





Bir Boyutlu Stok Kesme Probleminin Önerilen Sezgisel Algoritma Kullanılarak Çözümü

Solving of One-Dimensional Stock Cutting Problem Using Proposed Heuristic Algorithm

Serkan TOPAL^{1*} , Pınar YILDIZ KUMRU² 

¹ Endüstri Mühendisliği, Kocaeli Üniversitesi, Kocaeli, Türkiye, **Orcid:** 0009-0004-1143-6819

² Endüstri Mühendisliği, Kocaeli Üniversitesi, Kocaeli, Türkiye, **Orcid:** 0000-0002-6729-7721

Araştırma Makalesi

Gönderilme Tarihi : 11/05/2023

Kabul Tarihi : 11/12/2023

Anahtar Kelimeler

Bir Boyutlu Stok Kesme Problemi
Sezgisel Algoritma
Tavlama Benzetimi
Tam Sayılı Matematiksel Programlama

Özet

Bu çalışmada, bir boyutlu stok kesme problemi ele alınmıştır. Kesme işlemi sayısının fazla olması bir boyutlu kesme probleminin çözümünde hesaplama zorluklarını ortaya çıkarmaktadır. Kesme işlemlerinin yoğun yapıldığı ve hesaplama zorluklarının yaşandığı bir tesiste daha verimli bir kesme işlemi sağlanması hedeflenmiştir. Bu tesiste metal kesme işlemleri referans alınmış ve test verileri hazırlanarak farklı yöntemler aracılığıyla problemler çözülmüştür. Ele alınan kesme problemi için geliştirilen sezgisel algoritma, tavlama benzetimi, tam sayılı matematiksel programlama ve Real Cut 1D hazır paket programıyla çözümler karşılaştırılmıştır. Sonuçlar sarf edilen ana malzeme, fire oranları ve çözüm süreleri açısından değerlendirilmiştir. Kullanılan yöntemlerin sarf edilen ana malzeme ve fire oranı bakımından yüksek verimliliğe sahip oldukları görülmüştür. Hesaplama süresi açısından bakıldığında geliştirilen sezgisel algoritma ve Real Cut 1D programının diğer yöntemlere göre daha iyi sonuçlar ürettiği tespit edilmiştir. Geliştirilen sezgisel algoritma çözüm, fire oranı ve hesaplama süresi açısından gayet iyi sonuçlar üretmesi nedeniyle tercih edilebilir yöntemlerden biri olabileceği sonucuna ulaşılmıştır.

Research Paper

Received Date : 11/05/2023

Accepted Date : 11/12/2023

Keywords

One Dimensional Stock Cutting Problem
Heuristic Algorithm
Simulation Annealing
Integer Mathematical Programming

Abstract

In this study, a one-dimensional cutting stock problem has been addressed. The abundance of cutting operations poses computational challenges in solving one-dimensional cutting stock problems. The aim is to achieve a more efficient cutting process in a facility where cutting operations are intensive and computational challenges are encountered. Metal cutting operations in such a facility were considered as a reference, and test data were prepared to solve problems through various methods. A heuristic algorithm developed for the cutting problem, simulated annealing, an integer mathematical programming, and the Real Cut 1D ready-made package program were compared for solutions. The results were evaluated in terms of the main material used, waste ratios, and solution times. It was observed that the methods used were highly efficient in terms of the main material used and waste ratio. In terms of computation time, the developed heuristic algorithm and the Real Cut 1D program were found to produce better results compared to other methods. The developed heuristic algorithm is considered one of the preferable methods due to its good results in terms of solution, waste ratio, and computation time.

1. Giriş

Stok kesme probleminin temeli, belirli bir talebi karşılamak amacıyla bilinen boyuttaki malzemeleri daha küçük parçalara ayırmanın en iyi yolunu bulmayı içerir. Genel anlamda optimize edilecek amaç, malzeme israfının en aza indirmesini kapsar [1].

