

İleri Yöneylem Araştırması Uygulamaları Tam Sayılı Programlama

Dr. Özgür Kabak

2020-2021 Güz

-
- ▶ Gerçek hayattaki bir çok problem
 - ▶ tam sayılı değişkenlerin ve
 - ▶ doğrusal kısıt ve amaç fonksiyonları ile
 - ▶ modellenenabilir.

 - ▶ Tamamen tamsayılı programlar (Pure integer programming)
 - ▶ Karma tamsayılı programlar (Mixed integer programming)
 - ▶ 0-1 (ikili) tamsayılı programlar (Binary integer programming)



Uygulama alanları

- ▶ Kesikli girdi ve çıktısı olan problemler
- ▶ Mantıksal koşulların yer aldığı problemler
- ▶ Birleş (combinatorial) problemler
- ▶ Doğrusal olmayan problemler
- ▶ Ağ (network) problemleri



Tam sayılı programlama

Çözüm yöntemleri

- ▶ Kesme Düzlemi yöntemleri
- ▶ Sayma (Enumerative) yöntemleri
- ▶ Pseudo-Boolean yöntemleri
- ▶ Dal Sınır yöntemleri



Kesikli girdi ve çıktısı olan problemler

- ▶ Tam sayı değeri alan karar değişkenleri
- ▶ Gösterge değişkenler
 - ▶ Sabit maliyet içeren problemler
 - ▶ Şartlı durumlar (Ör. A üretilirse B de üretilmelidir.)



Mantıksal koşulların yer aldığı problemler

- ▶ $\vee$: veya
- ▶ $\cdot$: ve
- ▶ $\sim$: değil
- ▶ $\rightarrow$: ise
- ▶ $\leftrightarrow$: ancak ve ancak



Mantıksal koşulların yer aldığı problemler

► Veya kısıtları

- Verilen eşitsizliklerden en az birinin sağlanması isteniyorsa;

- $f(x_1, x_1, \dots, x_n) \leq 0$
 - $g(x_1, x_1, \dots, x_n) \leq 0$

- Modele aşağıdaki kısıtlar ilave edilir:

- $f(x_1, x_1, \dots, x_n) \leq My$
 - $g(x_1, x_1, \dots, x_n) \leq M(1 - y)$
 - $y = 0$ veya 1

