

	<b>SAKARYA ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ DERGİSİ</b> <i>SAKARYA UNIVERSITY JOURNAL OF SCIENCE</i>		
	<b>e-ISSN: 2147-835X</b> <b>Dergi sayfası: <a href="http://www.saujs.sakarya.edu.tr">http://www.saujs.sakarya.edu.tr</a></b>		
	<u>Geliş/Received</u> 20.03.2017 <u>Kabul/Accepted</u> 28.03.2018	<u>Doi</u> 10.16984/saufenbilder.298972	

## Genel indirgenmiş gradyan metodu ile eniyileme çözücüsü geliştirilmesi

Mehmet Samet TOPLU\*<sup>1</sup>, Durmuş Sinan KÖRPE<sup>2</sup>

### ÖZ

Bu çalışmada, literatürde bulunan bir denek taşı eniyileme problemi kullanılarak genel indirgenmiş gradyan metodu ile eniyileme çözücüsü geliştirilmiştir. Eniyileme çözücüsünün geliştirilmesinde MATLAB programlama dili kullanılmıştır. Denek taşı probleminde hedef fonksiyon, tasarım değişkenleri olarak tanımlanan farklı genişlik ve yükseklik değerlerine sahip N adet parçadan oluşan bir destekli kirişin hacmini en aza indirgenmesidir. Kirişin her bir parçasının sağ ucundaki eğilme gerilmesi, fonksiyonel kısıtlama olarak tanımlanırken her parçadaki genişliğin yüksekliğe oranı ise geometrik kısıtlama olarak tanımlanmıştır. Geliştirilen çözücü ve MATLAB içinde tanımlı olan ardışık ikinci derece programlama metodu ile elde edilen sonuçlar karşılaştırılmış ve 25 parça için uygun parametreler elde edilmiştir. Sonuç olarak, genel indirgenmiş gradyan metodu ile geliştirilen çözücünün eniyileme problemlerinin çözümünde kullanılabilecek uygun bir çözücü olduğu testlerle gösterilmiştir.

**Anahtar Kelimeler:** Eniyileme, Genel İndirgenmiş Gradyan Metodu, Olurlu Bölgeye Geçiş

## Development of an optimization solver with generalized reduced gradient method

### ABSTRACT

In this study, an optimization solver was developed by using the generalized reduced gradient method in order to solve a benchmark optimization problem in the literature. The MATLAB programming language was used in the development of the optimization solver. In the benchmark optimization problem, the objective function is the minimization of volume a cantilever beam that consist of N segments that have different height and thickness values which are the design variables. The bending stress value at the left end of each segment of the cantilevered is the functional constraint whereas the ratio of the width to the height at each segment is defined as the geometric constraint. The results that are obtained by the developed solver and sequential quadratic programming method that is in the MATLAB are compared and the proper parameters are obtained for 25 segments. As a result, it has been shown by tests that the developed solver by the generalized reduced gradient method is a descent solver that can be used to solve the optimization problems.

**Keywords:** Optimization, Generalized Reduced Gradient Method, Moving Back to Feasible Region

<sup>1</sup> Türk Hava Kurumu Üniversitesi, Makina ve Uçak Mühendisliği Bölümü, msamettoplu@gmail.com

<sup>2</sup> Türk Hava Kurumu Üniversitesi, Uçak Mühendisliği Bölümü, dskorpe@thk.edu.tr

## 1. GİRİŞ (INTRODUCTION)

Eniyileme, verilen kısıtlamalar altında istenen faktörleri en üst düzeye çıkararak ve istenmeyen faktörleri en aza indirgeyerek en uygun maliyetli veya en yüksek erişilebilir performansa sahip bir alternatif bulmaktır. Eniyilemenin ilk örneklerinin, milattan önce yaşamış olan Yunan filozofları Zenon, Plato, Aristoteles gibi isimler tarafından, çalışmaları esnasında oluşturulduğu bilinmektedir [1]. Modern eniyileme yöntemlerinin başlangıcı değişimler hesabına (calculus of variations) kadar dayanır. Değişimler hesabı ile ilgili genel çerçeveyi 18. yüzyılda ortaya koyan Lagrange'ın "Lagrange çarpanlar yöntemi" (Lagrangian Multiplier rule) olarak bilinen meşhur metodu da günümüzde eniyileme teorisinin ana konularından birini oluşturmaktadır [2]. En basit ve ilk eniyileme yöntemlerinden olan "En Dik İniş, (EDİ)" (steepest descent) metoduna ait uygulamanın Cauchy tarafından ilk kez gösterildiği 19. yüzyıl ortalarından 20. yüzyılın ortalarına kadar bu sahada maalesef çok az ilerleme kaydedilmiştir. Bu dönemden itibaren bilgisayar teknolojisindeki gelişmelere paralel olarak çok hızlı işlemcilerin kullanılmaya başlanmasıyla eniyileme konusundaki çalışmaların ve yeni uygulamaların miktarı da hızla artmıştır [3]. Eniyileme problemlerinin çözümünde kullanılan çok sayıda metot vardır. Bu metotlardan biri olan genel indirgenmiş gradyan metodunun (GİGM) en büyük avantajı gradyan temelli eniyileme metotları arasında en güçlü olan metot olması, en büyük dezavantajı ise eşitsizlik kısıtlamalarının ve eşitlik kısıtlamalarının sağlandığı ve tasarım değişkenlerinin alt limitleri ve üst limitleri arasında bulunduğu bölge olarak tanımlanan olurlu bölgeyi her yineleme sürecinde sağlamak zorunda olmasıdır [4]. Bütün bu metotların en büyük dezavantajı ise tekrarlanmalı işlemlerin uzun ve karışık olması, numerik işlemlerde karışıklığın ortaya çıkması, kontrol zorluğu ve fazla zaman kaybına sebep olmasıdır. Kullanılan bilgisayar programları arasında eniyileme problemlerini daha kolay çözen ve elde edilen sonuçlar bakımından en uygun olan programlardan birisi MATLAB'dır. Bu programın numerik işlemlerdeki üstünlüğü bilinmektedir. Mühendislik hesap ve uygulamalarını kolaylıkla gerçekleştirebilen matematik tabanlı bir programdır [5].

