Dokuz Eylül Üniversitesi
Sosyal Bilimler Enstitüsü Dergisi
Cilt 3, Sayı:2, 2001

PROJE SEÇİMİ VE KAYNAK PLANLAMASI İÇİN BİR ALGORİTMA

Nilgün MORALI[1] C. Cengiz ÇELİKOĞLU[2]

ÖZ

Kaynak tahsisi problemleri koşullara bağlı olarak bir doğrusal programlama modeli, tamsayılı programlama modeli ya da karma tamsayılı programlama modeliyle ifade edilir. Bu modellerde amaç toplam getirinin maksimizasyonudur. Bu amaca, kaynak ayrılan faaliyet sayısının maksimizasyonu şeklinde ikinci bir amaç eklendiğinde, problem amaç programlama teknikleriyle çözülebilir. Bu çalışmada problemin amaç programlamayla çözülmesi yerine kullanılmak üzere bir yaklaşık çözüm algoritması önerilmiştir.

Anahtar Sözcükler: Kaynak planlama, Karma tamsayılı programlama, Proje seçimi

AN ALGORITHM FOR PROJECT SELECTION AND RESOURCE PLANNING

ABSTRACT

The problems allocation of resources are expressed in a linear programming model, an integer programming model or a mixed integer programming model, varying with conditions. In these models, the aim is to maximize the total profit. When an aim of maximizing the number of activities for which resources are allocated is added to this aim, problem can be solved by intention programming techniques. In this piece of work, instead of solving the problem by using goal programming, an approximate solution algorithm is proposed.

Key Words: Resource planning, Mixed integer programming, Project selection

[1] Prof.Dr.; Dokuz Eylül Üniv. Fen-Edebiyat Fak., İstatistik Bölümü, Buca-İZMİR (nilgun.morali@deu.edu.tr) (Kaynaklar Kampusü 35160 Buca-İZMİR)

[2] Y.Doç.Dr.; Dokuz Eylül Üniv. Fen-Edebiyat Fak., İstatistik Bölümü, Buca-İZMİR (cengiz.celikoglu@deu.edu.tr) (Kaynaklar Kampusü 35160 Buca-İZMİR)

1. GİRİŞ

Çeşitli faaliyetlerin değişik düzeylerde gerçekleştirilebileceği ve kaynakların sınırlı olduğu durumlarda, her faaliyetin hangi düzeyde gerçekleştirileceğinin, dolayısıyla her faaliyete ne kadar kaynak ayrılacağının belirlenmesi, kaynak tahsisi problemi olarak adlandırılır. Böyle bir kaynak tahsisi probleminde amaç fonksiyonunu ve kısıtları belirleyen ilişkileri doğrusal olarak ifade etmek mümkün olursa, problem bir doğrusal programlama modeliyle formüle edilir ve kolaylıkla çözülür.

Ayrıca, çeşitli faaliyetlerin daha önceden belirlenmiş sabit miktarlarda kaynak kullanarak gerçekleştirilebileceği durumlarda, hangi faaliyetlere kaynak ayrılacağının hangilerine ayrılmayacağının belirlenmesi de kaynak tahsisi problemi olarak adlandırılır. Bu durumda problem genellikle bir 0-1 tamsayı programlama modeliyle ifade edilir ve dal-sınır algoritması ya da dinamik programlama çözüm tekniklerinden yararlanılarak çözülür.

Bu çalışmada kaynak tahsisi problemi yukarıdaki iki yaklaşımın bir birleşimi olarak tanımlanmıştır. $j = 1,2,...,N$ için p_j ile simgelenen N değişik faaliyet için en az l_j ve en çok u_j miktarda kaynağın gerektiği belirlenmiş olsun. Her faaliyetin sağlayacağı getiri (fayda ya da kazanç) b_j ve eldeki toplam kaynak B ile gösterilsin. Bu koşullarda toplam getiriyi maksimum kılmak üzere hangi faaliyetlerin gerçekleştirileceğinin ve bunlara ne kadar kaynak ayrılacağının belirlenmesi de bir kaynak tahsisi problemidir. p_j faaliyetine ayrılan kaynağı x_j karar değişkeniyle ve j faaliyetine kaynak ayırmamayı ya da ayırmayı y_j 0-1 tamsayı değişkeniyle göstermek üzere, problem bir karma tamsayı programlama modeliyle