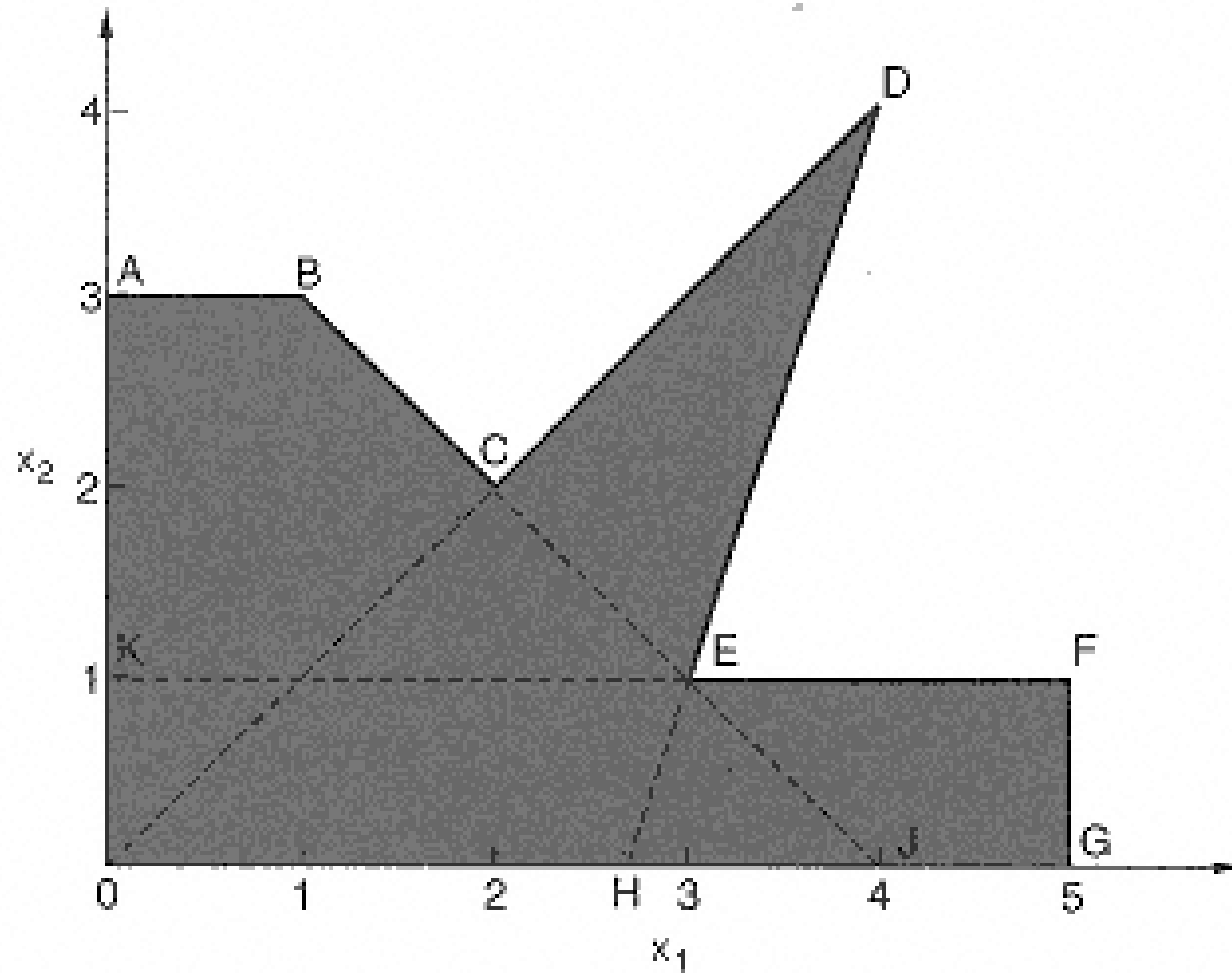
► Eğer ise kısıtları

- Eğer $f(x_1, x_1, \dots, x_n) > 0$ ise $g(x_1, x_1, \dots, x_n) \geq 0$ koşulu da sağlanmak isteniyorsa modele aşağıdaki kısıtlar ilave edilir:

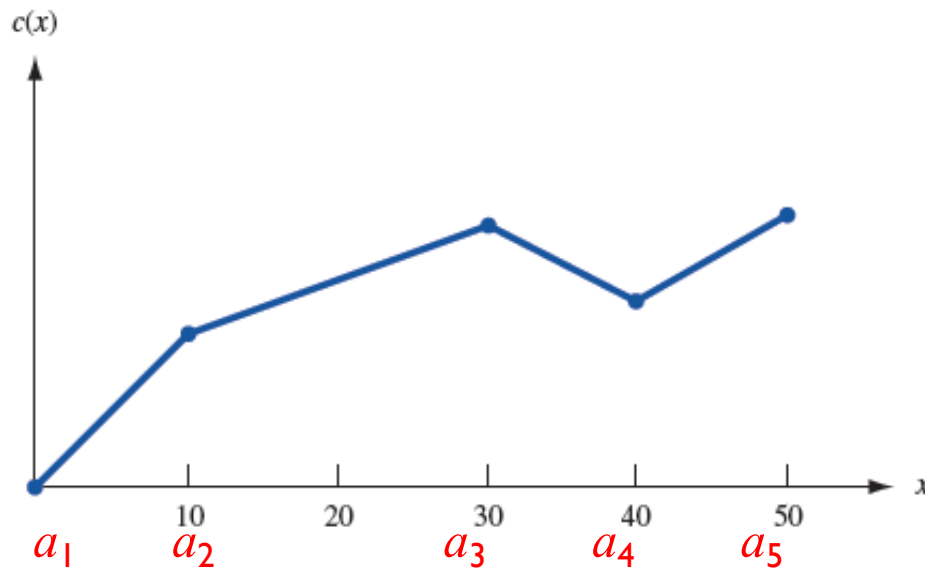
- $-g(x_1, x_1, \dots, x_n) \leq My$
 - $f(x_1, x_1, \dots, x_n) \leq M(1 - y)$
 - $y = 0$ veya 1



