

## Doğrusal Olmayan Kısıtsız Fonksiyonların Tümel Minimum Değerlerinin Uyarlanmış Tavlama Benzetimi Yöntemi İle Elde Edilmesi

Ayşen APAYDIN, Özlem TÜRKŞEN\*

Ankara Üniversitesi, Fen Fakültesi, İstatistik Bölümü, 06100, Tandoğan, Ankara

**Özet:** Fen, mühendislik ve ekonomi alanında karşılaşılan bir çok problem, tümel en iyi çözümleri elde edilmek istenen optimizasyon problemi olarak ele alınabilir. Gerçek dünyada karşılaşılan problemler genellikle, doğrusal olmayan ve stokastik yapıda olduğu için, çözümler problemlere uygun, doğrusal olmayan fonksiyonel yapıların tanımlanması ile mümkün olacaktır. Bu problemler çok sayıda yerel minimum nokta içereceğinden, doğrusal olmayan optimizasyon yöntemleri ile her zaman yeterli güvenilirlikte ve etkinlikte çözümler elde edilemeyebilir. Karmaşık optimizasyon problemlerinin tümel çözümlerinin elde edilmesinde, metasezgisel yöntemler alternatif yöntemler olarak önerilebilir. Çalışmada, son yıllarda oldukça sık kullanılan bir metasezgisel yöntem olan Tavlama Benzetimi Yöntemi hakkında bilgi verilerek, Uyarlanmış Tavlama Benzetimi Yöntemi' nin algoritmik adımları tanımlanmıştır. Tanımlanan algoritma ile birden fazla yerel minimuma sahip doğrusal olmayan test fonksiyonlarının tümel minimum değerleri elde edilmeye çalışılmıştır. Kullanılan algoritmanın yerel minimum tuzaklarından kurtulup, test fonksiyonlarının tümel minimum değerlerine oldukça yakın sonuçlar verdiği gözlenmiştir.

**Anahtar Kelimeler:** Doğrusal olmayan fonksiyonlar, tümel minimum, Tavlama Benzetimi Yöntemi, Uyarlanmış Tavlama Benzetimi Yöntemi.

## Obtaining The Global Minimum Values Of Nonlinear Unconstrained Functions by The Adaptive Simulated Annealing Method

**Abstract:** Many of the problems in the field of science, engineering and economy can be considered as an optimization problems of which the global solutions are wanted to be achieved. Because the problems encountered in the real world have generally non-linear and stochastic structure, the solution of the problems shall be possible by the definition of the appropriate and non-linear functional structures. The non-linear optimization methods may not always have the sufficient reliability and efficiency in achieving the best solution. In achieving the global solutions for the complex optimization methods, metaheuristic methods can be suggested as alternative methods. In this study, the information is given about the Simulated Annealing Method which is one of the heuristic method has been used for the last decade and the algorithmic steps relating to the Adaptive Simulated Annealing Method is defined. The global minimum values of the nonlinear functions which have more than one local minimum tried to get by formed algorithm. It has been observed that the used algorithm escaped from local minimum traps and got considerably close results to the global minimum values of the test functions.

**Keywords:** Nonlinear functions, global minimum, Simulated Annealing Method, Adaptive Simulated Annealing Method

\* E-mail: turksen@science.ankara.edu.tr

## 1. Giriş

Genel olarak bir çok alanda karşılaşılan modelleme ve kestirim problemleri optimizasyon problemi olarak ele alınabilir. Fen ve mühendislik alanında karşılaşılan problemlerin çoğu, doğrusal olmayan optimizasyon problemi olarak düşünülebilir. Bu problemler, tümel en iyi çözümlere gereksinim duyarlar. Karşılaşılan bir çok optimizasyon problemi çok sayıda yerel minimum içerdiğinden, bilinen doğrusal olmayan optimizasyon yöntemlerinin etkinliğinin ve güvenilirliğinin daha düşük olduğu gözlenmiştir [6].

Son yıllarda zor optimizasyon problemlerinin çözümü için Tavlama Benzetimi (Simulated Annealing) adı altında rasgele komşuluk aramalarına dayalı bir optimizasyon yöntemi geliştirilmiştir. Bu yöntem, çok sayıda yerel minimuma sahip optimizasyon problemlerinin tümel minimumunu bulmada kullanılan yöntemlerden biridir [15]. İlgilenilen fonksiyonun rasgele hesaplamalarına dayalı bu metasezgisel yöntem, fonksiyonun tüm yüzeyini inceler ve aşağı-yukarı yönlü hareketlerle yerel minimumlar arası geçişlere izin vererek yerel minimum tuzaklarından kurtulup, doğrusal olmayan fonksiyonların tümel minimumunu bulmaya çalışır [6]. Corana, Marchesi, Martini ve Ridella (1987), sürekli değişkenli fonksiyonlar için geliştirilmiş bir Tavlama Benzetimi Algoritması hazırlamışlar, bu algoritmayı 2 ve 4 boyutlu Rosenbrock fonksiyonları ile 2, 4, 10 boyutlu çoklu minimuma sahip fonksiyonlar üzerinde uygulamışlardır [6]. Tavlama Benzetimi Yöntemi' ni kullanarak elde ettikleri sonuçları, Nelder-Mead Simpleks, Uyarlanmış Rasgele Arama yöntemlerini kullanarak elde ettikleri sonuçlarla karşılaştırmışlar ve Tavlama Benzetimi Yöntemi' nin diğer yöntemlere göre daha fazla hesaplama zamanı içermesine karşın, ilgilenilen fonksiyonların tümel minimumlarını bulmada daha etkin bir yöntem olduğunu gözlemlemişlerdir. Ingber (1995), n-boyutlu uzayda doğrusal olmayan bir fonksiyona en iyi yaklaşımı sağlamak amacıyla geliştirdiği Tavlama Benzetimi Algoritması' nı iyileştirerek, yöntemi Uyarlanmış Tavlama Benzetimi Algoritması adı altında yeniden düzenlemiş ve diğer Tavlama Benzetimi Algoritmaları' ndan daha etkin olduğunu ileri sürmüştür.

Bu çalışmada Tavlama Benzetimi Yöntemi tanımlanarak, çok sayıda yerel minimuma sahip, doğrusal olmayan test fonksiyonlarının tümel minimum değerlerinin elde edilmesine yer verilecektir. Uyarlanmış Tavlama Benzetimi Yöntemi ile doğrusal olmayan fonksiyonların tümel minimumunun elde edilmesini amaçlayan çalışmanın 2. Bölümü' nde, çok değişkenli doğrusal olmayan kısıtsız optimizasyon yöntemleri ile metasezgisel yöntemler üzerinde durulacak ve bu yöntemler hakkında bilgi verilecektir. Bir metasezgisel yöntem olan Tavlama Benzetimi Yöntemi ile farklı ölçütler kullanılarak elde edilen Uyarlanmış Tavlama Benzetimi Yöntemi' ne ilişkin oluşturulan algoritmik adımlar tanımlanacaktır. Bölüm 3' te, tanımlanan algoritmanın geçerliliği literatürde kullanılan doğrusal olmayan test fonksiyonları üzerinde incelenecektir. 4. Bölümde ise, doğrusal olmayan fonksiyonların en iyi değerlerinin elde edilmesi amacıyla yapılan simülasyon çalışması sonuçları ve sonraki çalışmalar için öneriler sunulacaktır.