Plansız yapılan stok kesme işlemleri neticesinde olması gerekenden fazla malzemenin sarf edilmesi söz konusu olabilir. Aynı zamanda kesme işlemi sonrasında malzemenin kullanılmayan (fire) kısımları olabilir. Az sayıda kesme işlemi içeren problemlerin çözümü nispeten kolaydır. Ancak kesme işlem sayısı arttıkça problemi çözmek daha zor hale gelir. Problemin zor olması

* Sorumlu Yazar (Corresponding Author): srknea@yahoo.com



optimizasyonun sağlanmasını güçleştirir. Bundan dolayı zor ve karmaşık problem olarak adlandırılan stok kesme problemlerinin çözümü için en iyi yöntemi geliştirme çalışmaları devam eden araştırma konularından biridir [2]. Özellikle büyük veriler ile çalışılıyorsa kesin çözüme ulaşılanın imkansız olabileceği problem türleri de vardır. Çünkü arama yapılan uzay büyüklüğü arttıkça problem çözümleri olasılık temelli teknikler üzerinde kurulmasına ihtiyaç duyulur [3].

Stok kesme problemleri konusunda yapılan çalışmalar literatürde kapsamlı şekilde yer almaktadır ve problemler farklı uygulama örneklerine göre farklı amaç ve kısıtlara sahip olabilmektedir. Problemi farklılaştıran kriterler; problemin boyutu, problemin amaç fonksiyonu, ana malzeme çeşitliliği, küçük malzeme çeşitliliği ve şekli gibi çeşitli sınıflara ayrılmaktadır. Boyut, ilgili problemin boyutlarının (1B - 2B - 3B) olmasını dikkate alır. Problemin amaç fonksiyonu, çıktı değerinin optimize edilmesi anlamına gelir. Ana malzeme çeşitliliği; malzemenin boyutlarının, sayısının ve şekillerinin değişmesini ifade eder. Küçük malzeme çeşitliliği ve şekli; aynı veya farklı malzeme şekillerinin, boyutlarının ve sayısını tanımlamakta kullanılır [4]. Birçok kısıt ve amacın bir araya geldiği çok kriterli problemlerin çözümü için araştırma ve uygulama çalışmalarında farklı metodolojiler ele alınmakta, çeşitli formül ve yöntemler önerilmektedir [5]. Algoritmalar, formüller ve yöntemler kesin çözüm içeren matematiksel modellerde kullanılacağı gibi, yaklaşık çözümler sunan metasezgisel ve sezgisel metotlarda da kullanılabilir [6].

Gemi inşa, bakım ve onarım operasyonlarını içeren bir tersanede yapılan stok kesme işlemleri bu araştırmanın temel konusunu oluşturmaktadır. Tersanede imalat sürecinde levha, profil gibi malzemelerden talep miktarına bağlı olarak farklı ebatlarda, uzunluklarda ve miktarlarda kesme işlemleri operasyonel faaliyetlerde sıkça uygulandığı görülmüştür. Plansız yapılan kesme işlemleri sonrasında malzemenin kullanılmayan kısımları olduğu, başka bir ifadeyle fire (atıl) miktarlar oluştuğu tespit edilmiştir. Gereksiz malzeme sarfının ve fire miktarının azaltılması veya yok edilmesi malzeme tüketiminden kaynaklanan maliyetlerin azaltılmasına katkı sağlanacağı değerlendirilmiştir.

Ele alınan çalışma örneğinde, bir boyutlu stok kesme işleminin uygulandığı faaliyetler göz önüne alınmıştır. Bir boyutlu kesme problemlerin çözümünde uygulanabilecek yöntemler literatür taramaları neticesinde tespit edilmiştir. Yapılan literatür taramasında, metasezgisel yöntemler içerisinde tavlama benzetimi, genetik algoritma, karınca algoritması, tabu arama gibi yöntemler kullanıldığı anlaşılmıştır. Ayrıca kesin çözümler üretebilen matematiksel programlama modelleri, yaklaşık çözümler sunan sezgisel yöntemler, hazır paket programları, araştırmacı tarafından geliştirilen teknikler ve bu tekniklerin

farklı yöntemler ile bir arada kullanıldığı hibrit çözüm yaklaşımları olduğu görülmüştür.