Konveks olmayan bölgeler



Sürekli Parçalı fonksiyonların modellenmesi



$$a_1 < a_2 < \dots < a_k$$

$f(x), (a_i, f(a_i))$ $i = 1, \dots, k$ noktaları ile tanımlanmış bir sürekli parçalı fonksiyon

$$x = \sum_{i=1}^k \lambda_i a_i, \quad \sum_{i=1}^k \lambda_i = 1$$

$$f(x) = \sum_{i=1}^k \lambda_i f(a_i)$$

$$\text{Min } \sum_{i=1}^k \lambda_i f(a_i)$$

Öyle ki; $\sum_{i=1}^k \lambda_i = 1,$

- ▶ $\lambda_1 \leq y_1,$
- ▶ $\lambda_i \leq y_{i-1} + y_i \quad i = 2, \dots, k-1,$
- ▶ $\lambda_k \leq y_{k-1}$
- ▶ $\sum_{i=1}^{k-1} y_i = 1$
- ▶ $\lambda_i \geq 0$
- ▶ $y_i \in \{0,1\}$

- ▶ x 'in birbirini takip eden λ 'lar ile ifade edilebilmesi için $y_i, i = 1, \dots, k-1, 0-1$ değişkenleri tanımlanır.

Ağ (network) problemleri

- ▶ En kısa yol problemi
- ▶ En küçük maliyetli akış problemi
- ▶ En büyük akış problemi
- ▶ Proje Planlama - CPM-PERT

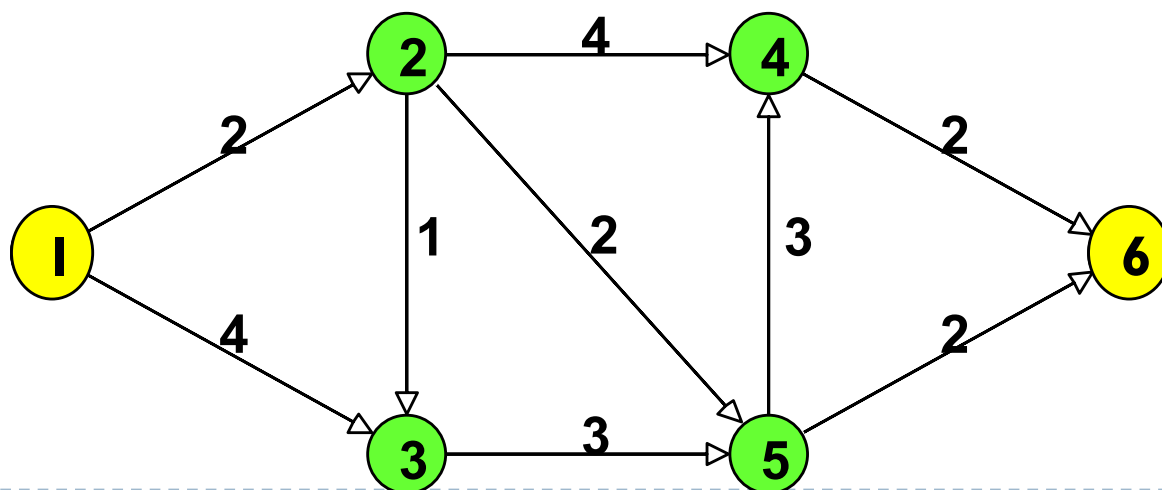


En kısa yol problemi

$$\text{Minimize} \quad \sum_{i=1}^m \sum_{j=1}^m c_{ij} x_{ij}$$

$$\text{Subject to} \quad \sum_{j=1}^m x_{ij} - \sum_{k=1}^m x_{ki} = \begin{cases} 1 & \text{if } i = 1 \\ 0 & \text{if } i \neq 1 \text{ or } m \\ -1 & \text{if } i = m \end{cases}$$

