eniyi tercih sayısıdır. Eniyi tercih sayısı ve karşı gelen olasılık değeri yazılır ve program durur. Yeterlilik düzeyi mutlak olasılık 1 seçileceği gibi uygun görülen daha düşük bir olasılık da olabilir. Yapılan uygulamalarda yeterlilik düzeyi 0.95 olarak alınmıştır.

Güvenilirlik ve yeterlilik düzeyleri yüksek değerler olacak seçildiğinde programın çalışma zamanı (running time) artmaktadır. Tersi ise elde edilecek eniyi tercih sayısının sağlıksız ve güvenilemez olmasına sebep olacaktır. Bu durumda program çalışma zamanı ne olursa olsun, güvenilirlik düzeyi 10 deneyin altına ve yeterlilik düzeyi 0.95 olasılığın altına düşürülmemesi uygun görünmektedir. Program eniyi tercih sayısını bulmak üzere, tek tercihten itibaren güvenilirlik düzeyi kadar deney sayısı ile işlemlerine başlamakta ve yeterlilik düzeyini sağlayan olasılık değerine eriştiğinde durmaktadır. Yapılan uygulamalar (bitirme ödevi sayısı / öğrenci sayısı) ile (tercih sayısı / bitirme ödevi sayısı) oranlarının belli değerleri için tercih sayısının başlangıçta 1 olarak alınmasının gereksiz olduğunu göstermiştir. Tablo III. 1'de güvenilirlik düzeyi 10 ve yeterlilik düzeyi 0.95 için tavsiye edilen başlangıç tercih sayıları verilmiştir. Böylece, tercih sayısının sözgelimi 2'den başlaması ile 10 deneylik bir bilgisayar çalışma zamanı tasarruf edilmektedir. Bu tasarruflar düşünülerek programda başlangıç tercih sayısının seçimi kullanıcıya bırakılmıştır.

Programın, eniyi tercih sayısının belirlenmesine yönelik bu kesimi eniyi atamayı araştırmamaktadır. Darboğaz değer K olarak bulunduğunda program o deney için (K + 1)'inci tercihindeki öğrencileri saymış olmaktadır. Dolayısıyla bu değer başka bir değişkende biriktirilerek yeni deneye geçilmektedir.

Algoritmanın en önemli işlemlerinden birini oluşturan eniyi atamanın bulunması, programı ikinci kesiminde ele alınmıştır. Öğrencilerden toplanan tercih listeleri sırayla bilgisayara yüklenir. Bu yükleme aşağıdaki bilgilerin programa verilmesi işlemidir.

- Bitirme ödevi sayısı,
- Öğrenci sayısı,
- Tercih sayısı,
- i. öğrencinin 1,2,...,K. tercihindeki ödev numarası.

Bitirme Ödevi Sayısı / Öğrenci Sayısı	Bitirme Ödevi Sayısı				
	10	20	30	40	50
1 < N/N1 < 1.25	2	3	3	4	4
1.25 < N/N1 < 2.5	1	2	2	3	3
2.5 < N/N1 < 4	1	1	1	2	2
4 < N/N1	1	1	1	1	1

TABLO III.1 : "Bitirme Ödevi Sayısı / Öğrenci Sayısı" Oranına Bağlı Olarak Önerilen Başlangıç Tercih Sayısı (Güvenilirlik Düzeyi = 10).

Programın bu kesiminde de öğrenci sayısından fazla bitirme ödevi sayısı kadar satıra 1 değeri ve tercih dışı ödevler içinde (K + 1) değeri verilir. Darboğaz değeri 1 olduğunda veya öğrenci sayısı kadar darboğaz öğe işlem dışı bırakıldığında -veya kısaca işlem içi öğe sayısı sıfır olduğunda- program durmaktadır.