Literatürde GİGM ile ilgili çok sayıda çalışma mevcuttur. Song v.d. tarafından yapılan bir çalışmada, sportif bir aracın arka tarafında kullanılmakta olan bir kontrol kolunun ağırlığının en düşük hale getirilmesi incelenmiştir. Beş adet tasarım değişkeni ile geometrisi tanımlanabilen kontrol kolunun başlangıç tasarımı üzerinde yüklerden dolayı oluşan gerilmeler eşitsizlik kısıtlaması olarak tanımlanmış, bu sayede eniyilenmiş tasarımın aynı yük üzerinde daha yüksek gerilmelere maruz kalması önlenmiştir. Hedef fonksiyon ise kontrol kolunun ağırlığının azaltılması olarak tanımlanmıştır. ANSYS workbench yapısal tasarım programı ile çözülen eniyileme problemi sonucunda kontrol kolunun ağırlığı %16 oranında azaltılırken, aynı problemin GİGM kullanılarak çözülmesi sonucunda ağırlık %18 oranında azaltılmıştır [6]. Faluyi ve Arum çalışmalarında GİGM ve yapay arı kolonisi metodunu (YAKM) kullanarak en düşük dolu gövdeli kiriş ağırlığını elde etmişlerdir. Çalışmada, profil başlıklarının genişliği ve kalınlığı, ağ yüksekliği ve kalınlığı ve takviye parçaların kalınlığı tasarım değişkenleri olarak tanımlanmıştır. Moment direnci ve kesme dayanımı kısıtlamalar olarak tanımlanmıştır. Bu tanımlara göre GİGM ile elde edilen sonuçlarda ağırlık azalması %7.44 iken YAKM ile elde edilen %7.25'dir [7]. Fedghouche ve Tiliouine çalışmalarında nihai tasarım yüklerinde betonarme T-kirişlerinin maliyetinin en düşük hale getirilmesi incelemişlerdir. T-kirişin kesit alanının boyutlarını oluşturan geometrik tasarım değişkenlerine ek olarak T-kiriş içindeki demir oranı da tasarım değişkeni olarak tanımlanmıştır. Eğilme gerilmesi ve kesme gerilmesi değerleri kısıtlama olarak tanımlanmaktadır. Sonuç olarak, eniyilenmiş tasarımların, başlangıç T-kiriş geometrisi ve demir oranına göre farklılık gösterdiği fakat kesme gerilmesinin kısıtlamasındaki değişikliklerin eniyilenmiş tasarımları etkilemediği ifade edilmiştir [8]. Körpe ve Özgen tarafından yapılan çalışmada, bir insansız hava aracının (İHA) düz uçuşta en düşük sürüklemesi için şekil değiştirebilen kanat eniyilemesinin temel sonuçları sunulmuştur. Elde edilen eniyilenmiş şekil değiştirebilen kanadın, çok düşük hızlarda %35 oranında ve çok yüksek hızlarda %15.5 oranında sürüklenme kuvvetinde azalma sağlanmıştır [9].

## 2. GENEL İNDİRGENMİŞ GRADYAN METODU (GENERALIZED REDUCED GRADIENT METHOD)

GİGM, doğrusal olmayan eşitlik kısıtlamalı eniyileme problemlerinin çözümünde kullanılır [10] ve doğrusal olmayan eniyileme problemlerini çözümü için kullanılan metotlar arasında çözücü olarak geliştirilmesi en zor metot olarak belirtilmiştir [4]. GİGM'nin ayrıntılarını anlamak için Eş. 1-Eş. 5'te doğrusal olmayan eniyileme probleminin genel biçimi verilmiştir [11].

$$\text{En düşük } f(X) \quad (1)$$

$$h_j(X) \leq 0 \quad j = 1, 2, \dots, m \quad (2)$$

$$C_k(X) = 0 \quad k = 1, 2, \dots, l \quad (3)$$

$$x_i^a \leq X_i \leq x_i^u \quad i = 1, 2, \dots, n \quad (4)$$

$$X = \{X_1 \quad X_2 \quad X_3 \quad \dots \quad X_n\}^T \quad (5)$$

Bu eşitliklerde tanımlanan,  $f(X)$ ,  $h_j(X)$ ,  $C_k(X)$ ,  $X$ ,  $x_i^a$  ve  $x_i^u$  sırasıyla en düşüğe indirilecek olan hedef fonksiyon, eşitsizlik kısıtlamaları, eşitlik kısıtlamaları, tasarım değişkenleri kümesi, tasarım değişkenlerinin alt ve üst limitleridir. GİGM teorisi kullanılarak çözülecek eniyileme problemlerinde kısıtlamaların hepsinin eşitlik kısıtlaması olması gerekmektedir. Bu sebepten dolayı değeri negatif olan  $m$  adet eşitsizlik kısıtlamasının her biri, eşitlik kısıtlamasına dönüştürülmesi amacıyla artık (slack) tasarım değişkenleri ile toplanır. Artık tasarım değişkenlerinin alt limiti 0 iken üst limiti yoktur. Yapılan düzenleme sonunda Eş. 6'da gösterildiği gibi eşitlik kısıtlaması sayısı  $l$ 'den  $l + m$ 'ye, toplam tasarım değişkeni sayısı da Eş. 7'de belirtildiği şekilde  $n$ 'den  $n + m$ 'ye yükselir.

$$g_j(X) = 0 \quad j = 1, 2, \dots, m + l \quad (6)$$

$$x_i^a \leq x_i \leq x_i^u \quad i = 1, 2, \dots, m + n \quad (7)$$

Eş. 6'yı kullanarak değişkenleri ortadan kaldırmak gerekir. Bu nedenle, tasarım değişkenleri Eş. 8'de gösterildiği gibi 2 gruba ayrılır:

$$X = \begin{Bmatrix} Y \\ Z \end{Bmatrix} = \begin{matrix} \text{Temel değişkenler} \\ \text{Temel olmayan değişkenler} \end{matrix} \quad (8)$$

Eş. 9 ve Eş. 10'da hedef fonksiyonun ve kısıtlamaların ilk türevleri verilmiştir.

$$df(X) = \sum_{i=1}^{n-l} \frac{\partial f}{\partial y_i} dy_i + \sum_{i=1}^{m+l} \frac{\partial f}{\partial z_i} dz_i = \nabla_Y^T f dY + \nabla_Z^T f dZ \quad (9)$$

$$dg_i(X) = \sum_{j=1}^{n-l} \frac{\partial g_i}{\partial y_j} dy_j + \sum_{j=1}^{m+l} \frac{\partial g_i}{\partial z_j} dz_j = [C] dY + [D] dZ \quad (10)$$

GİGM tüm kısıtlamaların sağlanmasına dayanır. Bir tasarım değişkenindeki herhangi bir değişiklik diğer tasarım değişkenleri tarafından telafi edilmelidir. Böylece kısıtlama veya kısıtlamalar sağlanır. Bu, Eş. 10'un sol tarafının sıfır olmasını ve Eş. 11'in elde edilmesini sağlar.

$$dZ = -[D]^{-1}[C]dY \quad (11)$$

Eş. 9'daki ifade Eş. 10'da kullanılırsa Eş. 12 elde edilir.

$$\frac{df(X)}{dY} = G_r = \nabla_Y^T f - [D]^{-1}[C]\nabla_Z^T f \quad (12)$$

Eş. 12'deki  $G_r$ , genel indirgenmiş gradyan vektörüdür. Genel indirgenmiş gradyan, temel değişkenler tarafından  $(n-m)$  boyutlu olurlu bölge üzerine tanımlanan orijinal  $n$  boyutlu gradyanın geometrik izdüşümü olarak ifade edilebilir [12]. Genel indirgenmiş gradyan vektörü, hedef fonksiyonu en aza indirmek için uygun bir arama yönünü (S) bulmak amacıyla kullanılır. Dik iniş metodunda arama yönü Eş. 13'te tanımlanır.