## 2. Çok Değişkenli Doğrusal Olmayan Kısıtsız Optimizasyon Yöntemleri ve Metasezgisel Yöntemler

Fen ve mühendislik alanındaki bir çok optimizasyon problemi tümel en iyi çözümlere gereksinim duyar. Tümel optimizasyon problemleri, gerçek dünya problemlerinin matematiksel modellemesinde tanımlanır.

İlgilenilen fonksiyonun birden fazla yerel minimuma sahip olduğu durumlarda amaç fonksiyonunun bir tek minimum değerinin bulunması istenir. Bu değer tümel minimum olarak adlandırılır ve optimizasyon hesaplamaları için ulaşılmak istenen ideal noktadır [7]. Minimumu bulunmak istenen amaç fonksiyonu üzerinde herhangi bir kısıtlama yok ise, ilgilenilen problem kısıtsız optimizasyon problemidir. Kısıtlanmamış problem, minimum değeri bulunmak istenen  $f$

amaç fonksiyonu  $f : R^n \rightarrow R$  ile tanımlı olmak üzere

$$\min_{x \in R^n} f(x) \quad (1)$$

problemi biçimindedir [1].

Çok değişkenli doğrusal olmayan kısıtsız optimizasyon problemlerinin çözümünde bir çok yöntem kullanılabilir. (1) ifadesinde tanımlanan kısıtsız minimizasyon probleminin çözümü için kullanılan bu yöntemleri türeve dayalı yöntemler ve doğrudan yöntemler başlıkları altında iki grupta toplamak mümkündür. Türeve dayalı yöntemler, 1847 yılı başlarında Cauchy tarafından optimizasyonda kullanılmıştır. Gradyant yöntemler olarak da bilinen bu yöntemler, fonksiyon değerleri ile birlikte fonksiyonun birinci ve daha yüksek dereceli türevlerini de göz önüne alarak en iyi noktayı araştırırlar. Doğrudan yöntemler, ilgilenilen çok değişkenli fonksiyonun kısmi türevleri kullanılmaksızın yalnızca amaç fonksiyon değerlerini göz önüne alarak en iyi noktayı araştıran yöntemlerdir [3].

Metasezgisel yöntemler, karmaşık optimizasyon problemlerinin çözümü için önerilen alternatif yöntemlerdir. 1980' lerin sonlarına doğru bilgisayarlarda hesaplama gücü arttıkça metasezgisel yöntemlerin gelişimi artmıştır [17]. Genellikle fizikte veya biyolojide gözlenmiş süreçlere ve doğadaki farklı disiplinler arası ilişkilere dayandırılarak oluşturulan bu yöntemler, zor kombinatorik optimizasyon problemlerinin çözümünde oldukça başarılı sonuçlar vermişlerdir [14]. Problemler için kesin çözüm yöntemi olmadığında ya da çözüm zamanı bakımından sıkıntı söz konusu olduğunda, özellikle gerçek zamanlı uygulamalarda metasezgisel yöntemler tercih edilmektedir [2].

Metasezgisel yöntemler büyük ölçekli problemlere uygulandığında, yoğun hesaplama zamanına gereksinim duyacağından yakınsama yavaş olacaktır. Yakınsamanın yavaş olma sebebi, bu yöntemlerin arama yönüne ilişkin çok fazla bilgi kullanmadan rasgele hareketlerle tanım uzayında arama yapmalarıdır. Yerel arama yöntemleri arama yönü için mantıksal hareketler oluşturarak yerel bilgiyi kullandıkları için daha hızlı yakınsamaya sahiptirler. Fakat kolaylıkla yerel minimum tuzağına düşebilirler. Tavlama Benzetimi Yöntemi, bu tuzaklardan kurtulmak amacıyla geliştirilmiş bir metasezgisel yöntemdir. Bazı sıvı haldeki maddelerin soğutulması ve minimum enerjili kristal yapıda donması ile genel bir sistemin minimumunu araştırma arasındaki analojiye (benzeşime) dayanan bir rasgele arama yöntemidir. Şu anki durumundan bir sonraki durumuna geçmek isteyen bir algoritma olarak düşünülebilir. Bu süreçte temel kontrol parametresi sıcaklıktır. Tablo 1' de, tümel minimumu bulunmak istenen optimizasyon problemi ve fiziksel sistem arasındaki ilişki tanımlanmıştır.

**Tablo 1.** Fiziksel sistem ve optimizasyon problemleri arasındaki ilişki [17]

Fiziksel Sistem	Optimizasyon Problemleri
Fiziksel sistemin durumu	Mümkün çözümler
Enerji ve minimum enerji	Amaç fonksiyonu ve tümel minimum
Durum değişimi	Komşuluk çözümleri
Sıcaklık	Kontrol parametresi
Donma durumu	Sezgisel çözüm

Tavlama Benzetimi düşüncesi, ilk olarak Metropolis, Rosenbluth ve Teller' in 1953 yılında hazırladıkları bir algoritma ile ortaya atılmıştır [13]. Hazırlanan algoritma, bilgisayar simülasyonu kullanarak, sabit sıcaklıkta kapalı bir düzende parçacıkların denge dağılımlarını hesaplamaya dayanır. Bu düşünceden yararlanarak 1983' te Kirkpatrick, Gelatt ve Vecchi ilk olarak, bir kombinatorik optimizasyon problemi olan Gezgın Satıcı Problemi' ne, sıcaklığın azalışına ilişkin bir soğuma fonksiyonu düzenleyerek, Metropolis Kriteri' ni uygulamışlardır [12]. Genel Tavlama Benzetimi sürecinin algoritmik adımları şöyledir:

**Adım 1:**  $T_0$ , başlangıç sıcaklığı ve  $x_0$ , bir rasgele başlangıç noktası seçilir.  $T = T_0$  ve  $x = x_0$  alınır.

**Adım 2:** Başlangıç noktasının fonksiyona verdiği değer  $E = f(x_0)$  hesaplanır.

**Adım 3:** Her bir  $k$  iterasyonu için, belirlenen  $T$  sıcaklık değerinde aşağıdaki işlemler yapılır:

i) Mevcut noktanın komşuluğunda yeni  $x'$  noktası seçilir.

ii)  $x'$  noktasının fonksiyon değeri  $E' = f(x')$  hesaplanır.

iii) Tanımlanan  $h(x) = -\frac{(E' - E)}{kT}$  kabul fonksiyonu ile  $x = x'$  ve  $E = E'$  alınır.

iv) Tanımlanan soğuma fonksiyonuna göre,  $T(k+1) = cT(k)$ ,  $0 < c < 1$  biçiminde  $T$  sıcaklığı azaltılır.