Eniyi atamanın bulunması işlemleri programın birinci kesimine göre daha hızlıdır. Bunun birinci kesimin (güvenilirlik düzeyi x eniyi tercih sayısı) kadar deneyine karşılık bu kesimde sadece bir deney yapılmış olmaktadır.Ancak bu kesimin tek deneyi -ki o da eniyi atamanın bulunmasıdır- ilk kesimdeki gibi K darboğaz değer için durmamaktadır. Belirtildiği gibi, en erken darboğaz değer 1 oluncaya kadar sürmekte aksi halde öğrenci sayısı kadar ardıştırma yapmaktadır. Açıklanan nedenle ikinci kesimin çalışma zamanı (1/Güvenilirlik düzeyi x eniyi tercih sayısı) olmamakta, bu değerin bir miktar üzerinde gerçekleşmektedir. Eğer bitirme ödevleri sayısı, öğrenci sayısından fazla ise, büyük bir olasılıkla hiç bir öğrencinin tercih listesine girmeyen bitirme ödevleri sözkonusudur. Bu ödevler ayıklandığında kalan bitirme ödevi sayısı öğrenci sayısından az ise (öğrenci sayısı - ayıklanmış bitirme ödevi sayısı) kadar öğrenci istemedikleri tercihlerinde olacak demektir. Böyle bir durum, güvenilirlik ve yeterlilik düzeylerinden en az birinin hatalı ve eksik seçildiğini gösterir. Sözkonusu (öğrenci sayısı - ayıklanmış bitirme ödevi sayısı) farkı sıfır veya negatif ise bütün öğrenciler tercihlerinden birine girebileceklerdir. Problemin her iki durumda ayıklanmış olarak bilgisayara verilmesi programın çalışma zamanını azaltacaktır.

Algoritma, 1984 - 85 öğretim yılı bahar döneminde Anadolu Üniversitesi, Mühendislik-Mimarlık Fakültesi, Endüstri Mühendisliği Bölümünde bitirme ödevleri dağıtımı işlemine uygulanmıştır. Bitirme ödevi sayısı (27) ve öğrenci sayısı (18) iken tercih sayısı (3) olarak belirlenmiştir. Tercihler alındıktan sonra hiç seçilmeyen bitirme ödevleri ayıklanmış ve öğrenci sayısına eşit, yani (18) bitirme ödevi kalmıştır. En-iyi atama ile üç öğrenci üçüncü tercihindeki ve diğerleri bir ve ikinci tercihlerindeki ödevlere atanmışlardır. Bu uygulama ile ilgili program listesi Ek II'de ve tercih listeleri ile elde edilen çıktılar Ek III'de verilmiştir.

S O N U Ç

Darboğaz Atama Problemine ilişkin kavram, teknik, model ve çözüm yöntemlerini kendi içinde anlamlı bir bütün haline getirmek, yanırısa yapısı itibariyle DAP olarak karşılaşılan gerçek yaşamdan alının bir probleme çözüm getirmenin amaçlandığı bu çalışmada ulaşılan sonuçlar ve yapılabilir araştırmalarla ilgili öneriler şöyle özetlenebilir.

İlk matematiksel programlama teknikleri kadar eski olmamakla birlikte, otuz yıllık geçmişe sahip darboğaz atama modelleri ulaştırma ve üretim hattı dengeleme gibi iyi bilinen alanlarda uygulanagelmiştir. Endüstriyel örgütler çerçevesini çoğunlukla aşmayan bu uygulamaların yaygınlaştırılması sosyal örgütlerde de kullanılmasıyla hızlandırılabilecektir.

Genelde insan etmeninin baskın olduğu sosyal örgütlerde niceliksel yöntemlerin kullanılması, teknoloji baskın endüstriyel örgütlerdekinden daha güçtür. Çünkü insanın bağlı olduğu örgütteki üretkenliği, ölçülmesi olanaksız olan -örgüte ait olma duygusu gibi- bir takım unsurlarla ilgilidir. Anlaşılması ve çözümlenmesi çok güç bu unsurları kişiler, ölçüm anlamında değil ama onun bir yaklaşımı anlamında, tercihlerine yansıtmaktadırlar. Böylece insan baskın örgütlerin önemli planlama göstergelerinden biri örgüt bireylerinin tercihleri olmaktadır. En önemli insan baskın örgütlerden olan eğitim kurumlarından birinde uygulaması yapılan DAM'nin , tercih göstergesinin kullanımına yatkın diğer örgütlere de uygulanması mümkündür. Geliştirilen algoritma ve yazılan bilgisayar programı, benzeri uygulamalar için tekrar kolaylıkla düzenlenebilecek özelliktedir.