Bu çalışmada, gerçek hayat örneğinden yola çıkılarak bir boyutlu stok kesme işlemini içeren test veri setleri hazırlanmıştır. Bu test örnekleri tavlama benzetimi, geliştirilen sezgisel yöntem, tam sayılı matematiksel programlama ve hazır paket program kullanılarak çözülmüştür. Yöntemlerin güçlü ve zayıf yönleri, çözümler üzerinden belirlenmiştir. Sonuç bölümünde, yöntemlerin genel performans değerlendirmesi yapılmıştır.

2. Literatür Taraması

Yapılan literatür taramalarında birçok stok kesme problemi çalışma konusu olarak ele alındığı görülmüştür. Bu bölümde bu çalışmalardan bazılarına yer verilmiştir.

Sanchez ve diğerleri çalışmalarında, stok kesme problemlerini çözmek için lineer programlama, tavlama benzetimi ve sürü optimizasyonu yöntemlerini içeren bir çalışma yapmışlardır. Metasezgisel yöntemler, örnek alınan deney çalışmalarında her zaman kesin yöntemler kadar global iyi çözümler üretmese de başarılı sonuçlara ulaşabileceği sonucuna varmışlardır [6]. Baykasog ve Özbel çalışmalarında, mermer blok kesme işlemi için geliştirdikleri matematiksel model ile metasezgisel yöntemi karşılaştırarak analiz etmişlerdir [7]. Saraç ve Sağır çalışmalarında, kesme problemi için tam sayılı matematiksel programlama modelleri kullanarak test problemlerini çözmüş ve elde edilen sonuçlarla karşılaştırma yapmışlardır [8]. Velasco ve Uchoa çalışmalarında, malzemeyi döndürme koşulu göz önüne alınarak tam sayı matematiksel programlamadan elde edilen yeni bir yöntemle iki boyutlu stok kesme problemini detaylı bir şekilde analiz etmişlerdir [9]. Tung ve diğerleri çalışmalarında, kumaş dikim işlemleri sırasında atıl parçaların olmaması veya en az seviyede kalması amacıyla kumaş döşemelerinden yapılacak kesim işleminin optimum seviyede olmasını hedeflemişlerdir. Kesim yapılacak parça sayısının fazla olması ve farklı şekillerde kesim yapılması gerektiğinden zor problem ortaya çıkmıştır. Makul bir sürede uygun bir kesim programı oluşturmak amacıyla tavlama benzetimi algoritmasından faydalanmışlardır [10]. Kasımbeyli ve Demirci araştırmalarında, stok kesme problemiyle ilgili üç farklı amaç içeren bir tam sayılı matematiksel model geliştirmişlerdir. Fire, ana malzeme çeşit sayısı ve talep fazlası kesim miktarını minimize edecek formül geliştirmişlerdir [11]. Pierini ve Poldi çalışmalarında, hammadde olan kâğıttan yapılacak kesme işlemlerinin planlanması amacıyla kolon türetme yöntemi ve bu yöntemden geliştirilen bir matematiksel model önermişlerdir [5]. Muter ve Sezer çalışmalarında, ele aldıkları stok kesme probleminin çözümü için kolon türetme