**Adım 4:**  $x'$  ve  $E'$ , sırasıyla en iyi nokta ve en iyi fonksiyon değeri olarak alınır [16].

## 2.1 Uyarlanmış Tavlama Benzetimi Yöntemi

Tavlama Benzetimi Algoritması, en iyi değeri bulunmak istenen noktaların komşuluk yapılarına ve kontrol parametresi olan sıcaklığa bağlı olarak farklılaştırılabilir. Ingber (1995), kontrol parametresi olan sıcaklığa ilişkin oluşturduğu farklı soğuma fonksiyonları ile en iyi sonucu elde etmeye yönelik çalışmalar yapmıştır. Tümel minimum için stokastik olarak bir çözüm uzayı arayan Uyarlanmış Tavlama Benzetimi Algoritması, bu algoritmalar içinde üstel olarak daha hızlı sonuç veren bir algoritmadır. Algoritma, optimal fonksiyon değerlerini sınırlı bir arama uzayında araştırarak biçimde düzenlenmiştir.  $x_k^i$ ,  $i$  boyutlu uzayda  $k$ . adımda tanımlanan bir parametre olmak üzere  $x_k^i \in [A_i, B_i]$  aralığında tanımlıdır. Burada  $A_i$ , her bir parametre için alt sınırı ve  $B_i$ , her bir parametre için üst sınırı gösterir.  $y_i \in [-1, 1]$  olmak üzere  $y_i$ , parametrelerin geçerli bir kümesi elde edilene kadar yeni parametre değerlerini elde etmek için kullanılan rasgele sayı olarak tanımlansın. Algoritma için  $g$  yaratıcı fonksiyonu,

$$g_T(y) = \prod_{i=1}^s \frac{1}{2(|y_i| + T_i) \ln(1 + 1/T_i)} \equiv \prod_{i=1}^s g_T^i(y_i) \quad (2)$$

dir. Bu dağılıma göre yeni nokta üretmek için,  $U_i \in U[0, 1]$  olan  $U_i$  rasgele değişkeni kullanılarak  $y_i$  sayıları,

$$y_i = \text{sgn}\left(U_i - \frac{1}{2}\right) T_i \left[ \left(1 + \frac{1}{T_i}\right)^{|2U_i - 1|} - 1 \right] \quad (3)$$

eşitliği ile tanımlanır. Soğuma fonksiyonu  $T_i(k) = T_{0i} e^{-c_i k^{1/s}}$  olarak seçildiğinde

$$\sum_{k=k_0}^{\infty} g_k \approx \sum_{k=k_0}^{\infty} \left[ \prod_{i=1}^s \frac{1}{2|y_i|c_i} \right] \frac{1}{k} = \infty \quad (4)$$

ifadesi elde edilir. Bu durumda tümel minimuma ulaşılabilir. Burada  $c_i$ , kullanıcı tarafından tanımlanan parametre olup,  $c_i = m_i e^{-n_i/s}$ ,  $i = 1, 2, \dots, s$  dir.  $m_i$  ve  $n_i$ , ilgilenilen her bir problem için tanımlanmış Uyarlanmış Tavlama Benzetimi Yöntemi' ne özgü parametrelerdir. Algoritmanın tümel minimuma ulaşmasında bu parametrelerin seçimi önemlidir.  $c_i$  parametresi,

$$T_{son} = T_0 e^{-m_i} \quad (5)$$

ifadesi sağlanacak biçimde seçilebilir [10]. Uyarlanmış Tavlama Benzetimi Algoritması' nın adımları aşağıdaki gibi tanımlanabilir:

**Adım 0:**  $f$ , en iyi değeri bulunmak istenen fonksiyon olmak üzere;  $T_0$ , başlangıç sıcaklığı;  $x_0$ , tanım aralığı üzerinde keyfi bir başlangıç noktası;  $T_{\min}$ , son sıcaklık;  $N_T$ , her bir sıcaklıkta yapılan yineleme sayısı;  $M_s$ , sıcaklık değişim sayısı belirlenir.  $A$ , parametre için alt sınır vektörü ve  $B$ , parametre için üst sınır vektörü seçilir.

**Adım 1:** Başlangıç noktasının fonksiyona verdiği değer  $E = f(x_0)$ , hesaplanır.

**Adım 2:** Her bir  $k$  yinelemesi için ( $k = 1, 2, \dots, M_s$ ),  $T$  sıcaklık değerinde

$$x' = x + y(B - A) \quad (6)$$

olacak biçimde yeni bir  $x'$  noktası seçilir. Burada,  $x' \in [A, B]$ ' dir.

**Adım 3:** Adım 2' de bulunan  $x'$  noktasının fonksiyona verdiği değer  $E' = f(x')$ , hesaplanır.

**Adım 4:**  $E' < E$  ise  $x = x'$  ve  $E = E'$  alınır. Aksi halde,  $h(\Delta E) = e^{-\Delta E/T(k)}$  biçiminde tanımlı bir kabul fonksiyonu ve  $(0, 1)$  aralığında Düzgün Dağılıma sahip bir  $Z$  rasgele değişkeni üretilir. Burada,  $\Delta E$  amaç fonksiyonun önceki ve mevcut değeri arasındaki farktır ( $\Delta E = E' - E = f(x') - f(x)$ ). Eğer;  $h(\Delta E) > Z$  ise,  $x = x'$  ve  $E = E'$  alınır,  $N_T = N_T + 1$  olur. Aksi halde;  $h(\Delta E) \leq Z$  ise, Adım 2' ye gidilerek, tanım aralığından yeni nokta üretilir. Bu işlemler, her bir sıcaklık değerinde belirlenen yineleme sayısı kadar yapılır.

**Adım 5:**  $T_i(k) = T_{0i}e^{-c_i k^{1/s}}$ ,  $i = 1, 2, \dots, s$ ,  $k = 1, 2, \dots, M_s$  ile tanımlı soğuma fonksiyonuna göre,  $T$  sıcaklığı azaltılır, yeni sıcaklık fonksiyonu elde edilir.  $M_s = M_s + 1$  alınır.

**Adım 6:**  $x$  ve  $E$ , sırasıyla en iyi nokta ve en iyi fonksiyon değeri olarak alınır. Bu işlemler Adım 0' da tanımlanan  $T_{\min}$  değeri elde edilene kadar devam eder.  $T_{\min}$  değerine ulaşıldığında arama süreci sona erer.

En iyi değeri elde edilecek fonksiyon için uygun soğuma fonksiyonu belirlenemediğinde, en iyi çözüme ulaşmak zor olacaktır. Ingber (1995) tarafından hazırlanan algoritma ile fonksiyonların hesaplama zamanlarının azaltılması başarılı bir biçimde sağlanmıştır.

### 3. Doğrusal Olmayan Kısıtsız Fonksiyonların Tümel Minimumlarının Bulunması

Çalışmanın bu bölümünde, literatürde karşılaşılan ve optimizasyon algoritmalarının etkinliklerinin test edilmesinde kullanılan doğrusal olmayan, birden fazla yerel minimumlu test fonksiyonlarının Uyarlanmış Tavlama Benzetimi Algoritması ile tümel minimum değerleri elde edilmeye çalışılmıştır. Test fonksiyonları, Chelouah ve Siarry (2003) ile Hedar (2004)' dan alınmıştır [5,9]. Tümel minimumları elde edilmek istenen 2, 3, 4 boyutlu doğrusal olmayan kısıtsız test fonksiyonlarının tanımları ve arama uzayları şöyledir:

#### 1. Beale Fonksiyonu (BL)

Tümel minimum civarında dar bir parabolik yapıya sahip test fonksiyonudur ve

$$BL(x) = (1.5 - x_1 + x_1 x_2)^2 + (2.25 - x_1 + x_1 x_2^2)^2 + (2.625 - x_1 + x_1 x_2^3)^2 \quad (7)$$

$$-4.5 \leq x_i \leq 4.5, \quad i = 1, 2$$

biçiminde tanımlıdır.

## 2. Bohachevsky Fonksiyonu (BH)

Tümel minimumu bulunmak istenen ikinci test fonksiyonu, Bohachevsky, Johnson ve Stein' in 1986 yılındaki çalışmalarından alınmıştır [4]. İlgilenilen fonksiyon simetrik olup, birden fazla yerel minimum ve büküm noktalarına sahiptir. Fonksiyon

$$BH(\underline{x}) = x_1^2 + 2x_2^2 - 0.3 \cos(2\pi x_1) - 0.4 \cos(4\pi x_2) + 0.7$$
$$-10 \leq x_i \leq 10, \quad i = 1, 2 \quad (8)$$

biçimindedir.