Öncelikli olarak konuyla ilgili yöneylem araştırmalarının geleneksel *"bilgi birikiminin kağıt üzerinde kalması ve uygulama olanağı bulunamaması"* endişesi, DAM'lerin tercih yönlü kullanımı benimsenir ise, bu modeller için azaltılmış olabilecektir. Çünkü görünen odur ki, DAM'ler için geliştirilen çözüm yöntemlerini tanıtan makale sayısı bunların uygulandığı alan türünden fazladır. Genelde tüm programlama tekniklerinin gelecekteki yerinin sağlamlığı bu endişeyi giderme oranında olacak gibi görünmektedir. Nitekim doğrusal programlamayı diğer programlamalar arasında sivrilten unsur da bu konudaki bilgi birikiminin fazlalığınan çok, uygulamacıları teşvik eden kullanım kolaylığı olsa gerekir.

Program yazılımında temel alt algoritma olarak kullanılan Gross algoritması, anlaşılması kolay olmakla birlikte programlanması diğerlerinden daha kolay değildir. Yanısıra, aramaktadır. Bu ise algoritmanın yavaş çalışan bir programı olması demektir. Gross algoritmasının yerine, diğer klasik DAM çözüm yöntemlerinden biri kullanılarak daha hızlı bir program elde edilebilir.

Uygulamadakı başarısını önceden kestirmek güç olmakla birlikte, bu ve benzeri problemler için tercih sayısı serbest bırakılabilir. Ancak bu durumda az tercih yapma eğilimindeki bireylerin algoritmanın işleyişi hakkında bilgili kılınmaları gerekir. Çünkü, sözgelimi iki tercih yapan bireyin istemediği tercihlerinden birinde olmaması için diğer bütün öğelerin iki veya birinci tercihlerine atanmış olmaları gerekir. Beş tercih yapan biri üçüncü tercihine atanmadıkça iki tercih yapan birinin -ki bu durumda üçüncü tercihteki ödevleri istemiyor demektir- bir veya ikinci tercihine atanma olanağı yoktur. Böyle bir tercih sayısının serbest bırakılması uygulaması, bazı anlaşmazlıklara neden olabilecektir. Bu anlaşmazlıkların aşılması halinde veya aşılması için tanıtılan programın birinci kısmı kullanılarak *"tavsiye edilen tercih sayısı"* gibi bir bilgi türetmek kişileri bundan haberdar etmek yerinde olacaktır.

Algoritmayı uygulamak üzere Basic programlama dili ile yazılan program, daha etkin dillerle denenebilir. Yanısıra kullanılan bilgisayarın teknik yapısı da programın çalışma hızını etkileyeceği gözönünde tutulmalıdır.

Genel DAM için benzeri çalışmalar yapılarak büyük çapta işgücü gereksinen örgütlere uygulanabilecektir. Bu örgütler kamu kuruluşları olabileceği gibi fabrikalar da olabilir.

Kamu kuruluşlarında şehir temelinde tercihler ve fabrikalar da ise atölye temelinde tercihler alınabilir. Her iki tür örgütlerde de problemin yapısı klasik DAP olmayıp grup atamalarının sözkonusu olduğuna dikkat edilmelidir.

DAP'lerinin yakın gelecekteki Darboğaz Programlama (Bottleneck Programming) başlığına kayarak, klasik çözüm yaklaşımlarından ayrı özgün çözüm yöntemlerine kavuşması mümkün gibi görünmektedir. Böyle bir oluşum bazı modelleme kavram ve tekniklerinin gelişiminin nedeni veya sonucu olabilir. Ancak, DAM'nin teknik köken olarak Darboğaz Programlama ve problem kökenli olarak Atama Problemleri üst başlığında ele alınmasının uygun ve zaman içinde kaçınılmaz olabileceği söylenebilir.