$$x_{ij} = 0 \text{ or } 1 \quad i, j = 1, 2, \dots, m$$



Özel Tamsayılı Programlama Modelleri

► Küme kapsama problemi

$$\begin{array}{ll}\text{Minimize} & \delta_1 + \delta_2 + \delta_3 + \delta_4 + \delta_5 + \delta_6 \\ \text{subject to} & \delta_1 + \delta_2 \quad \quad + \delta_5 \geq 1, \\ & \delta_1 \quad \quad + \delta_3 \quad \quad \geq 1, \\ & \delta_2 \quad \quad + \delta_4 \quad \quad \geq 1, \\ & \delta_3 \quad \quad + \delta_6 \geq 1, \\ & \delta_2 + \delta_3 \quad \quad + \delta_6 \geq 1.\end{array}$$

► Küme paketleme Problemi

$$\begin{array}{ll}\text{Maximize} & \delta_1 + \delta_2 + \delta_3 + \delta_4 + \delta_5 \\ \text{subject to} & \delta_1 + \delta_2 \quad \quad \leq 1, \\ & \delta_1 \quad \quad + \delta_3 \quad \quad + \delta_5 \leq 1, \\ & \delta_2 \quad \quad + \delta_4 + \delta_5 \leq 1, \\ & \delta_3 \quad \quad \leq 1, \\ & \delta_4 \quad \quad \leq 1, \\ & \delta_4 + \delta_5 \leq 1.\end{array}$$

Özel Tamsayılı Programlama Modelleri

► Küme Bölüntüleme Problemi

$$\delta_1 + \delta_2 + \quad + \delta_5 = 1,$$

$$\delta_1 \quad + \delta_3 = 1,$$

$$\delta_2 \quad + \delta_4 = 1,$$

$$\delta_3 \quad + \delta_6 = 1,$$

$$\delta_2 + \delta_3 \quad + \delta_6 = 1.$$

► Sırt çantası Problemi

$$\begin{array}{ll} \text{Maximize} & p_1\gamma_1 + p_2\gamma_2 + \cdots + p_n\gamma_n \\ \text{subject to} & a_1\gamma_1 + a_2\gamma_2 + \cdots + a_n\gamma_n \leq b, \end{array}$$



Gezgin Satıcı Problemi

- ▶ $\min \sum_i \sum_j c_{ij} x_{ij}$
- ▶ öyle ki; $\sum_i x_{ij} = 1 \quad \forall j$
- ▶ $\sum_j x_{ij} = 1 \quad \forall i$
- ▶ $u_i - u_j + Nx_{ij} \leq N - 1 \quad i \neq j; i, j > 1$
- ▶ $x_{ij} = 0 \text{ veya } 1 \quad \forall (i, j)$
- ▶ $u_i \geq 0 \quad \forall i$



Problem Karmaşıklığı

- ▶ **P problemler:**

- ▶ Deterministik Turing makinası ile polinomiyal zamanda çözülebilen karar problemleri

- ▶ **NP (nondeterministic polynomial time):**

- ▶ verilen bir çözümün doğru olup olmadığının polinomial zamanda doğrulanabildiği problemler.