## 3. Booth Fonksiyonu (BO)

Optimizasyon algoritmalarının etkinliğini test etmek için kullanılan bir çok yerel minimuma sahip bir test fonksiyonudur ve

$$BO(\underline{x}) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2, \quad -10 \leq x_i \leq 10, \quad i = 1, 2 \quad (9)$$

eşitliği ile tanımlıdır.

## 4. Colville Fonksiyonu (CV)

Optimizasyon algoritmalarının testinde oldukça sık kullanılan 4 boyutlu kısıtsız test fonksiyonlarından biridir ve

$$CV(\underline{x}) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90(x_3^2 - x_4)^2 + 10.1((x_2 - 1)^2 + (x_4 - 1)^2) + 19.8(x_2 - 1)(x_4 - 1)$$
$$-10 \leq x_i \leq 10, \quad i = 1, 2, 3, 4 \quad (10)$$

biçimindedir.

## 5. De Jong Fonksiyonu (DJ)

3 boyutlu bir test fonksiyonudur ve

$$DJ(\underline{x}) = x_1^2 + x_2^2 + x_3^2, \quad -2.56 \leq x_i \leq 2.56, \quad i = 1, 2, 3 \quad (11)$$

biçimindedir.

## 6. Easom Fonksiyonu (ES)

Optimizasyon algoritmalarının etkinliğini test etmek için kullanılan bir çok yerel minimuma sahip bir test fonksiyonu olup

$$ES(\underline{x}) = -\cos(x_1) \cos(x_2) \exp\left(-(|x_1 - \pi|^2 - (x_2 - \pi)^2)\right), \quad -100 \leq x_i \leq 100, \quad i = 1, 2 \quad (12)$$

eşitliği ile tanımlıdır.

## 7. Goldstein-Price Fonksiyonu (GP)

Goldstein-Price test fonksiyonu, dört yerel minimuma sahip bir tümel optimizasyon test fonksiyonudur ve

$$GP(\underline{x}) = \left(1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 13x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right)^*$$
$$\left(30 + (2x_1 - 3x_2)^2 (18 - 32x_1 + 12x_1^2 - 48x_2 - 36x_1x_2 + 27x_2^2)\right)$$
$$-2 \leq x_i \leq 2, \quad i = 1, 2 \quad (13)$$

ile tanımlıdır.

### 8. Griewank Fonksiyonu (GW)

Birden fazla yerel minimuma sahip bir test fonksiyonudur. Yerel minimumlar fonksiyon üzerinde düzenli olarak dağılmışlardır. Fonksiyon

$$GW(\underline{x}) = 1 + \sum_{i=1}^2 \left( \frac{x_i^2}{4000} \right) - \prod_{i=1}^2 \left[ \cos \left( \frac{x_i}{\sqrt{i}} \right) \right] , \quad -600 \leq x_i \leq 600 , \quad i = 1, 2 \quad (14)$$

ile tanımlıdır.

### 9. Matyas Fonksiyonu (MT)

Optimizasyon algoritmalarının etkinliğini test etmek için kullanılan test fonksiyonlarından biri olup, fonksiyon

$$MT(\underline{x}) = 0.26(x_1^2 + x_2^2) - 0.48x_1x_2 , \quad -5 \leq x_i \leq 10 , \quad i = 1, 2 \quad (15)$$

biçimindedir.

### 10. Rastrigin Fonksiyonu

Griewank fonksiyonuna benzer olarak çok sayıda yerel minimum içeren test fonksiyonudur. Çalışmada 2 ve 3 boyutlu Rastrigin fonksiyonları ile ilgilenildi.

#### i. Rastrigin Fonksiyonu-1 (RR1)

$$RR1(\underline{x}) = 20 + \sum_{i=1}^2 (x_i^2 - 10 \cos(2\pi x_i)) , \quad -5.12 \leq x_i \leq 5.12 , \quad i = 1, 2 \quad (16)$$

#### ii. Rastrigin Fonksiyonu-2 (RR2)

$$RR2(\underline{x}) = 30 + \sum_{i=1}^3 (x_i^2 - 10 \cos(2\pi x_i)) , \quad -5.12 \leq x_i \leq 5.12 , \quad i = 1, 2, 3 \quad (17)$$

### 11 . Rosenbrock Fonksiyonu

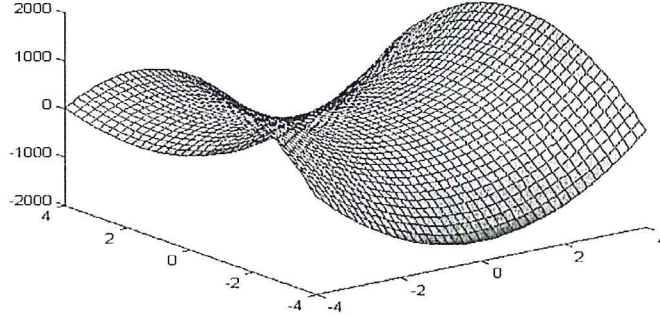
Rosenbrock test fonksiyonu, Banana fonksiyonu olarak da bilinen klasik bir optimizasyon problemidir. Fonksiyonun tümel minimumu, uzun, dar ve parabolik bir yapının içindedir. Tümel minimuma yakınsamak çok zor olduğundan, fonksiyon optimizasyon algoritmalarının etkinliğinin test edilmesinde sıkça kullanılmaktadır. Çalışmada 2 ve 4 boyutlu Rosenbrock fonksiyonları ile ilgilenildi.

#### i. Rosenbrock Fonksiyonu-1 (RB1)

$$RB1(\underline{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 , \quad -2 \leq x_i \leq 2 , \quad i = 1, 2 \quad (18)$$

#### ii. Rosenbrock Fonksiyonu-2 (RB2)

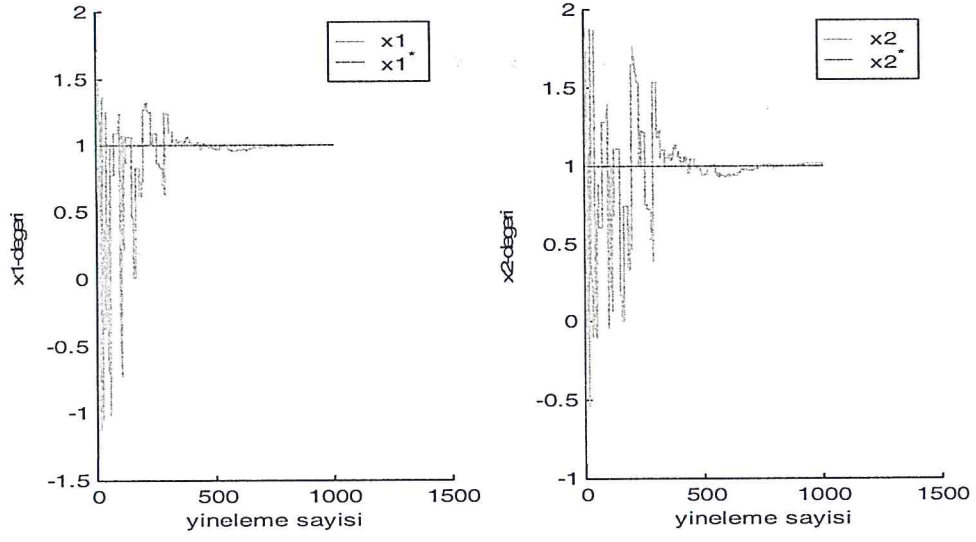
$$RB2(\underline{x}) = \sum_{i=1}^3 \left[ 100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2 \right] , \quad -2 \leq x_i \leq 2 , \quad i = 1, 2, 3, 4 \quad (19)$$