Atama modelleri üzerindeki duyarlılık çözümlemeleri henüz tam anlamıyla ele alınmış ve incelenmiş değildir. Bu alanın yeni araştırmalara konu olması, duyarlılık çözümlemelerindeki gelişmelere de bağlıdır. Başka çalışmalarda bu yönlü incelemelere gidilmesi uygun olacaktır.

YARARLANILAN KAYNAKLAR

BHATIA, H.L. : "Time Minimizing Assignment Problem", *SCIMA*, Vol. 6, No. 3, 1977.

BURKARD, R./ U. DERIGS : *Assignment and Matching Problem; Solution Methods with Fortran-Program*, Springer-Verlag, Berlin, 1980.

CARPANETO, G./ P.TOTH : "Algorithm for the Solution of the Bottleneck Assignment Problem". *Computing 27*, 1981.

DANTZIG, B.G. : *Linear Programming and Extensions*, The RAND Corporation, Princeton, N.J., 1963.

DERIGS, U./ U.ZIMMERMANN : "An Augmenting Path Method for Solving Linear Bottleneck Assignment Problems", *Computing 19*, 1978.

EDMONDS, J./ D.R.FULKERSON : "Bottle Neck Extrema", *Journal of Combinatoriel Theory*, Vol. 8, 1970.

FORD, L.R./ D.R.FULKERSON : *Flows in Networks*, The RAND Corporation, Princeton, N.J., 1974.

FRANCIS, R.L./ J.A.WHITE : *Facility Layout and Location An Analytical Approach*, Prentice-Hall, Inc. Englewood Cliffs, N.J., 1974.

FRIEZE, A.M. : "The Bottleneck Linear Programming", *Operational Research Quarterly*, Vol. 26, No. 4, 1975.

GARFINKEL, S.R./ G.L.NEMHAUSER : *Integer Programming*, John-Wiley Sons, Inc., New York, 1972.

GARFINKEL,R.S. : "An Improved Algorithm for the Bottleneck Assignment Problem", *Working Paper Series*, No. 7003, College of Business Administration, University of Rochecter, New York, 1970.

GARFINKEL, R.S./ M.R.RAO : "The Bottleneck Transportation Problem", *Naval Research Logistic Quartorly*, Vol. 18, 1971.

GILMORE, P.C. : "Optimal and Suboptimal Algorithms for the Quadratic Assignment Problem", *Journal of Social and Industrial Applications of Mathematies*, Vol. 10, No. 2, 1962.

GRABOWSKI, W. : "Problem of Transportation in Minimum Time", *Bulletin De L'Académie Polonaise Des Sciences*, Vol. 12, No. 2, 1964.

HADLEY, G. : *Linear Programming*, Addison-Wesley Publishing Inc., New York, 1963.

HAMMER, P.L. : "Communications On 'The Bottleneck Transportation Problem' and 'Some Remarks on the Time Transportation Problem' ". *Naval Research Logistics Quarterly*, Vol. 18, 1971.

ISERMANN, H. : "Linear Bottleneck Transportation Problems", *Asia-Pacific Journal of Operational Research*, Vol. 1, 1984.

KLEIN, M./ H.TAKAMORI : "Parallel Line Assignment Problems", *Management Science*, Vol. 19, No.1, 1972.

KUHN, H.W. : "The Hungarian Method for the Assignment Problem", *Naval Research Logistic Quarterly*, Vol. 2, No. 1, 1955.

LAWLER, E.L. : "The Quadratic Assignment Problem", *Management Science*, Vol. 9, No. 4, 1963.

PAGE, E.S. : "A Note on Assignment Problems", *Computational Journal*, Vol. 6, No. 3, 1963.

PLANE, D.R./ Jr.C.McMILLAN : *Discrete Optimization: Integer Programming and Network Analysis for Management Decisions*, Prentice-Hall, Inc., New Jersey, 1971.

PROLL, L.G. : "A Simple Method of Assigning Projects to Students", *Operational Research Quarterly*, Vol. 23, 1972.