- ▶ **NP-zor (NP-hard)**

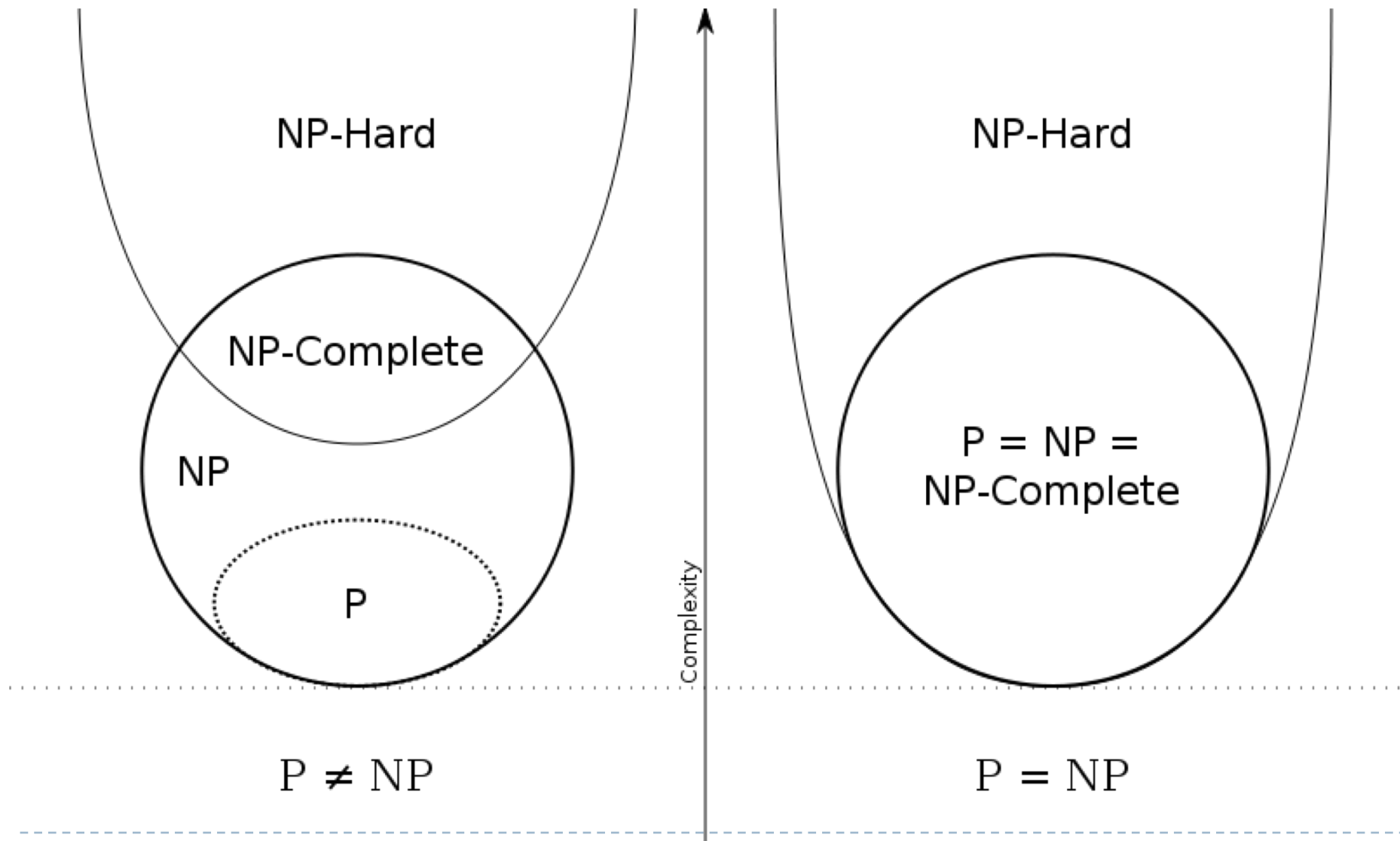
- ▶ En az NP'de en zor problemler kadar zor olan problem sınıfları

- ▶ **NP-tam (NP-complete)**

- ▶ NP'deki en zor problemleri içeren problem sınıfı. Polinom zamanda çözülemeyen, polinom zamanda doğrulanabilen problemler.