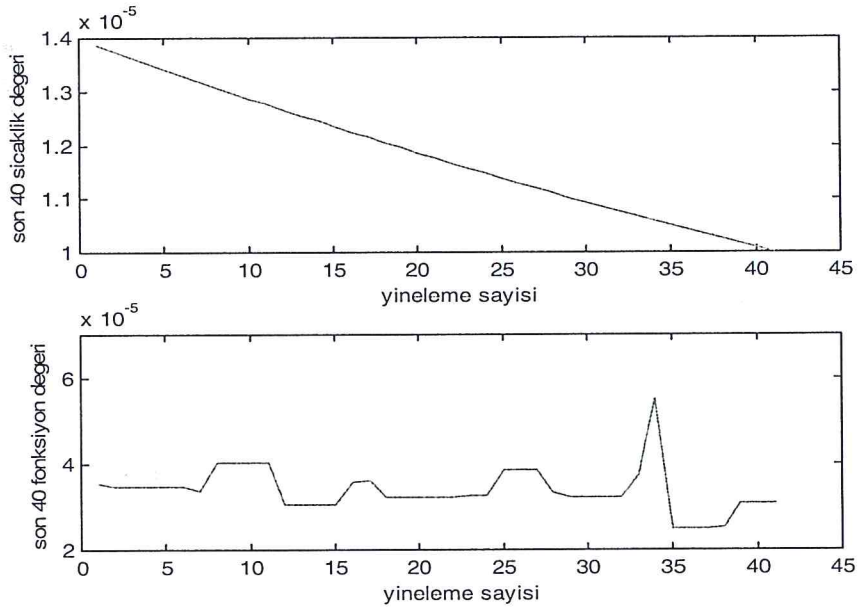
Şekil 1. 2 boyutlu Rosenbrock fonksiyon grafiği

Uyarlanmış Tavlama Benzetimi Yöntemi' nin Bölüm 2.1' de tanımlanan algoritmik adımları işletilmek üzere, Matlab kodlamasından yararlanılarak bir program oluşturulmuştur. Uyarlanmış Tavlama Benzetimi Algoritması' nda, yerel optimum tuzaklarına düşülmesini engellemek için göreceli olarak yüksek bir sıcaklık değeri ( $T$ ) değeri ile aramaya başlanır. Algoritma süresince sıcaklık yavaş yavaş azaltılırken, önceden tanımlanmış yineleme sayısına ( $N_T$ ) bağlı olarak, her sıcaklık değerinde belli sayıda hareket deneyerek arama işlemi sürdürülür. Tümel minimumu bulunmak istenen her bir test fonksiyonu için,  $s$  problemin boyutu olmak üzere, her bir sıcaklıkta yapılan yineleme sayısı ( $N_T$ ),  $s \times 10$  alınmıştır. Önceden belirlenen son sıcaklık değeri ( $T_{\min}$ ), durdurma kriteri olarak düşünülmüştür. Hesaplamalar süresince kaç farklı sıcaklık değerinde arama yapıldığını gösteren yineleme değeri ( $M_s$ ), 100, 1000 ve 5000 alınarak, farklı başlangıç sıcaklıkları ( $T_0$ ) için fonksiyon değerleri ve nokta değerleri elde edilmiştir.  $A_i$  ve  $B_i$ , tümel değeri elde edilmek istenen değişkenler için tanımlanan alt ve üst sınırlar olmak üzere, her bir test fonksiyonunun arama uzayından keyfi başlangıç noktaları ( $x_0$ ) seçilerek, fonksiyonların tümel minimum değerlerine ulaşılmaya çalışılmıştır. İlk olarak simülasyon çalışmaları Şekil 1' de grafiği görülen, optimizasyon algoritmalarının etkinliğinin test edilmesinde sıkça kullanılan 2 boyutlu Rosenbrock test fonksiyonu üzerinde yapılmıştır.  $T_{\min} = 10^{-5}$  ve  $T_0 = 100$  alınarak, değişkenlerin tanım aralığından keyfi olarak seçilen  $x_0 = [1.7 \ -0.9]'$  başlangıç noktası için, 1000 yineleme sonucunda değişkenlere ilişkin elde edilen tümel değerlerin grafikleri Şekil 2' dedir. Şekil 2 incelendiğinde,  $x_{tümel}^* = [1 \ 1]'$ ,  $RB1_{tümel}^* = 0$  gerçek tümel değerlerine, oldukça yakın  $x^* = [1.0046 \ 1.0096]'$  sonucuna ulaşıldığı gözlenmiştir. Hesaplama süresi, CPU= 1.7970 sn ve algoritma sonucu elde edilen fonksiyon değeri  $RB1^* = 3.0804c-005'$  dir. Şekil 3' te son 40 yinelemedeki sıcaklık ve fonksiyon değerleri yer almaktadır.





Şekil 2.  $x_0 = [1.7 \ -0.9]'$ ,  $T_0 = 100$ ,  $M_S = 1000$  için elde edilen tümel nokta değerleri



Şekil 3. Son 40 yinelemedeki sıcaklık ve fonksiyon değerleri

**Tablo 2.**  $x_0 = [1.7 \ -0.9]'$  başlangıç noktası için elde edilen tümel değerler ve hesaplama süreleri

$T_0$	Yineleme sayısı ( $M_s$ )		
	100	1000	5000
1	$RBI^* = 3.3735e-004$ $\bar{x}^* = [1.0180 \ 1.0359]'$ CPU: 0.2820 sn	$RBI^* = 7.8518e-004$ $\bar{x}^* = [1.0087 \ 1.0174]'$ CPU: 2.0160 sn	$RBI^* = 1.6516e-005$ $\bar{x}^* = [0.9960 \ 0.9921]'$ CPU: 9.8590 sn
10	$RBI^* = 1.1803e-005$ $\bar{x}^* = [1.0032 \ 1.0064]'$ CPU: 0.2820 sn	$RBI^* = 1.0484e-005$ $\bar{x}^* = [1.0030 \ 1.0058]'$ CPU: 2 sn	$RBI^* = 4.8444e-007$ $\bar{x}^* = [1.0006 \ 1.0012]'$ CPU: 9.7660 sn
100	$RBI^* = 1.0307e-004$ $\bar{x}^* = [1.0083 \ 1.0172]'$ CPU: 0.2970 sn	$RBI^* = 1.1738e-006$ $\bar{x}^* = [1.0006 \ 1.0011]'$ CPU: 1.9690 sn	$RBI^* = 3.5020e-007$ $\bar{x}^* = [1.0004 \ 1.0009]'$ CPU: 9.6880 sn
1000	$RBI^* = 4.1187e-004$ $\bar{x}^* = [0.9799 \ 0.9604]'$ CPU: 0.2820 sn	$RBI^* = 1.8593e-005$ $\bar{x}^* = [1.0033 \ 1.0068]'$ CPU: 1.9530 sn	$RBI^* = 1.5859e-006$ $\bar{x}^* = [1.0008 \ 1.0015]'$ CPU: 9.6100 sn
10000	$RBI^* = 1.7005e-005$ $\bar{x}^* = [1.0041 \ 1.0083]'$ CPU: 0.2820 sn	$RBI^* = 1.6138e-006$ $\bar{x}^* = [0.9987 \ 0.9975]'$ CPU: 1.9690 sn	$RBI^* = 3.9140e-007$ $\bar{x}^* = [1.0006 \ 1.0012]'$ CPU: 9.6880 sn