$$S = -G_r \quad (13)$$

$\|S\|$  belirli bir değerden az ise mevcut  $X$  değerleri problemin en uygun tasarım değişkenleri olarak alınır. Arama yönü analizi sırasında  $X$ 'in üst ve alt sınırları dikkate alınmalıdır.  $G_r$  bulunduktan sonra eğer tasarım değişkenleri üst sınırlarına yaklaşıyorsa ve  $G_r$  sıfırdan küçük ise  $G_r$  sıfıra çekilir. Eğer alt sınırına yaklaşıyorsa ve  $G_r$  sıfırdan büyük ise  $G_r$  yine sıfıra çekilir. Bunun için Eş. 13, Eş. 14 olarak değiştirilir.

$$S = \begin{cases} 0 & \text{Eğer } (Y_i - Y_i^u) < \gamma \text{ ve } G_{r_i} < 0 \\ 0 & \text{Eğer } (Y_i - Y_i^a) < \gamma \text{ ve } G_{r_i} > 0 \\ & \text{aksi halde } -G_{r_i} \end{cases} \quad (14)$$

Bu denklemde kullanılan  $\gamma$ , sınır değeridir. Bu çalışmada geliştirilen çözücü, genel indirgenmiş gradyan vektörünün normu, durma büyüklük değeri olan  $\zeta$ 'den küçükse duracaktır. Bu eşitlikteki S temel değişkenler için arama yönü vektörüdür. Temel olmayan değişkenler için T vektörü Eş. 15 kullanılarak tanımlanır.

$$T = -[D]^{-1}[C]S \quad (15)$$

Arama yönü vektörünün ne kadar ilerleyeceğini bulmak amacıyla adım uzunluğunun ( $\lambda$ ) üst limitinin bulunması için temel ve temel olmayan tasarım değişkenlerinin üst ve alt sınırlarına bakılır. Eş. 16 ve Eş. 17'de bulunan  $\lambda_i$  değerleri arasında en düşük olanı üst sınır olarak seçilir. Alt sınır ise sıfırdır.

$$\lambda_i = \begin{cases} \frac{Y_i^u - Y_i}{S_i} & \text{Eğer } S_i > 0 \\ \frac{Y_i^a - Y_i}{S_i} & \text{Eğer } S_i < 0 \end{cases} \quad (16)$$

$$\lambda_i = \begin{cases} \frac{Z_i^u - Z_i}{T_i} & \text{Eğer } T_i > 0 \\ \frac{Z_i^a - Z_i}{T_i} & \text{Eğer } T_i < 0 \end{cases} \quad (17)$$

Bu sınırlar arasından en uygun noktayı bulmak için altın oran arama metodu kullanılır. En uygun adım uzunluğunu bulduktan sonra, Eş. 18'e göre yeni tasarım değişkenleri bulunur.

$$X_{yeni} = \frac{Y + \lambda S}{Z + \lambda T} \quad (18)$$

Eniyileme için kullanılan bu metotta en önemli faktörlerden birisi, temel ve temel olmayan değişkenlerin nasıl tanımlanacağına yöneliktir. Kısıtlamaların temel olmayan değişkenlere göre Jakobien matrisi olan D matrisinin kesinlikle tersi olmalıdır. Aynı zamanda temel olmayan değişkenlerin alt ve üst limitlerinden uzakta olan tasarım değişkenleri arasından seçilmesi, metodunun en uygun değere hızlı bir şekilde ulaşmasını sağlamaktadır. Her iki durumu da göz önünde bulundurarak temel olmayan değişkenlerin seçilmesi gerekmektedir. D matrisini tersi olacak bir matris şekilde seçebilmek için Eş. 19'da tanımlanan ve kısıtlamaların tasarım değişkenlerine göre Jakobien matrisi olan B matrisinde Gauss eliminasyon yöntemi uygulanır.

$$B = \begin{bmatrix} \nabla g_1(X)^T \\ \nabla g_2(X)^T \\ \nabla g_3(X)^T \\ \cdot \\ \cdot \\ \cdot \\ \nabla g_{m+l}(X)^T \end{bmatrix}_{(m+l) \times (n+m)} \quad (19)$$

Yöntemi daha detaylı bir şekilde anlatmak gerekirse, B matrisinin birinci satırındaki elemanlar arasında mutlak değeri en büyük olan eleman seçilir. Bu eleman pivot elemandır. Seçilen bu eleman bulunduğu sütun numarası (r), r'inci tasarım değişkeninin temel olmayan tasarım değişkeni olduğu anlamına gelir. Aynı zamanda r'inci tasarım değişkeninin alt ve üst limitine olan uzaklığının hata sınır terimi olan ve kullanıcı tarafından tanımlanan  $\delta$  değerinden küçük olup olmadığı kontrol edilir. Eğer küçük ise birinci satır üzerinde bu r'inci sütun atlanarak tekrar işlem başlar. Eğer bütün tasarım değişkenleri alt ve üst limitlerine  $\delta$  değerinden daha yakın ise,  $\delta$  değeri 10'a bölünür ve yeni  $\delta$  değeri  $0,1\delta$  olur. Bu bölme

işlemi  $10^{-6}\delta$  olana kadar devam eder. Eğer hala bu kısıtlama için temel olmayan tasarım değişkeni bulunamamışsa bu kısıtlama o yineleme sürecinde devre dışı bırakılır. Temel olmayan değişken yukarıda anlatıldığı gibi seçildikten sonra, bu satırdaki bütün elemanlar pivot elemanın mutlak değerine bölünür ve alttaki satırlardan çıkartılarak yeni matris elde edilir. Bu işlem temel olmayan değişkenlerinin hepsi seçilene kadar devam eder [13].