Problem Karmaşıklığı



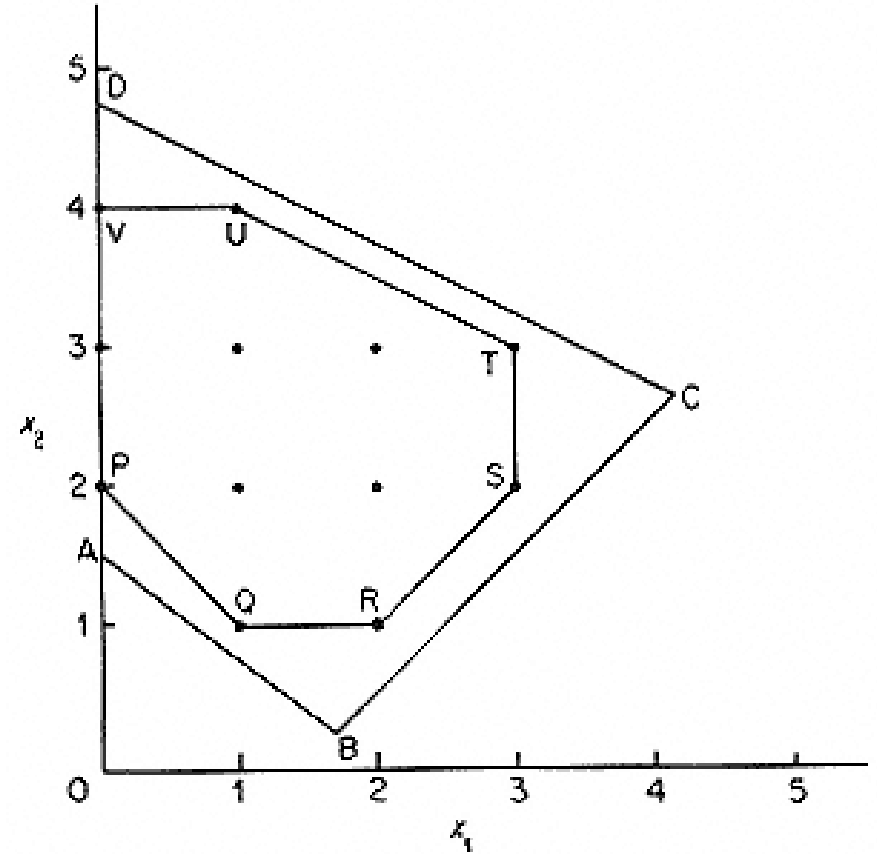
Tam sayılı programlama

İyi formülasyon nasıl yapılır?

- ▶ Doğrusal programlamadaki kurallar burada da geçerli.
- ▶ Modeldeki değişken sayısı?
 - ▶ 0-1 değişken sayısı çözme zorluğunu doğrudan etkilemez
 - ▶ Tam sayılı değişkenler 0-1 değişkene nasıl çevrilir?
 - ▶ Ekstra değişken eklemek çözümü kolaylaştırır mı?
- ▶ Modeldeki kısıt sayısı



- ▶ Modeldeki kısıt sayısı
 - ▶ DP gevşetme
 - ▶ Dışbükey örtü (Convex hull)
- ▶ Dışbükey örtüyü bulmak zor olsa da bazı türdeki problemler için
 - ▶ TP formülasyonu halihazırda dışbükey örtü olabilir
 - ▶ Problem kolayca yeniden formüle edilerek dışbükey örtü elde edilebilir
 - ▶ Problem dışbükey örtüye çok yakın bir DP'ye dönüştürülebilir