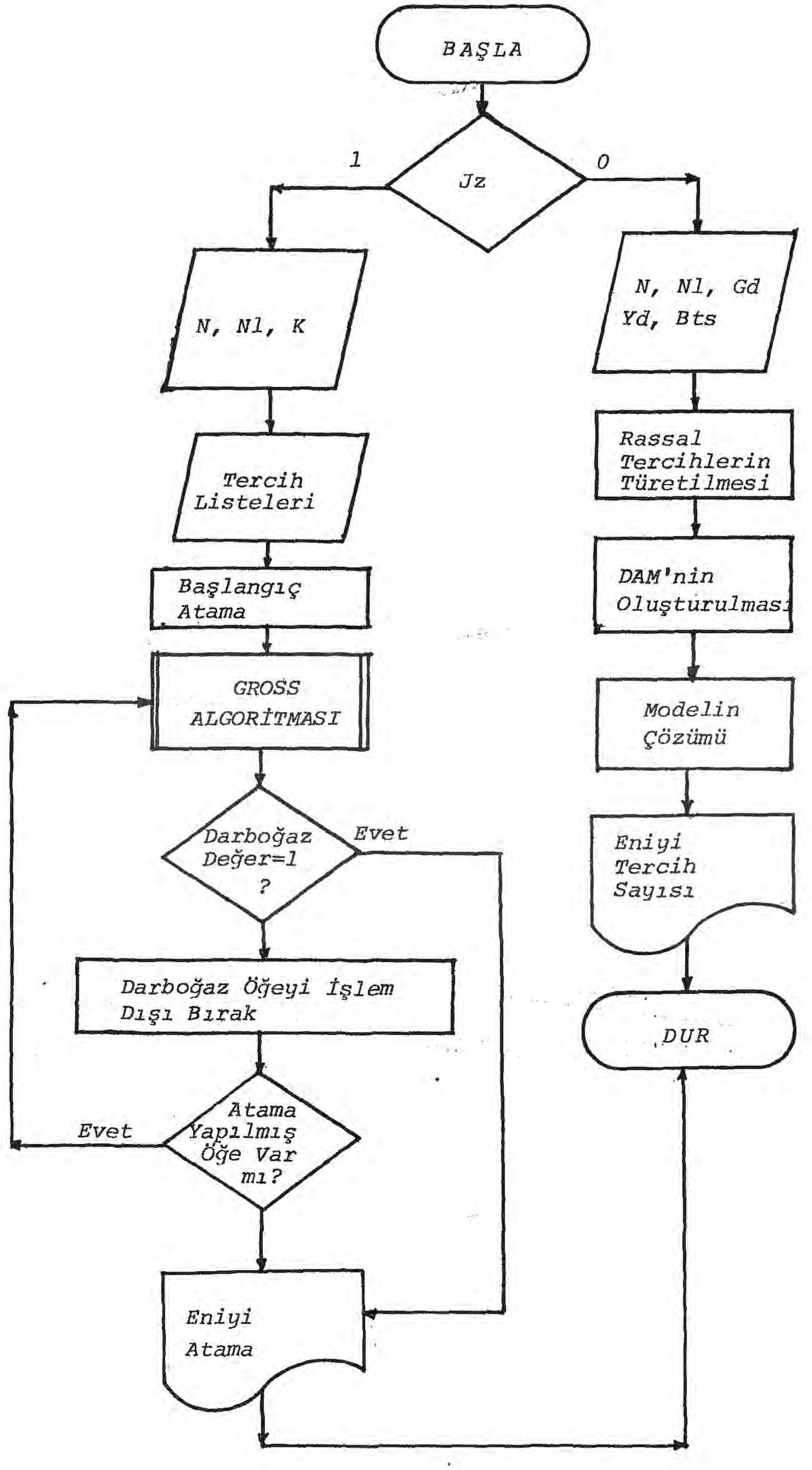
ROSS, G.T./ A.A. ZOLTNERS : "A Branch and Bound Algorithm for the Generalized Assignment Problem", *Mathematical Programming,* No. 8, 1975.