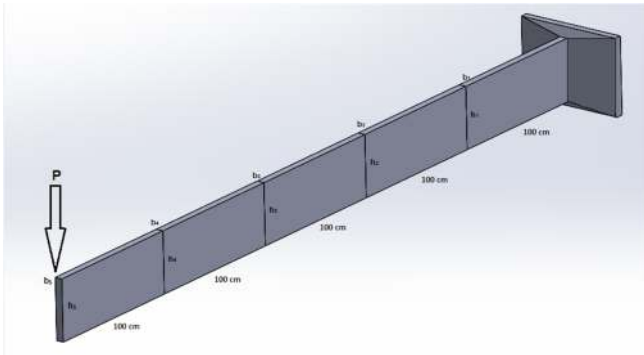
Eniyileme problemlerinin çözümünde ortaya çıkan en büyük sorun ilk tasarım değişkenlerinin olurlu olmayan bölgede olmasıdır. En uygun  $\lambda$  değerini bulduktan sonra, yeni X değerleri bulunur ve olurlu bölge kontrolü uygulanır. Chinneck yaptığı araştırmada, tüm kısıtlamaları kullanmak yerine ihlal edilmiş kısıtlamalar arasında bir olurlu bölge vektörü bulup olurlu bölgeye geçiş yapılabileceğini ifade etmiştir [14]. Bu çalışmada olurlu bölgeye geçiş için kullanılan metod, Chinneck'in yöne ve kısıtlamaya dayalı temel kısıtlama konsensüsüdür. Bu konsensüste olurlu bölge vektörü Eş. 20'de gösterilir.

$$fv_i = v_i d_i \nabla g_i(X) / \|\nabla g_i(X)\|^2 \quad (20)$$

Burada  $\nabla g_i(X)$  kısıtlamanın gradyanı,  $\|\nabla g_i(X)\|$  gradyan vektörünün boyudur. Denklemdaki  $d_i$  için eğer  $g_i(X)$  kısıtlamasını sağlamak için  $g_i(X)$  değerini arttırmak gerekiyorsa +1,  $g_i(X)$  kısıtlamasını sağlamak için  $g_i(X)$  değerini azaltmak gerekiyorsa -1 değerini alır. Denklemdaki  $v_i$  ise ihlal edilen kısıtlamadır. Olurlu bölge vektör boyu  $\|fv_i\|$  ile gösterilir. Kısıtlama konsensüs yöntemleri, olurlu bölgeye uzak olan bir noktadan olurlu bölgeye çekilmesinde etkili metotlardır. Kısıtlama konsensüs yöntemindeki ilk adım, geçerli x noktasında ihlal edilen her kısıtlama için olurlu bölge vektörünü bulmaktır. İkinci adım ise güncelleme yapmak için kullanılan konsensüs vektörüne ulaşmak amacıyla ihlal edilen kısıtlamaların tümündeki olurlu bölge vektörlerini birleştirmektir [15]. Bu konsensüsteki kısıtlamalar, çözücüye tanımlanan büyüklük teriminden ( $\varepsilon$ ) küçük ise işlem başarıyla son bulur. Eğer büyük ise olurlu bölge vektörünün boyu hesaplanır ve bu vektörün boyu gradyan boyu toleransından ( $\alpha$ ) daha az ise işlem başarıyla son bulur.

### 3. DENEK TAŞI ENİYİLEME PROBLEMİ (BENCHMARK OPTIMIZATION PROBLEM)

Çalışmanın bu kısmında geliştirilmiş olan çözücünün test edildiği literatürde bulunan denek taşı eniyileme problemi anlatılmaktadır [13]. Şekil 1'de P yükü altında eşit N adet parçaya ayrılmış destekli kiriş örneği gösterilmiştir. Destekli kirişin hacminin en düşük hale getirilmesi hedef fonksiyon olarak tanımlanmıştır. Destekli kirişin tasarım değişkenleri, her bir parça için önden görünüşte gösterilen genişlik (b) ve yükseklik (h). Şekil 1'deki görünüme göre her parçanın sağ ucundaki eğilme gerilmesi ( $\sigma$ ) 14000 N/cm<sup>2</sup>'den az olması fonksiyonel kısıtlama ve her bir parçanın yüksekliğinin genişliğin yirmi katını geçmemesi geometrik kısıtlama olarak tanımlanmaktadır. N parçalı destekli kirişin toplam uzunluğu 500 cm'dir. Bütün parçalar için Young modülleri (E) aynı olup  $2 \times 10^7$  N/cm<sup>2</sup>'dir. Eş. 21'de eylemsizlik momenti (I) değeri gösterilmiştir.



Şekil 1. Destekli kiriş (Cantilevered beam)

$$I_i = \frac{b_i h_i^3}{12} \quad (21)$$

Destekli kirişin sol ucuna uygulanan yükten dolayı oluşan eğilme momenti (M) her parça için Eş. 22, eğilme momentine karşılık gelen eğilme gerilmesi için Eş. 23 kullanılarak hesaplanır.

$$M_i = P \left[ L + l_i - \sum_{j=1}^i l_j \right] \quad (22)$$

$$\sigma = \frac{M_i h_i}{2I_i} \quad (23)$$

Eniyileme probleminde kullanılacak matematiksel ifadeler Eş. 24'te 5 adet formül ile gösterilmiştir.

$$\text{En düşük } f(X) = V = \sum_{i=1}^N b_i h_i l_i$$

$$\sigma_i - 14000 \leq 0 \quad i = 1, N$$

$$h_i - 20b_i \leq 0 \quad i = 1, N \quad (24)$$

$$b_i \geq 1 \quad i = 1, N$$

$$h_i \geq 5 \quad i = 1, N$$

Havacılıkta kanat içinde bulunan yük taşıyıcı kirişler, otomotiv endüstrisinde ana yük taşıyıcı şasiyi oluşturan kirişler ve inşaat sektöründe köprülere yüksek mukavemet sağlayan kirişler bu eniyileme probleminde incelenen destekli kirişe benzer yapılardır. Temel kiriş teorisi kullanılarak yapılan bu çalışma sayesinde bu ve benzer geometriye sahip kirişlerin ön tasarım süreci daha kısa sürede sonlandırılabilir.

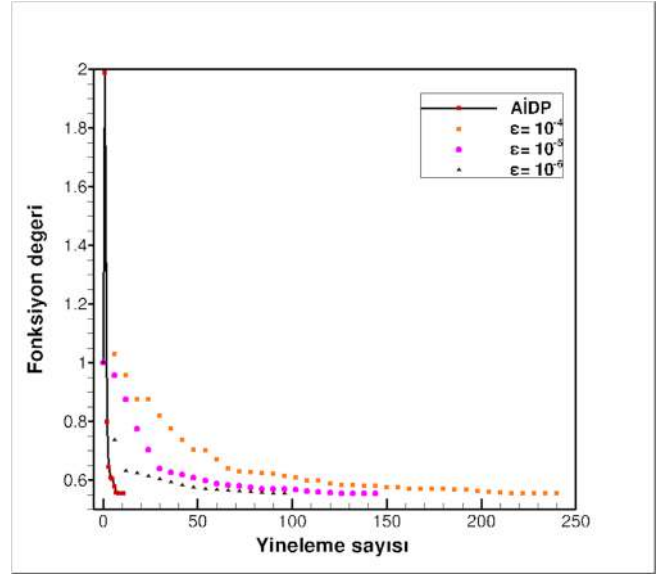
### 4. ÇÖZÜCÜ SONUÇLARININ KARŞILAŞTIRILMASI (COMPARISON OF SOLVER RESULTS)