$$\max Z = b_1x_1 + b_2x_2 + ... + b_Nx_N$$

$$x_1 + x_2 + ... + x_N \leq B$$

$$x_j \leq u_j \quad j = 1,2,...,N$$

$$x_j\,[1 - y_j] + [l_j - x_j]y_j \leq 0 \qquad j = 1,2,...,N$$

$$x_j \geq 0 \quad y_j = 0 \text{ veya } 1 \qquad j = 1,2,...,N$$

biçiminde formüle edilebilir. Bu problem de karma tamsayı programlama problemleri için varolan çözüm tekniklerinden biriyle çözülebilir; ancak, alt sınıra bağlı kısıtın doğrusal olmaması nedeniyle çözümün kolaylıkla elde edilmesi mümkün değildir.

Yukarıda tanımlanan problem gerçek hayatta bir işletmenin değişik departmanlarından gelen yatırım projelerinin desteklenmesi ya da bir üniversitede araştırma fonundan çeşitli projelere kaynak dağıtılması biçiminde karşımıza çıkabilir. Böyle durumlarda ise karar vericilerin genellikle olabildiğince fazla sayıda projeye olumlu yanıt vermek gibi bir amaçları daha vardır. Bu amaç matematiksel olarak

$$\max Z_2 = y_1 + y_2 + ... + y_N$$

şeklinde ifade edilir. Bu amacın da yukarıdaki modele eklenmesiyle problemin çözümü amaç programlama teknikleriyle mümkün olur.

Bu çalışmada yukarıda tanımlanan çok amaçlı karar modeli için bir yaklaşık çözüm algoritması geliştirilmiş ve örnek bir problem üzerinde elde edilen yaklaşık optimal çözüm değişik bütçe sınırları için tartışılmıştır.

2. YÖNTEM

Karar vericinin kolaylıkla benimseyebileceği bir çözüm elde edilmesi amacıyla aşağıda bir yaklaşık çözüm algoritması önerilmiştir. Başlangıçta her proje için desteklenme oranı bu projeye ayrılan kaynağın gerekli en büyük kaynak miktarına oranı olarak (x_j/u_j) şeklinde tanımlanmış ve "projelerin desteklenme oranlarının getirileriyle orantılı olmaları" ilkesi kabul edilmiştir. Bu ilke daha açık olarak

$$\frac{x_j / u_j}{x_i / u_i} = \frac{b_j}{b_i} \quad \text{veya} \quad \frac{x_j / u_j}{b_j} = \frac{x_i / u_i}{b_i}$$

biçiminde yazılır. Bu koşulun tüm projeler için geçerli olması

$$\frac{x_1}{b_1 u_1} = \frac{x_2}{b_2 u_2} = \ldots = \frac{x_N}{b_N u_N}$$

şeklinde N bilinmeyenli N-1 denklemli bir sistem oluşturur. Bu sistemde x_1 sabit kabul edildiğinde

$$x_j = \frac{b_j u_j}{b_1 u_1} x_1 \qquad j = 2,3,\ldots,N$$

elde edilir. Ayrıca kaynak kısıtı da eşitlik biçiminde dikkate alındığında

$$x_1 + x_2 + ... + x_N = B$$

denklemi

$$\frac{b_1 u_1 + b_2 u_2 + \ldots + b_N u_N}{b_1 u_1} x_1 = B$$

biçimini alır ve her x_j için çözüm

$$x_j = \frac{B}{b_1 u_1 + b_2 u_2 + \ldots + b_N u_N} b_j u_j \qquad j = 1,2,\ldots,N$$

bulunur. Bu çözümde

$$l_j \leq x_j \leq u_j \qquad j = 1, 2,..., N$$

kısıtı sağlanıyorsa, bütün projelerin desteklenmesine karar verilmiş ve ayrılacak kaynak miktarları belirlenmiş olur. Buna karşın, kısıtlardan en az biri sağlanmazsa, bulunan çözümün geçerli çözüm olmadığı anlaşılır. Bu durumda öncelikle desteklenecek projelerin seçilmesi ve sonra seçilen projelere kaynak tahsisi şeklinde iki aşamadan oluşan bir algoritma aşağıdaki gibi uygulanabilir.