yöntemini temel alarak, tasarladıkları satır ve sütun oluşturma algoritmasıyla stok kesme problemlerinin çözümlerini değerlendirmişlerdir [12]. Tanır ve diğerleri çalışmalarında, çelik endüstrisinde karşılaşılan kesme problemi üzerinde çalışmışlardır. Geliştirilen sezgisel model bir işletmenin stok kesme problemi üzerinde başarılı sonuçlar vermiştir [13]. Vishwakarma ve Powar çalışmalarında, bir boyutlu stok kesme problemini incelemişlerdir. Kesme kaybını en aza indirecek kesme planı oluşturmak için matematiksel ve metasezgisel yöntemleri kullanmışlardır [14]. Parreno ve Alvarez-Valdes'in çalışmalarında, giotin kullanılarak yapılan iki boyutlu stok kesme problemini, geliştirdikleri tam sayılı doğrusal programlamayla çözmüşlerdir [15]. Wuttke ve Heese çalışmalarında, tekstil endüstrisinde iki boyutlu bir stok kesme problemini sezgisel ve matematiksel yöntemler kullanarak optimal sonuçlar elde etmişlerdir [16].

3. Malzeme ve Yöntem

Bu bölümde, çalışmada kullanılan tavlama benzetimi, tam sayılı matematiksel yöntem ve sezgisel metot hakkında genel bilgiler verilmiştir.

3.1. Tavlama Benzetimi

Tavlama, malzemeyi bir miktar ısıttıktan sonra yavaş yavaş soğutma işlemine verilen addır. Tavlama ile malzeme rahatlatılır, yumuşatılır ve malzemenin iç yapısı daha kullanılabilir duruma getirilir. Tavlama benzetimi algoritması ise katı maddelerin önce ısıtılması ve daha sonra kademeli bir şekilde soğutulması temeline dayanır. Tavlama benzetimi algoritması özellikle hesaplama alanında kullanılan metasezgisel algoritmalarından biridir ve algoritmanın hedefi, belirli bir problem için genel bir iyileştirme (global optimizasyon) sağlamaktır.

Benzetilmiş tavlama algoritması, yerel arama algoritmalarının yerel minimum veya maksimuma takılıp algoritmayı durdurması ve global sonucu aramamasından kaynaklanan eksikliğe çözüm sağlamak için geliştirmiştir. Bu algoritmanın global arama özelliği sayesinde diğer yerel arama algoritmalarından ayrılır [17]. Benzetilmiş tavlama algoritması global arama özelliğini oluşturan parametrelere sahiptir ve bu parametreler şunlardır: başlangıç sıcaklık değeri, her sıcaklık değerinde tekrarlanan iterasyon sayısı, sıcaklık azaltmak için soğutma oranı ve algoritmayı durdurma koşuldur [18].

Başlangıç sıcaklık değeri, çözüme başlamadan önce tanımlanan maksimum sıcaklık seviyesini ifade eder. Sıcaklığın yeterince yüksek bir seviyeden başlatılması tüm global değerlerin aranması açısından önemlidir. Düşük sıcaklık değeri belirlenmesi, aramanın yerel en iyide

kalmasına neden olabilir ve bundan dolayı başlangıç çözümde yüksek bir sıcaklık değerinden başlanıp arama ilerledikçe kademeli bir şekilde sıcaklık değeri azaltılır [19]. İterasyon sayısı, her bir mevcut sıcaklık değerinde aramanın kaç kere tekrar etmesi gerektiğini ifade eder. Soğutma katsayısı, sıcaklık değerini düşürmek için kullanılan katsayıdır. Soğutma katsayı değeri 0 ile 1 arasında seçilir. Katsayının 1'e yakın seçilmesi soğumanın daha yavaş olmasını sağlar [20]. Durma kriteri ise belirlenen kriterlere göre algoritmanın durdurulması anlamına gelir. Örneğin, minimum sıcaklık değerine ulaşıldığında algoritmanın durması sağlanabilir [17]. Algoritmanın durması için belli bir sürenin bitmesi, döngü sayısının tamamlanması veya herhangi bir iyileşme elde edilememesi gibi hedefler de belirlenebilir.