Bu çalışmada, MATLAB programlama dili kullanılarak eniyileme çözücüsü geliştirilmiştir. Bu programlama dilinin kullanılmasının sebebi birçok matematiksel işlem için altyordamların hazır olması ve bu altyordamların kolayca çağırılması, mühendislik analiz programları ile ara yüzü sayesinde kolayca bağlantı kurulabilmesi ve en önemlisi geliştirilen çözücünün karşılaştırıldığı ve gradyan temelli eniyileme çözücülerinin arasında en iyisi olan Ardışık İkinci Derece Programlama (AİDP) eniyileme çözücüsünün altyordam olarak içinde tanımlı olmasıdır [16]. AİDP eniyileme çözücüsü, denek taşı eniyileme probleminde olduğu gibi doğrusal olmayan eniyileme problemlerini doğrusal eniyileme problemlerine dönüştürerek hedef fonksiyonu en düşük hale getirir. Bu eniyileme çözücüsünün en büyük avantajı, zaman zaman eniyileme problemini çözmekten daha uzun süre alan olurlu bölgeye geçiş sürecine ihtiyaç duymamasıdır ve bu yüzden daha az hedef fonksiyon çağırmasıdır. Dezavantajı ise doğrusal eniyileme probleminin çözümü için geçen süre ve yüksek sayıda tasarım değişkeni ve kısıtlamaya sahip eniyileme problemlerinin çözümünde zorlanmasıdır [4].

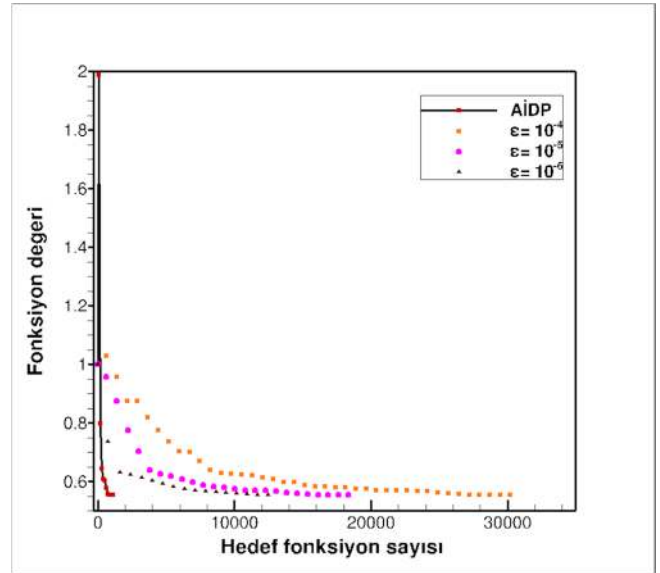
Eniyileme probleminde kullanılan eşit uzunlukta N adet parçadan oluşan kiriş için MATLAB programında bir altyordam geliştirilmiştir. Tasarım değişkeni sayısı ve eşitsizlik kısıtlamaları sayısı 2N ve kullanılan eniyileme probleminde eşitlik kısıtlaması olmadığı için bu parametre 0 olarak tanımlanmıştır. Başlangıç tasarım değişkenlerinin değerleri, genişlik (b) ve yükseklik (h) sırasıyla 5 cm ve 40 cm'dir. Bu tasarım değişkenleri için alt ve üst limitler belirtilmiştir.

Genişlik için alt limit 1 cm iken üst limit 100 cm, yükseklik için alt limit 5 cm iken üst limit 100 cm'dir. Geliştirilen bu çözücünde tasarım değişkenleri, fonksiyon ve kısıtlamalar ölçeklendirilmiştir. Tasarım değişkenleri ve bunların alt ve üst limitleri başlangıç genişlik değeri 5 ve başlangıç yükseklik değeri olan 40 ile ölçeklendirilmiştir. Hedef fonksiyon ise başlangıç hedef fonksiyon değeri olan 100000 ile fonksiyonel kısıtlama olan eğilme gerilmesi 14000 ile ve son olarak geometrik kısıtlama olan genişlik yükseklik oranı ise 60 değeriyle ölçeklendirilmiştir. İkinci bölümde detaylı bir şekilde anlatılan metotlar, kiriş probleminin çözümü için geliştirilen altyordam ile birleştirilerek eniyileme problemleri farklı sınır ve büyüklük değerlerine göre çözülmüştür.

Eniyileme problemi çözümü için Intel Core i7-6700 CPU 3.50 GHz işlemcili ve 16 GB RAM'a sahip yüksek hesaplama kapasiteli bir bilgisayar kullanılarak MATLAB R2016a sürümünde geliştirilen çözücü çalıştırılmış ve elde edilen sonuçlar aşağıdaki grafiklerde gösterilmiştir. Grafiklerde, GİGM'de  $\varepsilon, \delta, \gamma$  ve  $\zeta$  terimlerine farklı sayısal değerler tanımlanarak yineleme sayısına göre en düşük hedef fonksiyon değeri ve fonksiyon sayıları alınmıştır. Bu terimlerin yanında MATLAB içinde tanımlı AİDP (Ardışık İkinci Derece Programlama) sonuçlarına göre karşılaştırma yapılmıştır. Kiriş probleminde parça sayısı 25 iken 50 tasarım değişkeni ve 50 kısıtlama vardır. Olurlu bölge dışında kalan kısıtlamaların sayısı 7 olduğu görülmektedir. Çözücünde parça sayısı 25 iken yapılan testler sonucu elde edilen sonuçlar grafikler halinde aşağıda verilmiştir. Farklı  $\varepsilon$  büyüklük değerlerine göre yapılan testler ile elde edilen sonuçlar Şekil 2 ve Şekil 3'te grafik olarak verilmiştir.



Şekil 2. Hedef fonksiyon değerinin yineleme sayısı ile değişimi (Variation of objective function value with iteration number)



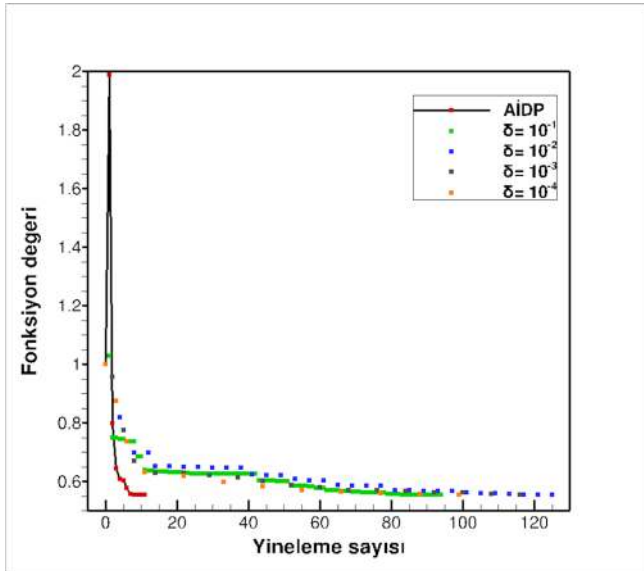
Şekil 3. Hedef fonksiyon değerinin hedef fonksiyon sayısı ile değişimi (Variation of objective function value with number of function)

Şekil 2'deki grafikte, farklı  $\varepsilon$  büyüklük değerlerine göre yapılan testler ile elde edilen sonuçlar AİDP'den elde edilen sonuçlar ile karşılaştırılmıştır. Şekil 3'te ise çözücü işlem yaparken çağırdığı fonksiyon sayısına göre karşılaştırılma yapılmıştır. Grafiklerdeki verilerin daha iyi bir şekilde anlaşılması için Tablo 1'de sonuçlar verilmiştir.