Eğer bütçe tüm projeler için minimum kaynak gereksinimini karşılamaya yetiyorsa, tüm projelerin desteklenmesine karar verilir ve algoritmanın doğrudan ikinci aşaması uygulanır.

1. AŞAMA : *Desteklenecek projelerin seçilmesi*

j projesinin desteklenmesi için herşeyden önce $x_j \geq l_j$ olması gerekir. Daha açık olarak

$$\frac{B}{b_1u_1 + b_2u_2 + \ldots + b_N u_N} b_j u_j \geq l_j \quad \text{ya da} \quad C_1 = \frac{b_1u_1 + b_2u_2 + \ldots + b_N u_N}{B} \leq \frac{b_j u_j}{l_j}$$

olmalıdır. Bu koşulu sağlayan n proje olduğunu varsayalım. Bu n proje için gerekli kaynakların üst sınırları toplamı bütçeyi aşıyorsa ikinci aşamaya geçilir. Aksi halde, eldeki projeler için gerekli kaynakların alt sınırları toplamı bütçeden küçükse, kalan tüm projelerin desteklenmesine karar verilir ve ikinci aşamaya geçilir; büyükse $\min_j \{\frac{b_j u_j}{l_j}\}$ olan proje sistemden çıkarılarak yeniden x_j değerleri hesaplanır. $\min_j \{\frac{b_j u_j}{l_j}\}$ olan birden fazla proje varsa, bunlar arasından en az kaynak gereksinimi en büyük olan proje sistemden çıkarılır. Bu işlemlere ikinci aşamaya geçme koşulu sağlanana kadar devam edilir.

2. AŞAMA : *Her proje için ne kadar kaynak ayrılacağının belirlenmesi*

Seçilen projeler en az alt sınırda destekleneceğine göre j projesi için ayrılan kaynağın alt sınırın üstünde kalan kısmını w_j karar değişkeniyle gösterirsek $x_j = l_j + w_j$ (j = 1,2,...,n) olur. Bu durumda desteklenme oranı $w_j/(u_j - l_j)$ ile tanımlanır. Projelerin desteklenme oranlarının getirileriyle orantılı olmaları ilkesine göre

$$\frac{w_1}{b_1(u_1 - l_1)} = \frac{w_2}{b_2(u_2 - l_2)} = \ldots = \frac{w_n}{b_n(u_n - l_n)}$$

elde edilir ve bütçe kısıtı

$$B = x_1 + x_2 + ... + x_n = (l_1\text{-}w_1) + (l_2\text{-}w_2) + ... + (l_n\text{-}w_n)$$

biçimini alır. Sonuç olarak çözüm

$$w_j = \frac{B - (l_1 + l_2 + \ldots + l_n)}{b_1(u_1 - l_1) + b_2(u_2 - l_2) + \ldots + b_n(u_n - l_n)} b_j(u_j - l_j) \qquad j = 1,2,\ldots,n$$

formülüyle belirlenir. Bu çözümün geçerli olması için $w_j \leq u_j - l_j$ olması gerekir. $w_j \geq u_j - l_j$ ise ya da daha açık olarak

$$C_2 = \frac{b_1(u_1 - l_1) + b_2(u_2 - l_2) + ... + b_n(u_n - l_n)}{B - (l_1 + l_2 + \ldots + l_n)} \leq b_j$$

koşulunu sağlayan proje varsa, getirisi en yüksek olan proje için $w_j = u_j - l_j$ (kısaca $x_j = u_j$) alınır ve diğer projeler için kalan bütçe yeniden paylaştırılır. Böyle bir durum ortaya çıkmazsa, elde edilen çözüm yaklaşık optimal çözümdür.