Tavlama benzetimi algoritması probleme uygulandığında, bir başlangıç çözüm üretilir daha sonra çözüme aday bir komşu çözüm girer. Komşu çözüm, mevcut çözümden daha iyi olduğunda kabul edilir. Eş. (1)'de ifade edilmiştir. Minimize amaçlı problemlerde düşük değer iyi, maksimize amaçlı problemlerde yüksek değer iyi kabul edilir. Komşu çözüm, mevcut çözümden iyi değilse Eş. (2)'ye göre kabul koşuluna bakılır. Kabul koşuluna göre yeni çözüm kabul de edilebilir, ret de edilebilir. Çözümün kabul olasılığı Eş. (3) ve Eş. (4)'e istinaden kontrol edilir. Sıcaklık belli bir soğuma oranında düşürülür ve arama devam eder. Tekrar komşu çözümler oluşturulur en son algoritma bitiş sıcaklık değerine ulaştığında algoritma durur. Algoritma tamamlandığında, o zamana kadar bulunan en iyi çözüm ekrana yazdırılır [21].

$$p = 1, \text{ eğer } f(y) \leq f(x) \quad (1)$$

$$p = e^{-\frac{(f(y)-f(x))}{t}}, \text{ diğer durumda} \quad (2)$$

$$e^{-\frac{(f(y)-f(x))}{t}} > r, \text{ kabul} \quad (3)$$

$$e^{-\frac{(f(y)-f(x))}{t}} < r, \text{ ret} \quad (4)$$

p: Boltzmann olasılık durumu

e: Euler sayısı

r: 0 ile 1 arasında rassal değer

t: Sıcaklık değeri

Tavlama benzetimi algoritması (akışı) için temel hatlarıyla aşağıdaki adımlar tanımlanabilir.

Adım 1. Başla.

Adım 2. Başlangıç için bir "x" çözümü üret.

Adım 3. "x" çözüm sonucundan, komşu çözüm olan "y" çözümünü üret.

Adım 4. "y" değeri "x" değerinden iyi mi? Evet ise adım 6'ya git. Hayır ise adım 5'e git.

Adım 5. Kabul edilme koşulunu kontrol et. Kabul edildiyse adım 6'ya git. Kabul edilmediyse adım 7'ye git.

Adım 6. $x = y$ olarak ata.

Adım 7. Sıcaklığı azalt.

Adım 8. Durma koşulunu kontrol et.

Adım 9. Durma koşulu sağlandı mı? Evet ise döngüden çık, hayır ise döngüye devam et.

Adım 10. Sonucu raporla.

Adım 11. Bitiş.

3.2. Tam Sayılı Matematiksel Programlama

Bu bölümde bir boyutlu stok kesme problemlerinde kullanılan matematiksel model sunulmuştur.

Kümelere:

$i =$ Talep edilen çubuk parça uzunluk sayısı, $i = 1 \dots L$

$j =$ Ana çubuk sayısı, $j = 1 \dots N$

Parametreler:

$N =$ Toplam ana çubuk sayısı

$l =$ Talep edilen çubuk parçasının uzunluğu, $l \in L$

$r =$ Her talep edilen çubuk parçası uzunluğuna karşılık gelen talep sayısı, $r \in L$

$u =$ Ana çubuğun uzunluğu

$M =$ Büyük bir pozitif sayı

Karar değişken tanımları:

$x_j = \begin{cases} 1 & \text{eğer } j\text{'nci çubuk kullanılıyorsa } (\forall i \in N) \\ 0 & \text{diğer durumlarda} \end{cases}$

$y_{ij} = j\text{'nci çubuktan kesilen } i\text{'nci talep uzunluğunun sayısı}$
 $\forall i \in N, \forall j \in L$

Matematiksel Model:

Yukarıda belirtilen karar değişkenleri ve parametrelere göre oluşturulan matematiksel model aşağıdaki gibidir.