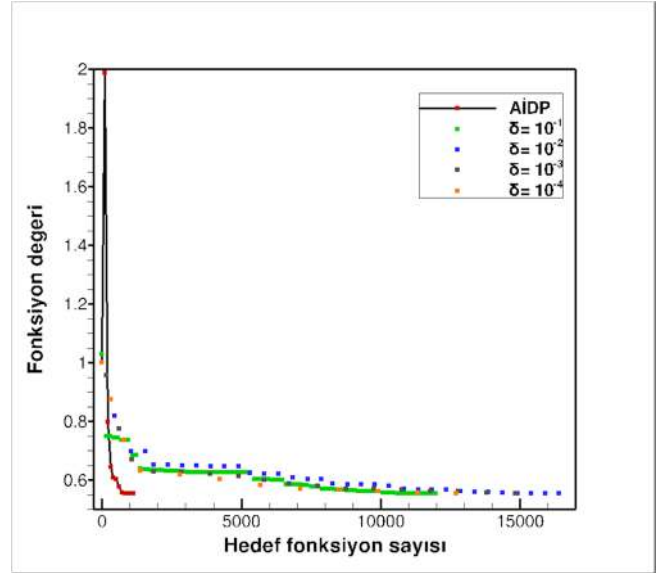
Tablo 1. Farklı  $\varepsilon$  değerlerine göre elde edilen sonuçlar  
(Results for different  $\varepsilon$  values)

	Yineleme sayısı	Fonksiyon değeri	Süre	Hedef fonksiyon sayısı
AİDP	10	0,5545	2,4362	1125
$10^{-4}$	241	0,5545	2,8883	30420
$10^{-5}$	145	0,5545	2,1005	18588
$10^{-6}$	101	0,5545	1,5656	13176

GİGM çözücüsündeki diğer terimler  $\gamma = 10^{-6}$ ,  $\zeta = 10^{-6}$  ve  $\delta = 10^{-4}$  iken üç farklı  $\varepsilon$  değeri için test yapılmıştır. Bu testler sonucu yineleme sayısı, fonksiyon değeri, süre ve fonksiyon sayısı elde edilmiştir. Buradaki amaç, geliştirilen GİGM çözücüsü ile MATLAB içinde tanımlı olan AİDP'den elde edilen sonuçların kıyaslanmasıdır. Tablo 1'de görüldüğü üzere fonksiyon değerleri aynı çıkmakta fakat yineleme sayısına ve hedef fonksiyon sayısına bakıldığında AİDP'nin daha iyi olduğu görülmektedir. Elde edilen sonuçlara göre  $10^{-6}$  değeri, AİDP ile elde edilen sonuçlara en yakın olan değer ve süre bakımından diğer değerlere göre en uygun olan değerdir. Farklı  $\delta$  sınır değerlerine göre yapılan testler ile elde edilen sonuçlar Şekil 4 ve Şekil 5'te grafik olarak verilmiştir.



Şekil 4. Hedef fonksiyon değerinin yineleme sayısı ile değişimi (Variation of objective function value with iteration number)



Şekil 5. Hedef fonksiyon değerinin hedef fonksiyon sayısı ile değişimi (Variation of objective function value with number of function)

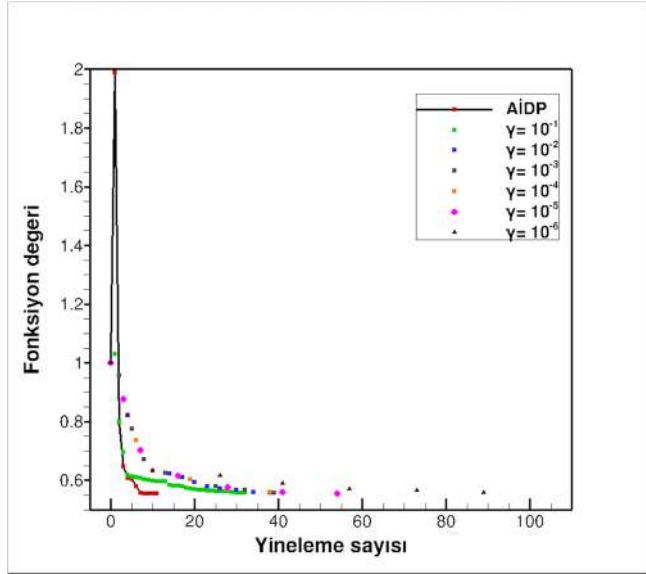
Şekil 4'teki grafikte, farklı  $\delta$  sınır değerlerine göre yapılan testler ile elde edilen sonuçlar AİDP'den elde edilen sonuçlar ile karşılaştırılmıştır. Şekil 5'te ise çağrılan hedef fonksiyon sayısına göre karşılaştırılma yapılmıştır. Grafiklerdeki verilerin daha iyi bir şekilde anlaşılması için Tablo 2'de sonuçlar verilmiştir.

Tablo 2. Farklı  $\delta$  değerlerine göre elde edilen sonuçlar  
(Results for different  $\delta$  values)

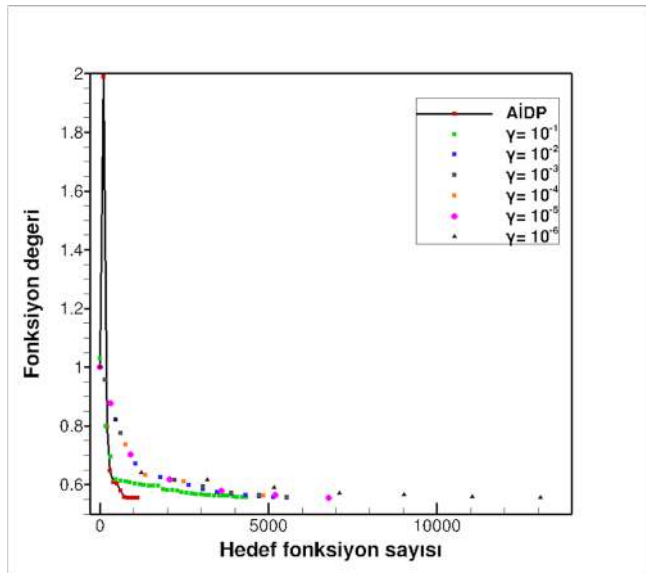
	Yineleme sayısı	Fonksiyon değeri	Süre	Hedef fonksiyon sayısı
AİDP	10	0,5545	2,4362	1125
$10^{-1}$	94	0,5545	1,3783	12063
$10^{-2}$	128	0,5545	1,8289	16501
$10^{-3}$	123	0,5545	1,7193	15800
$10^{-4}$	101	0,5545	1,5120	13176