3. ÖRNEK

Yukarıda önerilen algoritmanın nasıl işlediğini incelemek amacıyla on proje önerisinin dikkate alındığı bir kaynak tahsisi problemi geliştirilmiş ve projelerin getirileri, gerekli kaynak miktarları ve değişik bütçe kısıtları için yaklaşık optimal çözümler Tablo 1'de verilmiştir. Bu problemde bütün projelerin desteklenebilmesi için gerekli minimum bütçe kısıtlaması 5720(x10^6) liradır. Bu nedenle bütçe kısıtlamaları sırasıyla 4000, 5000, 6500 seçilmiştir.

Tablo1. Örnek Problem Veri ve Sonuçları

Proje J	b_j	l_j (x10^6)	u_j(x10^6)	$b_j u_j/l_j$	B_1=4000(x10^6) İçin çözüm	B_2=5000(x10^6) için çözüm	B_3=6500(x10^6) için çözüm
1	1,0	1000	1250	1,25	1132	1064	1160
2	0,9	450	700	1,40	569	508	594
3	0,8	600	660	0,88	625	612	631
4	0,7	700	900	0,90	774	736	790
5	0,6	1200	1500	0,75	0	1246	1316
6	0,5	450	720	0,80	521	485	537
7	0,4	320	600	0,75	379	349	392
8	0,3	300	350	0,35	0	0	310
9	0,2	500	1000	0,40	0	0	564
10	0,1	200	300	0,15	0	0	206
Toplam		5720	7980		4000	5000	6500

B_3=6500 olduğunda bütün projeleri en az alt sınırda desteklemek mümkün olduğu için doğrudan 2. aşamaya geçilir. C_2=1,56 hesaplanır ve bu değer bütün b_j'lerden büyük olduğu için x_j değerleri kolaylıkla bulunur.

B_2=5000 olduğunda, C_1=0,97 bulunur; 1. ve 2. proje seçilir; $u_1+u_2<5000$ olduğundan 10. proje sistemden çıkarılarak yeniden C_1 değeri hesaplanır. Benzer şekilde 8. ve 9. proje de sistemden çıkarıldıktan sonra, eldeki bütçe kalan yedi projenin en azından alt sınırda desteklenmesine yettiği için 2. aşamaya geçilir. C_2=3,89 bulunduğundan x_j değerleri kolaylıkla elde edilir.

B_1=4000 olduğunda, C_1=1,22 bulunur; yine 1. ve 2. proje seçilir; $u_1+u_2<4000$ olduğundan 10. proje sistemden çıkarılarak yeniden C_1 değeri hesaplanır. Benzer şekilde 8. ve 9. proje de sistemden çıkarıldıktan sonra, 5. veya 7. projelerden birinin daha sistemden çıkarılması gerekir. Bu durumda en az kaynak gereksinimi daha yüksek olan 5. projenin sistemden çıkarılması tercih edilir. Eldeki bütçe kalan altı projenin en azından alt sınırda desteklenmesine yettiği için 2. aşamaya geçilir. C_2=1,89 bulunduğundan x_j değerleri hesaplanır.

4. SONUÇ

Bu çalışmada tanımlandığı biçimiyle karşılaşılan bir kaynak tahsisi probleminde projeler az sayıdaysa, varolan matematiksel programlama teknikleri ve eldeki bilgisayar olanaklarından yararlanarak problemin kesin çözümünü elde etmek mümkün olabilir. Buna karşın, proje sayısı bu olanakları kullanmaya izin vermeyecek kadar çok olduğunda, yukarıda verilen algoritma aracılığıyla bir yaklaşık çözüm kolaylıkla bulunabilir.

KAYNAKÇA

Acar A. (1989). *Linear Programming for Managerial Decisions*, ODTÜ, Ankara.

Moralı N. (1994). "Using AHP in Prioritization of the Procurement Proposals in Universities", *Proceedings of the 3rd International Symposium on the Analytic Hierarchy Process*, Wasington D.C., ss. 321-330.

Kwak N.K. ve Diminnie, C.B. (1987). "A Goal Programming Model for Allocating Operating Budgets of Academic Units", *Socio-Economic Planning Sciences*, c. 21, ss. 331-339.