ROSS, G.T./ A.A. ZOLTNERS : "Weighted Assignment Models and Their Applications", *Management Science,* Vol. 25, No. 7, 1979.

SRINIVASAN, V./ G.L. THOMPSON : "Algorithms for Minimizing Total Cost, Bottleneck Time and Bottleneck Shipment in Transportation Problems", *Naval Research Logistic Quarterly,* Vol. 23, 1976.

SZWARC, W. : "Some Remarks on the Time Transportation Problem", *Naval Research Logistic Quarterly,* Vol. 18, 1971

K I : Bilgisayar Programı Akış Şeması.



EK II : Geliştirilen Algoritmanın Bilgisayar Programı.

```
10 OPEN 'PR:' AS FILE 1
20 DIM K(45,45),A(45,45),Y(1,45),Z(1,45),W(1,45),X(45,45),D(45),S(45),U(45)
30 ; CHR$(12)
40 ; YEL 'ENIYI TERCIH SAYISINI BULMAK ICIN'
50 ; CYA '0(SIFIR),'
60 ; YEL 'ENIYI ATAMAYI BULMAK ICIN'
70 ; GRN '1(BIR)'
80 ; YEL 'YAZINIZ'
90 INPUT Jz
100 IF Jz=0 ; #1 'ISTENEN PROGRAM:ENIYI TERCIH SAYISINI BULMAK'
110 IF Jz=1 ; #1 'ISTENEN PROGRAM:ENIYI ATAMAYI BULMAK'
120 ON Jz+1 GOTO 130,1300
130 INPUT 'PROJE SAYISI='N
140 ; #1 'PROJESAYISI=' N
150 INPUT 'OGRENCI SAYISI='N1
160 ; #1 'OGRENCISAYISI=' N1
170 INPUT 'YETERLILIK DUZEYI='Yd
180 ; #1 'YETERLILIK DUZEYI=' Yd
190 INPUT 'GUVENILIRLIK DUZEYI='Gd
200 ; #1 'GUVENILIRLIK DUZEYI=' Gd
210 Q=N/N1
220 ; #1 'PROJE/OGRENCIORANI=' Q
230 INPUT 'BASLANGIC TERCIH SAYISI='Bts
240 ; #1 'BASLANGIC TERCIH SAYISI=' Bts
250 FOR M1=Bts TO N
260   ; #1 'TERCIHSAYISI=' M1
270   K=M1/N
280   ; #1 'TERCIHORANI:K=M1/N=' K
290   At=0
300   FOR Ds1=1 TO Gd
310     ; BLU NWBG DBLE FLSH RED 'DOKUNMAYINIZ HESAP YAPIYOR'
320     REM ------RASSAL TERCIH MATRISI OLUSTURMA---
330     FOR I=1 TO N : FOR J=1 TO N
340         IF I>N1 K(I,J)=1 : GOTO 360
350         K(I,J)=M1+1
360     NEXT J : NEXT I
370     FOR T3=1 TO N1 : FOR T=1 TO M1
380         RANDOMIZE
390         X(T3,T)=INT((RND*N)+1)
400         Pr=0
410         FOR T5=1 TO T
420           RANDOMIZE
430           IF Pr>0 X(T3,T)=INT(RND*N+1) : GOTO 400
440           IF X(T3,T5)=X(T3,T) Pr=1
450         NEXT T5
460         Pr=0
470         K(T3,X(T3,T))=T
480     NEXT T : NEXT T3
490     REM ****************************
```

```
500 REM ----BASLANGIC ATAMA ------
510 FOR I=1 TO N : FOR J=1 TO N
520     IF I=J A(I,J)=1 ELSE A(I,J)=0
530     S(I)=I