Tablo 2' de 2 boyutlu Rosenbrock test fonksiyonuna ilişkin  $x_0 = [1.7 \ -0.9]'$  başlangıç noktası için, farklı yineleme sayılarında elde edilen simülasyon sonuçları özetlenmiştir. Aynı  $T_{min} = 10^{-5}$  için, aynı yineleme sayısında,  $T_0$  değerleri farklı olsa da algoritmanın tümel minimum değerlere ulaştığı gözlenmiştir. Bununla birlikte aynı  $T_0$  değeri için, yineleme sayısı artırıldığında algoritma daha küçük adımlarla hareket edeceğinden,  $T_{min}$  civarında algoritmanın çok sayıda arama yaptığı söylenebilir. Bu durum, hesaplama süresinin artmasına neden olmaktadır. Hesaplama süresinin uzamasına rağmen, Tablo 2' de yer alan simülasyon sonuçlarından, yineleme sayısı 5000 iken ulaşılan tümel değerlerin, yineleme sayısı 100 iken elde edilen değerlere göre daha duyarlı olduğu gözlenmiştir.

Algoritma için en önemli zorluk, başlangıç sıcaklığının ve soğuma fonksiyonunun iyi tanımlanmış olmasıdır. Hesaplanan fonksiyon değerinde önemli bir değişiklik olmayana kadar sıcaklık azaltma süreci devam eder. Algoritmaya özgü parametreler ilgilenilen probleme uygun olarak belirlenemediğinde, tümel değerlerin elde edilmesi oldukça güç olmaktadır. Büyük  $T_{min}$  için, görece olarak küçük  $T_0$  başlangıç değeri ile algoritmaya başlanırsa, uygun tavlama programı elde edilememekte ve her bir sıcaklıkta yeterli arama yapılmadan süreç sona ermektedir. Tanımlanan diğer 2 boyutlu test fonksiyonları için, farklı başlangıç noktalarında elde edilen simülasyon sonuçları Tablo 3' te yer almaktadır. Fonksiyonların ulaşılmak istenen gerçek çözüm değerlerine, tablonun son sütununda yer verilmiştir. Algoritmanın başlangıç koşulları için  $T_{min} = 10^{-5}$ ,  $T_0 = 100$  ve  $M_s = 100$  değerleri alınmıştır.

Tablo 3' ten görüldüğü gibi algoritma, oldukça kısa sürede her bir test fonksiyonu için tanımlanan gerçek çözüm değerlerine ulaşmayı başarmıştır. Tablo 4' te ise 3 ve 4 boyutlu fonksiyonların Uyarlanmış Tavlama Benzetimi Algoritması ile elde edilen sonuçlarına yer

verilmiştir. 3 boyutlu De Joung ve Rastrigin fonksiyonları için  $T_{\min} = 10^{-7}$  alınırken, 4 boyutlu Colville ve Rosenbrock fonksiyonları için  $T_{\min} = 10^{-9}$  alınmıştır. Başlangıç sıcaklığı bu dört fonksiyon için  $T_0 = 100$ ' dır. Tümel değeri bulunmak istenen değişken (parametre) sayısı artarken, diğer bir ifadeyle problemin boyutu büyürken, algoritmanın kritik sıcaklık etrafında çok sayıda arama yaparak yerel minimum tuzaklarına takılmasına engellemek için daha küçük  $T_{\min}$  değeri seçilmiştir. Tablo 4' ten de görüldüğü gibi, yineleme sayısının artmasıyla hesaplama zamanının uzamasına rağmen, tümel değerlere oldukça yakın sonuçlar elde edilmektedir. Algoritma yerel minimum tuzaklarından kurtulup, tümel değerleri elde etmede oldukça iyi performans göstermiştir.

**Tablo 3.** 2 boyutlu test fonksiyonlarının farklı başlangıç değerleri için Uyarlanmış Tavlama Benzetimi Algoritması ile elde edilen tümel çözümler

Fonksiyon	Başlangıç değeri ( $x_0$ )	Uyarlanmış Tavlama Benzetimi Algoritması ile elde edilen değerler		CPU (sn)	Ulaşılmak istenen tümel çözüm değerleri
		Fonksiyon değeri	Nokta değeri ( $x^*$ )		
Beale	[-2.5 0.7]	7.5913e-006	[2.9968 0.4987]	0.6010	$x_{tümel}^* = [3 \ 0.5]^t$ $BL_{tümel}^* = 0$
	[1.8 -3.4]	6.5665e-005	[3.0203 0.5047]	0.6310	
	[-4.4 2]	1.2119e-005	[3.0087 0.5022]	0.6010	
Bohachevsky	[1.9 -2.6]	2.0786e-005	[0.0006116 0.0006778]	0.6810	$x_{tümel}^* = [0 \ 0]^t$ $BH_{tümel}^* = 0$
	[-8.4 3.2]	4.2639e-005	[0.0006 0.0010]	0.6510	
	[4.3 0.9]	7.5388e-006	[0.0005685 0.0002943]	0.6810	
Booth	[-1.5 7]	7.4240e-005	[0.9937 3.0042]	0.6210	$x_{tümel}^* = [1 \ 3]^t$ $BO_{tümel}^* = 0$
	[-3.9 5]	5.3958e-007	[1.0005 2.9998]	0.6010	
	[6.3 8.9]	6.2799e-006	[1.0012 2.9999]	0.5510	
Easom	[-2.5 0.7]	-1	[3.1420 3.1413]	0.5150	$x_{tümel}^* = [\pi \ \pi]^t$ $ES_{tümel}^* = -1$
	[-35 86]	-1	[3.1413 3.1418]	0.5310	
	[-67 0.4]	-1	[3.1414 3.1412]	0.5320	
Goldstein Price	[1.5 1.5]	3.0000	[0.0001 -1.0002]	0.9110	$x_{tümel}^* = [0 \ -1]^t$ $GP_{tümel}^* = 3$
	[-0.8 1.2]	3.0000	[0.0003 -0.9997]	0.8510	
	[1.3 -0.4]	3.0000	[0.0001 -1.0003]	0.8510	
Griewank	[100 100]	4.7881e-006	[-0.0022 -0.0031]	2.2500	$x_{tümel}^* = [0 \ 0]^t$ $GW_{tümel}^* = 0$
	[-45 300]	2.1842e-005	[-0.0013 -0.0092]	2.2660	
	[523 -1.4]	3.8399e-005	[-0.0044 0.0107]	2.2810	
Matyas	[-4.5 6.7]	7.1956e-006	[0.0084 0.0119]	0.6110	$x_{tümel}^* = [0 \ 0]^t$ $MT_{tümel}^* = 0$
	[3.5 -4.6]	4.6669e-006	[-0.0000052 - 0.0000085]	0.5910	
	[2.2 5.9]	2.6920e-006	[0.0082 0.0082]	0.6510	
Rastrigin-1	[-1.2 1.5]	1.2810e-005	[0.0001550 0.0002014]	0.7110	$x_{tümel}^* = [0 \ 0]^t$ $RR1_{tümel}^* = 0$
	[4.7 -3.9]	1.5657e-005	[-0.0002446 0.0001382]	0.6210	
	[2.4 0.1]	1.4139e-005	[0.0001419 0.0002262]	0.6210	
Rosenbrock-1	[-0.5 1.2]	1.5257e-004	[1.0107 1.0221]	1.0010	$x_{tümel}^* = [1 \ 1]^t$ $RB1_{tümel}^* = 0$
	[1.6 -0.8]	2.5527e-004	[0.9841 0.9687]	1.0010	
	[-1.9 0.6]	8.1083e-006	[1.0022 1.0043]	1.0310	