GİGM çözücüsündeki diğer terimler  $\gamma = 10^{-6}$ ,  $\zeta = 10^{-6}$  ve  $\varepsilon = 10^{-6}$  iken  $\delta$  terimine farklı değerler verilerek test yapılmıştır. Tablo 2'de görüldüğü üzere bütün  $\delta$  değerlerinde ve AİDP'de elde edilen fonksiyon değerleri aynı çıkmıştır. Yineleme sayısı ve hedef fonksiyon sayısı bütün  $\delta$  değerlerinde çok yüksek çıkmıştır. Süre olarak, geliştirilen GİGM çözücüsünün AİDP'den daha iyi olduğu sonucuna varılmıştır. Burada bütün değerler arasında AİDP'ye en yakın olan terim  $10^{-1}$  değeridir. Şekil 4'teki grafikte,  $10^{-1}$ 'in testi yapılırken 1. yinelemeden 2. yinelemeye geçişte fonksiyon değerinde diğer değerlere göre daha fazla bir düşüş yaşandığı görülmektedir. Farklı  $\gamma$  sınır değerlerine göre yapılan testler ile elde edilen

sonuçlar Şekil 6 ve Şekil 7’de grafik olarak verilmiştir.



Şekil 6. Hedef fonksiyon değerinin yineleme sayısı ile değişimi (Variation of objective function value with iteration number)



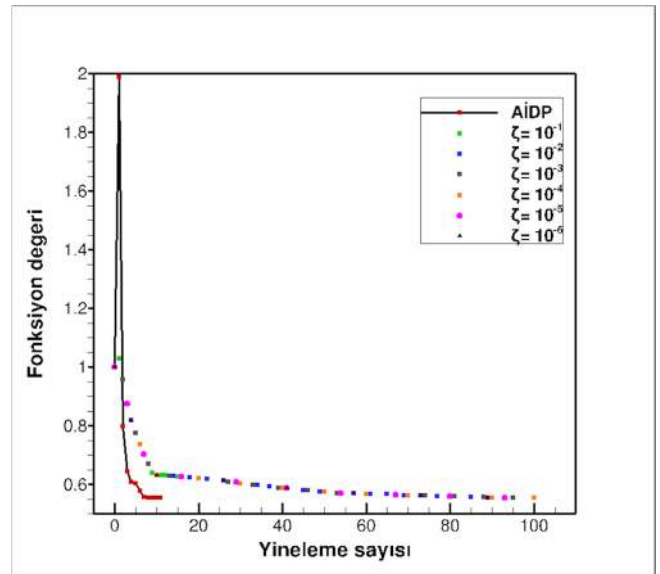
Şekil 7. Hedef fonksiyon değerinin hedef fonksiyon sayısı ile değişimi (Variation of objective function value with number of function)

Şekil 6’daki grafikte, farklı  $\gamma$  sınır değerlerine göre yapılan testler ile elde edilen sonuçlar AİDP’den elde edilen sonuçlar ile karşılaştırılmıştır. Şekil 7’de ise çağrılan hedef fonksiyon sayısına göre karşılaştırılma yapılmıştır. Grafiklerdeki verilerin daha iyi bir şekilde anlaşılması için Tablo 3’te sonuçlar verilmiştir.

Tablo 3. Farklı  $\gamma$  değerlerine göre elde edilen sonuçlar (Results for different  $\gamma$  values)

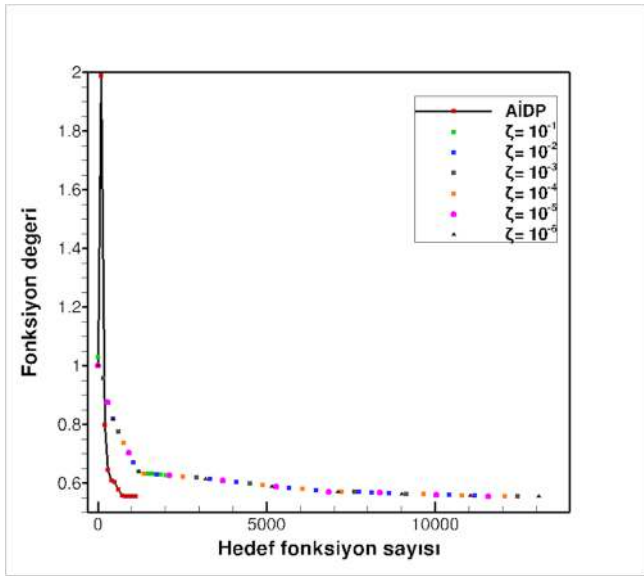
	Yineleme sayısı	Fonksiyon değeri	Süre	Hedef fonksiyon sayısı
AİDP	10	0,5545	2,4362	1125
$10^{-1}$	32	0,5575	0,7319	4435
$10^{-2}$	41	0,5545	0,8813	5770
$10^{-3}$	43	0,5545	0,9660	6042
$10^{-4}$	43	0,5545	0,9272	6042
$10^{-5}$	55	0,5545	1,0818	7518
$10^{-6}$	101	0,5545	1,4974	13176

GİGM çözücüsündeki diğer terimler  $\zeta = 10^{-6}$ ,  $\delta = 10^{-4}$  ve  $\varepsilon = 10^{-6}$  iken  $\gamma$  terimine farklı değerler verilerek test yapılmıştır. Fonksiyon değerleri açısından elde edilen sonuçlara göre  $\gamma$  değeri  $10^{-1}$  değeri haricinde bütün değerlerde AİDP ile elde edilen değere ulaşıldığı görülmektedir. Süre bakımından en uygun olanı  $10^{-2}$  değeridir. Bu sonuçlara göre  $10^{-2}$  değeri, AİDP ile elde edilen sonuçlara en yakın olan değerdir. Farklı  $\zeta$  büyüklük değerlerine göre yapılan testler ile elde edilen sonuçlar Şekil 8 ve Şekil 9’da grafik olarak verilmiştir.



Şekil 8. Hedef fonksiyon değerinin yineleme sayısı ile değişimi (Variation of objective function value with iteration number)





Şekil 9. Hedef fonksiyon değerinin hedef fonksiyon sayısı ile değişimi (Variation of objective function value with number of function)

Şekil 8'deki grafikte, farklı  $\zeta$  büyüklük değerlerine göre yapılan testler ile elde edilen sonuçlar AİDP'den elde edilen sonuçlar ile karşılaştırılmıştır. Şekil 9'da ise çözücü işlem yaparken çağırdığı fonksiyon sayısına göre karşılaştırılma yapılmıştır. Grafiklerdeki verilerin daha iyi bir şekilde anlaşılması için Tablo 4'te sonuçlar verilmiştir.