540 NEXT J : NEXT I
550 REM *************************
560 REM ----DARBOGAZ ELEMAN BULMA-----
570 D(M1+1)=0
580 D=0
590 FOR I=1 TO N1 : IF S(I)()I GOTO 640
600   FOR J=1 TO N
610     IF A(I,J)=0 GOTO 640
620     IF K(I,J)(=D GOTO 640
630     D=K(I,J) : S=I : U=J
640 NEXT J : NEXT I
650 IF D(M1+1 GOTO 930
660 REM ------YORUNGE OLUSTURMA------
670 L=1
680 W(1,L)=U
690 R=0
700 R=R+1
710 IF R=N+1 GOTO 850
720 IF S(R)()R GOTO 700
730 IF A(R,W(1,L)))0 GOTO 700
740 IF K(R,W(1,L)))=D GOTO 700
750 FOR K2=1 TO L
760   IF Z(1,K2)=R GOTO 700
770 NEXT K2
780 Y(1,L)=K(R,W(1,L)) : Z(1,L)=R
790 M=0
800 M=M+1
810 IF M=N+1 GOTO 700
820 IF A(Z(1,L),M)=1 L=L+1 ELSE GOTO 800
830 W(1,L)=M
840 IF W(1,L)=U GOTO 1050 ELSE GOTO 690
850 IF (L)1) L=L-1 : GOTO 1100
860 S(S)=S+1 : D(M1+1)=D(M1+1)+1 : GOTO 580
870 FOR I=1 TO N : FOR J=1 TO N
880     IF I)N1 GOTO 920
890     IF A(I,J)=1 THEN 900 ELSE 910
900     IF K(I,J)=M1+1 THEN D(M1+1)=D(M1+1)+1 ELSE 910
910 NEXT J : NEXT I
920 ; BLU NWBG DBLE FLSH RED 'DOKUNMAYINIZ HESAP  YAPIYOR'
930 Rt=Rt+D(M1+1)
940 ; #1 'RT=' Rt
950 NEXT Ds1
960 Ott=1-(Rt/(Gd*N1))
970 ; #1 'TERCIH SAYISI=' M1 'OLASILIK=' Ott
980 IF Ott)=Yd GOTO 1020
990 Ott=0
```

```
1000 : #1 "***FIN******FIN*************"
1010 NEXT M1
1020 ; #1 CHR$(28,29) 'ENIYI TERCIH SAYISI=' M1
1030 ; #1 CHR$(28,29) 'OLASILIK DEGERI=' Ott
1040 STOP
1050 ; : ; MAG NWBG YEL FLSH 'YORUNGE BITTI,YENI ATAMA YAPILABILIR'
1060 REM YENI ATAMA
1070 FOR K4=1 TO L-1
1080   FOR I=1 TO N
1090     A(I,W(1,K4))=0
1100     A(Z(1,K4),W(1,K4))=1
1110   NEXT I
1120 NEXT K4
1130 L=0
1140 FOR I=1 TO N+1
1150   Y(1,I)=0 : Z(1,I)=0 : W(1,I)=0
1160 NEXT I
1170 GOTO 580
1180 P=Z(1,L) : R=0
1190 R=P+R
1200 R=R+1
1210 IF R=N+1 GOTO 850
1220 IF S(R)()R GOTO 1200
1230 IF A(R,W(1,L)))0 GOTO 1200
1240 IF K(R,W(1,L)))=D GOTO 1200
1250 FOR K3=1 TO L
1260   IF Z(1,K3)=R GOTO 1200
1270 NEXT K3
1280 Y(1,L)=K(R,W(1,L)) : Z(1,L)=R
1290 GOTO 790
1300 ; CHR$(12)
1310 INPUT 'PROJE SAYISI='N
1320 INPUT 'OGRENCI SAYISI='N1
1330 INPUT 'TERCIH SAYISI='M1
1340 ; #1 'PROJE SAYISI=' N
1350 ; #1 'OGRENCI SAYISI=' N1
1360 ; #1 'TERCIH SAYISI=' M1
1370 ; #1 'PROJE/OGRENCI ORANI=' N/N1
1380 ; #1 'TERCIH ORANI=' M1/N