**Tablo 4.**  $T_0 = 100$  için 3 ve 4 boyutlu fonksiyonların Uyarlanmış Tavlama Benzetimi Algoritması ile elde edilen tümel değerleri

Fonksiyon	Yineleme sayısı			Gerçek çözüm değerleri
	100	1000	10000	
<b>De Jong</b> $x_0 = [1 \ -1.9 \ 0.8]^t$	$DJ^* = 3.0674e-007$ $\underline{x}^* = [0.0002011$ $\quad -0.0002613$ $\quad -0.0004450]$ CPU: 1.5310 sn	$DJ^* = 1.2887e-007$ $\underline{x}^* = [0.0003338$ $\quad -0.0000870$ $\quad 0.0000992]$ CPU: 1.5310 sn	$DJ^* = 1.4641e-007$ $\underline{x}^* = [0.0001602$ $\quad 0.0003474$ $\quad -0.0000060]$ CPU: 34.7970 sn	$\underline{x}_{tümel}^* = [0 \ 0 \ 0]^t$ $DJ_{tümel}^* = 0$
<b>Rastrigin-2</b> $x_0 = [-3.9 \ 2.5 \ 1]^t$	$RR2^* = 0.0227$ $\underline{x}^* = [0.0046$ $\quad 0.0086$ $\quad 0.00043]$ CPU: 0.922 sn	$RR2^* = 9.0532e-004$ $\underline{x}^* = [0.0004670$ $\quad -0.0004226$ $\quad -0.0002445]$ CPU: 3.953 sn	$RR2^* = 4.0446e-004$ $\underline{x}^* = [-0.0000436$ $\quad -0.0000641$ $\quad -0.0001199]$ CPU: 36.859 sn	$\underline{x}_{tümel}^* = [0 \ 0 \ 0]^t$ $RR2_{tümel}^* = 0$
<b>Colville</b> $x_0 = [2 \ 0.3 \ -5 \ -6.8]$	$CV^* = 0.0498$ $\underline{x}^* = [0.8756$ $\quad 0.7681$ $\quad 1.1085$ $\quad 1.2319]$ CPU: 1.2020 sn	$CV^* = 0.0583$ $\underline{x}^* = [1.1182$ $\quad 1.2511$ $\quad 0.8634$ $\quad 0.7463]$ CPU: 10.3850 sn	$CV^* = 4.2318e-004$ $\underline{x}^* = [0.9890$ $\quad 0.9782$ $\quad 1.0107$ $\quad 1.0215]$ CPU: 113.9540 sn	$\underline{x}_{tümel}^* = [1 \ 1 \ 1]^t$ $CV_{tümel}^* = 0$
<b>Rosenbrock-2</b> $x_0 = [-1.5 \ 0.7 \ 1.3]$	$RB2^* = 0.0096$ $\underline{x}^* = [1.0209$ $\quad 1.0422$ $\quad 1.0862$ $\quad 1.1798]$ CPU: 1.734 sn	$RB2^* = 0.0079$ $\underline{x}^* = [0.9802$ $\quad 0.9608$ $\quad 0.9231$ $\quad 0.8517]$ CPU: 5.265 sn	$RB2^* = 1.7452e-004$ $\underline{x}^* = [0.9971$ $\quad 0.9942$ $\quad 0.9885$ $\quad 0.9771]$ CPU: 53.937 sn	$\underline{x}_{tümel}^* = [1 \ 1 \ 1]^t$ $RB2_{tümel}^* = 0$

Tümel minimumu bulunmak istenen bir diğer test fonksiyonu Judge Fonksiyonu' dur [11]. Judge Fonksiyonu, parametrik bir fonksiyon olup, birden fazla yerel minimuma sahip ekonometrik bir modeldir [8].

$$y_t = \theta_1 x_{t1} + \theta_2 x_{t2} + \theta_2^2 x_{t3} + e_t, \quad t = 1, 2, \dots, 20 \quad (20)$$

biçiminde tanımlanan modelde,

$$\underline{y} = [y_1, y_2, \dots, y_{20}]^t, \quad \underline{\theta} = [\theta_1 \ \theta_2]^t, \quad \underline{e} = [e_1, e_2, \dots, e_{20}]^t$$

dir.

$$\underline{f}(\underline{\theta}) = \begin{bmatrix} \theta_1 x_{11} + \theta_2 x_{12} + \theta_2^2 x_{13} \\ \theta_1 x_{21} + \theta_2 x_{22} + \theta_2^2 x_{23} \\ \vdots \\ \theta_1 x_{20,1} + \theta_2 x_{20,2} + \theta_2^2 x_{20,3} \end{bmatrix} \quad (21)$$

olup,  $\underline{\theta} = [\theta_1 \ \theta_2]'$  parametre kümesinin hataları en küçük yapacak biçimde tahmin edicisini bulmak için,

$$H(\underline{\theta}) = (\underline{y} - \underline{f}(\underline{\theta}))' (\underline{y} - \underline{f}(\underline{\theta})) \quad (22)$$

ifadesinin minimum yapılması gerekmektedir [11]. Burada amaç, hata kareleri toplamı minimum olacak biçimde model parametrelerini Uyarlanmış Tavlama Benzetimi Yöntemi ile tahmin etmektir.

Goffe ve arkadaşları (1994) çalışmalarında, (22) eşitliği ile verilen  $H(\underline{\theta})$  fonksiyonunun minimumunu bulmak için Simpleks, Eşlenik Gradyant, Quasi-Newton ve Tavlama Benzetimi Yöntemi' ni kullanmışlardır. Bu yöntemlerin çözüm aşamasında, EK-1' de verilen veriler kullanılmış ve 100 yineleme sonucunda iki yerel minimum noktaya ulaşılmıştır. Bu yerel minimum noktalar, 16.083 ve 20.482 olarak bulunmuştur. Judge Fonksiyonu' nun tümel minimumu ise  $H(\underline{\theta}^*) = 16.083$  ' tür.

Minimum noktaların bulunması sürecindeki yineleme sayıları Tablo 5' te verilmiştir. Tablo 5 incelendiğinde, Simpleks, Eşlenik Gradyant, Quasi-Newton yöntemleri ile iki yerel minimum değeri elde edilirken, Tavlama Benzetimi Yöntemi' nin yerel minimum tuzağına takılmadan tümel minimuma ulaştığı söylenebilir.

**Tablo 5.** Judge Fonksiyon değerlerine ilişkin yineleme sayıları

Yöntemler	Toplam yineleme sayısı	100 denemede 16.083 fonksiyon değerini bulan yineleme sayısı	100 denemede 20.482 fonksiyon değerini bulan yineleme sayısı
Simpleks	100	60	40
Eşlenik Gradyant	100	52	48
Quasi-Newton	100	52	48
Tavlama Benzetimi	100	100	—

Judge Fonksiyonu' nun  $0 \leq \theta_i \leq 10$ ,  $i=1,2$  tanım aralığından seçilen farklı başlangıç değerleri için Uyarlanmış Tavlama Benzetimi Algoritması kullanılarak elde edilen tümel değerler Tablo 6' dadır. Algoritma için  $T_{\min} = 10^{-7}$ ,  $T_0 = 100$  ve  $M_s = 100$  alınmıştır.