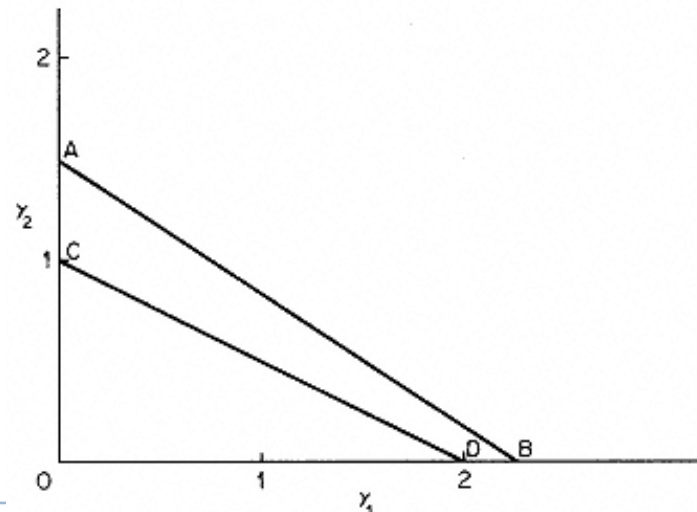


Tamsayılı programlama modelini basitleştirme

► Sınırların daraltılması

$$\begin{array}{ll}\text{Minimize} & 5\delta_1 + 7\delta_2 + 10\delta_3 + 3\delta_4 + \delta_5 \\ \text{subject to} & \delta_1 - 3\delta_2 + 5\delta_3 + \delta_4 - \delta_5 \geq 2, \quad (\text{R1}) \\ & -2\delta_1 + 6\delta_2 - 3\delta_3 - 2\delta_4 + 2\delta_5 \geq 0, \quad (\text{R2}) \\ & -\delta_2 + 2\delta_3 - 2\delta_4 - \delta_5 \geq 1. \quad (\text{R3})\end{array}$$

► Bir kısıt yerine başka bir kısıt ekleme



Tamsayılı programlama modelini basitleştirme

- Bir kısıt yerine birden çok kısıt ekleme

$$3\delta_1 + 3\delta_2 - 2\delta_3 + 2\delta_4 + 2\delta_5 \leq 4. \quad (8)$$



$$\delta_1 + \delta_2 - \delta_3 + \delta_4 \leq 1, \quad (9)$$

$$\delta_1 + \delta_2 - \delta_3 + \delta_5 \leq 1, \quad (10)$$

$$\delta_1 + \delta_2 + \delta_4 + \delta_5 \leq 2, \quad (11)$$

$$2\delta_1 + \delta_2 - \delta_3 + \delta_4 + \delta_5 \leq 2, \quad (12)$$

$$\delta_1 + 2\delta_2 - \delta_3 + \delta_4 + \delta_5 \leq 2, \quad (13)$$



Tamsayılı programlama ne zaman ve nasıl kullanılmalı?

- ▶ Bahsedilen uygulama alanları geçerli ise TP kullanımını değerlendirmek gerekir:
 - ▶ Kesikli girdi ve çıktısı olan problemler
 - ▶ Mantıksal koşulların yer aldığı problemler
 - ▶ Birleş (combinatorial) problemler
 - ▶ Doğrusal olmayan problemler
 - ▶ Ağ (network) problemleri
- ▶ Problemin potansiyel büyüklüğü değerlendirilmelidir:
 - ▶ Eğer tamsayılı değişken sayısı birkaç yüzden fazla ise ve problemin özel bir yapısı yok ise TP'nin hesaplama maliyeti çok yüksek olacaktır.
 - ▶ Önerilecek TP'nin özel yapısı olup olmadığı veya özel bir yapıya dönüştürülüp dönüştürülemeyeceği araştırılmalıdır.
- ▶ Model, problemin tamamı yerine küçük bir örnek üzerinde denenmelidir.
- ▶ Eğer problemin TP ile çözümü zor ise uygun sezgisel yaklaşımlar araştırılmalıdır.
- ▶ Model çözümünde akıllıca çözüm stratejileri geliştirilebilir. Bunun için gerçek problem konusunda uzman birinden destek alınabilir.
- ▶ TP modelleme ve çözümü konusunda çalışmalar artarak daha büyük ve karmaşık problemlerin çözümü mümkün olmaktadır.