1390 DIM K(N,N),A(N,N),Y(1,N+1),Z(1,N+1),W(1,N+1),X(N1,M1),D(M1+1),S(N),U(N)
1400 FOR I=1 TO N : FOR J=1 TO N
1410     IF I)N1 K(I,J)=1 : GOTO 1430
1420     K(I,J)=M1+1
1430 NEXT J : NEXT I
1440 FOR T3=1 TO N1 : FOR T=1 TO M1
1450     ; 'OGRENCI NO=';T3; : ; 'TERCIH NO=';T; : INPUT 'PROJE NO='X(T3,T)
1460     K(T3,X(T3,T))=T
1470 NEXT T : NEXT T3
1480 FOR I=1 TO N : FOR J=1 TO N
1490 ; #1 USING "##" K(I,J); : NEXT J : ; #1 : NEXT I
```

```
1500 FOR I=1 TO N : ; : FOR J=1 TO N
1510      IF I=J A(I,J)=1 ELSE A(I,J)=0
1520      S(I)=I
1530 NEXT J : ; : NEXT I
1540 D=0
1550 FOR I=1 TO N1 : IF S(I)<>I GOTO 1600
1560   FOR J=1 TO N
1570      IF A(I,J)=0 GOTO 1600
1580      IF K(I,J)<=D GOTO 1600
1590      D=K(I,J) : S=I : U=J
1600 NEXT J : NEXT I
1610 ; #1 'D=' D,'S=' S,'U=' U
1620 IF D=0 OR D=1 GOTO 1830
1630 L=1
1640 W(1,L)=U
1650 R=0
1660 R=R+1
1670 IF R=N+1 GOTO 1810
1680 IF S(R)<>R GOTO 1660
1690 IF A(R,W(1,L))>0 GOTO 1660
1700 IF K(R,W(1,L))>=D GOTO 1660
1710 FOR K2=1 TO L
1720   IF Z(1,K2)=R GOTO 1660
1730 NEXT K2
1740 Y(1,L)=K(R,W(1,L)) : Z(1,L)=R
1750 ; #1 'L=' L;'SATIR=' Z(1,L);'SUTUN=' W(1,L);'DEGER=' Y(1,L)
1760 M=0
1770 M=M+1
1780 IF A(Z(1,L),M)=1 L=L+1 ELSE GOTO 1770
1790 W(1,L)=M
1800 IF W(1,L)=U GOTO 1920 ELSE GOTO 1650
1810 IF L>1 L=L-1 : ; #1 'L2=' L;'SATIR2=' Z(1,L);'SUTUN2=' W(1,L);'DEGER2=' Y(1,L) : GOTO 2040
1820 S(S)=S+1 : GOTO 1540
1830 FOR I=1 TO N1 : FOR J=1 TO N
1840      IF A(I,J)=1 ; #1 'OGRENCI NO=' I,'PROJE NO=' J,'TERCIH NO=' K(I,J) : D(K(I,J))=D(K(I,J))+1
1850 NEXT J : NEXT I
1860 FOR R2=1 TO M1+1
1870    ; #1 'TERCIH NO=' R2,'OGRENCI SAYISI=' D(R2)
1880 NEXT R2
1890 ; #1 'MEVCUT COZUM ENIYI'
1900 ; #1 'D=' D,'S=' S,'U=' U
1910 STOP
1920 ; #1 'YORUNGE OLUSTU,YENI ATAMA YAPILABILIR'
1930 FOR K4=1 TO L-1
1940   FOR I=1 TO N
1950      A(I,W(1,K4))=0
1960      A(Z(1,K4),W(1,K4))=1
1970   NEXT I
1980 NEXT K4
1990 L=0
```

```
2000 FOR I=1 TO N+1
2010   Y(1,I)=0 : Z(1,I)=0 : W(1,I)=0
2020 NEXT I
2030 GOTO 1540
2040 P=Z(1,L) : R=0
2050 R=P+R
2060 R=R+1
2070 IF R=N+1 GOTO 1810
2080 IF S(R)<>R GOTO 2060
2090 IF A(R,W(1,L))>0 GOTO 2060
2100 IF K(R,W(1,L))=D GOTO 2060
2110 FOR K3=1 TO L
2120   IF Z(1,K3)=R GOTO 2060
2130 NEXT K3
2140 Y(1,L)=K(R,W(1,L)) : Z(1,L)=R
2150 GOTO 1760
```