**Tablo 6.** Farklı başlangıç değerleri için Uyarlanmış Tavlama Benzetimi Algoritması ile elde edilen Judge Fonksiyon çözüm sonuçları

Fonksiyon	Başlangıç değeri ( $\theta_0$ )'	Uyarlanmış Tavlama Benzetimi Algoritması ile elde edilen değerler		CPU (sn)
		Fonksiyon değeri	Nokta değeri ( $\underline{\theta}^*$ )'	
Judge	[3.2 9.4]	16.0817	[0.8646 1.2358]	1.7340
	[5 7]	16.0817	[0.8652 1.2355]	1.7350
	[2.48 6]	16.0817	[0.8644 1.2360]	1.7190

Çizelge incelendiğinde, tanım aralığından keyfi olarak seçilen başlangıç değerleri için Uyarlanmış Tavlama Benzetimi Algoritması ile elde edilen sonuçların, ulaşılmak istenen  $H(\theta^*) = 16.08173$  tümel minimum değeri ile aynı olduğu görülür. Buna göre algoritmanın yerel minimum tuzağına takılmaktan kurtulduğu söylenebilir.

#### 4. Sonuç ve Öneriler

Tavlama Benzetimi Yöntemi,  $s$  boyutlu uzayda, verilen  $s$  değişkenli çok sayıda yerel minimuma sahip optimizasyon problemlerinin tümel minimumlarının elde edilmesinde oldukça etkin bir metasezgisel yöntemdir. Literatürde bulunan test fonksiyonlarının tanım aralıklarından seçilen farklı başlangıç değerlerine ilişkin elde edilen sonuçların özetlendiği çizelgelerden de izlenebileceği gibi, Uyarlanmış Tavlama Benzetimi Yöntemi ile 2, 3, 4 boyutlu, birden fazla yerel minimuma sahip doğrusal olmayan fonksiyonların tümel minimum değerlerine oldukça yakın sonuçlar elde edilmiştir. Uyarlanmış Tavlama Benzetimi Yöntemi'nde sıcaklık, her bir sıcaklıkta yapılan yineleme sayısı, sıcaklık fonksiyonu değişim sayısı gibi uygun başlangıç değerlerinin ve algoritmaya özgü tanımlanan parametrelerin seçimi, yöntemin tümel sonuçlara ulaşabilmesinde büyük önem taşır. Elde edilen sonuçlara göre, algoritma parametreleri uygun olarak belirlendiğinde, yöntemin doğrusal olmayan fonksiyonların tümel minimum değerlerinin elde edilmesinde oldukça iyi sonuçlar veren bir akıllı arama tekniği olduğu söylenebilir.

Zor problemlere kolaylıkla uyarlanabilen yöntemin en büyük dezavantajı, yoğun hesaplama zamanına ihtiyaç duymasıdır. Değişken sayısı arttıkça bu hesaplama süresi uzamakta, algoritmayı iyileştirecek farklı kısıtlamalara ihtiyaç duyulmaktadır. Şüphesiz, hesaplama gücü yüksek bilgisayarlar kullanıldığında hesaplama zamanı da azalacaktır.

Türeve dayalı yöntemler ya da doğrudan yöntemlerden biri ile Uyarlanmış Tavlama Benzetimi Yöntemi birleştirilip, melez yaklaşım sağlayarak tümel minimuma daha kolay ulaşmanın yanı sıra, bundan sonra yapılacak çalışmalarda, yöntem için büyük önem taşıyan soğuma fonksiyonlarının farklılaştırılması ile algoritmanın iyileştirilmesi ve yöntemin daha kısa sürede tümel en iyi çözümlere ulaşması sağlanabilir.

EK-1: (22) eşitliğinin çözümünde kullanılan veriler

t	$y_t$	$x_{t1}$	$x_{t2}$	$x_{t3}$
1	4.284	1	0.286	0.945
2	4.149	1	0.973	0.585
3	3.877	1	0.384	0.310
4	0.533	1	0.276	0.058
5	2.211	1	0.973	0.455
6	2.389	1	0.543	0.779
7	2.145	1	0.957	0.259
8	3.231	1	0.948	0.202
9	1.998	1	0.543	0.028
10	1.379	1	0.797	0.099
11	2.106	1	0.936	0.142
12	1.428	1	0.889	0.296
13	1.011	1	0.006	0.175
14	2.179	1	0.828	0.180
15	2.858	1	0.399	0.842
16	1.388	1	0.617	0.039
17	1.651	1	0.939	0.103
18	1.593	1	0.784	0.620
19	1.046	1	0.072	0.158
20	2.152	1	0.889	0.704

**Kaynaklar**

1. Apaydın, A., **Optimizasyon**, A.Ü.F.F. Döner Sermaye İşletmesi Yayını, No:41,Ankara, (1996).
2. Bakır, M.A., Altunkaynak, B., **Tamsayılı Programlama**, Nobel Yayın Dağıtım, Ankara (2003).
3. Bhatti M.A., **Practical Optimization Methods With Mathematica Applications**, Springer-Verlag, New-York (2000).
4. Bohachevsky, I.Q., Johnson, M.E., Stein, M.L., **Generalized Simulated Annealing for Function Optimization**, Technometrics, 28(3), 209-217, (1986).
5. Chelouah, R., Siarry, P., **Genetic and Nelder-Mead algorithms hybridized for a more accurate global optimization of continuous multimimima functions**, European Journal of Operational Research., 148, 335-348, (2003).
6. Corana, A., Marchesi, M., Martini, C., Ridella, S., **Minimizing Multimodel Functions of Continuous Variables with the "Simulated Annealing" Algorithm**, ACM Transactions on Mathematical Software, 13(3), 262-280, (1987).
7. Frandsen, P.E., Jonasson, K., Nielsen, H.B., Tingleff, O., **Unconstrained Optimization GAMS**, <http://gams.nistrav>, Erişim Tarihi: 03.02.2005, (2004).
8. Goffe, W.L., Ferrier, G.D., Rogers, J., **Tümel optimization of statistical functions with simulated annealing**, Journal of Econometrics, 60, 65-99, (1994).
9. Hedar, A.R., **Studies on Metaheuristics for Continuous Global Optimization Problems**, Kyoto, Japan, (2004).
10. Ingber, L., **Adaptive Simulated Annealing (ASA): Lessons Learned**. Control and Cybernetics (to be published), (1995).
11. Judge, G.G., Griffiths, W.E., Hill, R.C., Lütkepohl, H., Lee, T.C. **The theory and practice of econometrics**, 2nd ed., Wiley, New-York, (1985).
12. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., **Optimization by Simulated Annealing**, Science. 220, 4598, (1983).
13. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., **Equation of State Calculations by Fast Computing Machines**, The Journal of Chemical Physics, 21(6); 1087-1092, (1953).
14. Pardalos, P.M., Romeijn, H.E., Tuy, H., **Recent developments and trends in tümel optimization**, Journal of Computational and Applied Mathematics, 124, 209-228, (2000).
15. Penny, J., Lindfield, G., **Numerical Methods Using Matlab**, Prentice Hall, London, (1995).
16. Rosen, B., **Function Optimization based on Advanced Simulated Annealing**, IEEE, 289-293, (1993).
17. Winker, P., Gilli, M., **Application of optimization heuristics to estimation and modelling problems**, Computational Statistics and Data Analysis, (2003).