Tablo 4. Farklı  $\zeta$  değerlerine göre elde edilen sonuçlar (Results for different  $\zeta$  values)

	Yineleme sayısı	Fonksiyon değeri	Süre	Hedef fonksiyon sayısı
AİDP	10	0,5545	2,4362	1125
$10^{-1}$	16	0,6269	0,4503	2235
$10^{-2}$	92	0,5559	1,4049	12043
$10^{-3}$	101	0,5545	1,5550	13176
$10^{-4}$	101	0,5545	1,4958	13176
$10^{-5}$	101	0,5545	1,5371	13176
$10^{-6}$	101	0,5545	1,5035	13176

GİGM çözücüsündeki diğer terimler  $\gamma = 10^{-6}$ ,  $\delta = 10^{-4}$  ve  $\varepsilon = 10^{-6}$  iken farklı  $\zeta$  değerleri verilerek yapılan testlerde  $10^{-3}$ ,  $10^{-4}$ ,  $10^{-5}$  ve  $10^{-6}$  değerlerinde sonuçların aynı çıktığı gözlemlenmiştir. Bu değerler arasında süre bakımından en uygun olanı  $10^{-4}$  değeri AİDP ile elde edilen sonuca en yakın olduğu saptanmıştır. Süre olarak bütün değerler içinde en uygun olanı  $10^{-1}$  olmasına rağmen fonksiyon değeri bakımından AİDP'ye uzak kalır. Bütün grafiklerde görüldüğü üzere AİDP ve GİGM çözücüsünün başlangıç değerleri 1'dir. Yineleme sonucunda bu

değer, AİDP'de 1,9882 olurken GİGM'de 1,0288 olur ve sonraki yinelemede değerler 1'in altına düşerek olurlu bölgeye geçiş sağlanır. AİDP'de bu değer çok yüksek çıktığı için olurlu bölgeye geçişte GİGM çözücüsünün AİDP'den daha iyi olduğu sonucuna varılmıştır.

## 5. SONUÇLAR (RESULTS)

Bu çalışmada, eniyileme problemi çözümü için genel indirgenmiş gradyan metodu kullanılarak eniyileme çözücüsü geliştirilmiştir. Bu çözücünün geliştirilmesinde MATLAB programlama dili kullanılmıştır. Çalışmada anlatılan eniyileme çözücüsünde parça sayısı 25'e eşit olduğunda testler yapılarak sonuçlar alınmıştır. Burada farklı parametreler ile yapılan testler sonucu elde edilen değerler göre çözücü  $\gamma = 10^{-2}$ ,  $\delta = 10^{-1}$ ,  $\varepsilon = 10^{-6}$  ve  $\zeta = 10^{-4}$  iken en uygun olduğu görülmüştür. Geliştirilen eniyileme çözücüsündeki bu parametreler ile yapılan test sonuçlarının, ardışık ikinci derece programlama ile elde edilen sonuçlara yaklaştığı gözlemlenmiştir. Elde edilen sonuçlar, ardışık ikinci derece programlama kullanıldığı takdirde çok daha az hedef fonksiyon çağırdığı fakat genel indirgenmiş gradyan metodunun içinde tanımlanan parametrelerin doğru seçilmesi halinde çözüm süresi bakımından avantajlı olduğunu ifade etmektedir. Bu çalışmada kullanılan denek taşı problemin 500 parçalı halinin çözümünün yapıldığı başka bir çalışmada, geliştirilmiş olan genel indirgenmiş gradyan metodu çözücüsü problemi ardışık ikinci derece programlama çözücüsünden yaklaşık 700 saniye daha kısa bir sürede çözmüştür [17].

## KAYNAKLAR (REFERENCES)

- [1] S. Kiranyaz, T. İnce ve M. Gabbouj, *Multidimensional Particle Swarm Optimization for Machine Learning and Pattern Recognition*, Berlin, Heidelberg: Springer, 2014.
- [2] J. Dutta, "Optimization Theory - A Modern Face of Applied Mathematics," *Indian Institute of Technology Kanpur*, cilt6, no. 3, 2004.
- [3] S. S. Rao, *Engineering Optimization: Theory and Practice*, 4th ed., New Jersey: John Wiley & Sons, 2009.

- [4] L. S. Lasdon ve A.D. Waren, “Large Scale Nonlinear Programming,” *Computers & Chemical Engineering*, cilt 7, no.5, pp. 595-604, 1983.
- [5] N. Bozkurt ve S. Yazıcıoğlu, “Yapı Sistemlerinin MATLAB Optimizasyon Araç Kutusu ile Optimum Boyutlandırılması,” *BEÜ Fen Bilimleri Dergisi*, cilt 4, no. 2, pp. 189-197, 2015.
- [6] B.C. Song, Y.C. Park, S.W. Kang, K.H. Lee, “Structural Optimization of An Upper Control Arm, Considering the Strength,” *Institution of Mechanical Engineers*, 223, 727-735, 2009.
- [7] F. Faluyi ve C. Arum, “Design Optimization of Plate Girder Using Generalized Reduced Gradient and Constrained Artificial Bee Colony Algorithms,” *International Journal of Emerging Technology and Advanced Engineering*, cilt 2, no. 7, pp. 304-312, 2012.
- [8] F. Fedghouche ve B. Tiliouine, “Minimum Cost Design of Reinforced Concrete T-Beams At Ultimate Loads Using Eurocode2,” *Engineering Structures*, cilt 42, pp. 43-50, 2012.
- [9] D. S. Körpe ve S. Özgen, “Morphing Wing Optimization For Steady Level Flight,” *Proceedings of the Institution of Mechanical Engineers Part G: Journal of Aerospace Engineering*, cilt 231, no. 13, pp. 2317-2330, 2017.
- [10] E. D. Klerk, C. Roos ve T. Terlaky, *Nonlinear Optimization (CO 367)*, Waterloo, 2006.
- [11] D. Körpe, “Aerodynamic Modelling and Optimization of Morphing Wings,” Doktora Tezi, Orta Doğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2014.
- [12] A. Ravindran, K. M. Ragsdell ve G. V. Reklaitis, *Engineering Optimization: Methods and Applications*, New Jersey: John Wiley & Sons, 2006.
- [13] G. N. Vanderplaats, *Multidiscipline Design Optimization*, Monterey, Kaliforniya: Vanderplaats Research and Development, 2007.
- [14] W. Ibrahim ve J. W. Chinneck, “Improving Solver Success in Reaching Feasibility for Sets of Nonlinear Constraints,” *Computers and Operations Research*, cilt 35, no. 5, pp. 1394-1411, 2008.
- [15] J. W. Chinneck, *Feasibility and Infeasibility in Optimization: Algorithms and Computational Methods*, New York: Springer, 2008.
- [16] S. Hong, “Design of Power Systems for Extensible Surface Mobility Systems on the Moon and Mars,” Yüksek Lisans Tezi, Massachusetts Institute of Technology, 2007.
- [17] M. S. Toplu, “Genel İndirgenmiş Gradyan Metodu ile Eniyileme Çözücüsü Geliştirilmesi,” Yüksek Lisans Tezi, Türk Hava Kurumu Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2017.