$$Z_{\min} = \sum_{j=1}^N x_j \quad (5)$$

Kısıtlayıcılar;

$$\sum_{i=1}^L l_i y_{ij} \leq u \quad (j = 1, \dots, N) \quad (6)$$

$$\sum_{j=1}^N l_i y_{ij} = r_i \quad (i = 1, \dots, L) \quad (7)$$

$$y_{ij} - M * x_j \leq 0 \quad (i = 1, \dots, L, j = 1, \dots, N) \quad (8)$$

$$x_j \geq x_{j+1} \quad (j = 1, \dots, N - 1) \quad (9)$$

$$x_j \in \{0,1\}, y_{ij} \geq 0 \text{ ve tam sayı,} \quad (10)$$

Eş (5), modelde amaç fonksiyonudur ve kullanılan ana çubuk sayısını minimize etmeyi gösterir. Eş (6), bir çubuktan kesilen parçaların toplam uzunluğunun, çubuğun boyunu geçemeyeceğini ifade eder. Eş (7), tüm talebin karşılanması gerektiğini belirtmektedir. Eş (8), bir sabit yük kısıtı olup j 'nci çubuk kullanılmaz ise haliyle bu çubuktan herhangi bir parçanın kesilemeyeceğini belirtmektedir. Eş (8)'de yer alan M değeri, sabit yük kısıtını olurlu bölgeyi değiştirmeden aktif yapabilecek büyük bir sayıyı ifade etmektedir. M 'nin yeteri kadar büyük seçilmesi modelin performansını büyük ölçüde etkilemektedir. Neyse ki stok kesme probleminde M 'yi makul bir değer olarak her bir i ve j çifti için bulabilecek bir yöntem vardır. Eş. (9), stok kesme probleminde en büyük dezavantajlarından birisi olan simetriyi önlemek için modele eklenmiştir. Eş. (10), çözüme girecek çubukların ikili değişkenle ve çözüme girecek çubuklardan kesilen talep uzunluklarının tam sayı kısıtı ile yönetilmesidir.

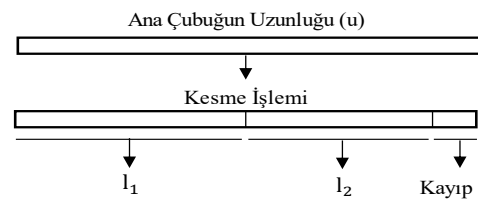
3.3. Sezgisel Yöntemler

Problem çözüm tekniği olarak kullanılan sezgisel metodlar, bir arama yöntemi olarak herhangi bir neticenin bulunmasını garanti etmez. Sezgisel yöntem iyi olan sonucu, başka bir ifadeyle en iyi çözüme yakın çözümleri üretmeye çalışan bir metottur. Sezgisel yöntemlerin çeşitli yaklaşımlarla çözüm bulmaya çalışan teknikleri vardır. Bu tekniklerden bir kısmı; an az kalan sezgiseli, aşağı sol sezgiseli, temas yüzeyi sezgiseli, ilk uygun azalan sezgiseli gibi çeşitli yöntemlerden bazıları olarak tanımlanabilir [22].

4. Uygulama

4.1. Problem Tanımı

Ana çubuktan (u), aynı veya farklı uzunluklarda daha küçük çubuk parçaların (l) kesilip alınması sonucunda ana çubuktan geriye kayıp (atık) uzunluğu kalabilir. Bir boyutlu stok kesme problemine ilişkin Şekil 1'de örnek kesme işlemi sonucunda ortaya çıkan kayıp miktarını simgeleyen uzunluk gösterilmiştir. Bu kaybın oluşmasını engellemek hedeflenir. Aynı zamanda talep miktarı karşılanırken en az sayıda (u)'nun kullanılması en önemli husustur. Çünkü asıl maliyet sarf edilen ana malzemeden kaynaklanır.



Şekil 1. Kesim kaybının şematik olarak gösterimi

Stok kesme işlemi, herhangi bir metot uygulamadan gerçekleştirildiğinde ana çubuğun daha fazla sarf edilmesine ve yüksek miktarda fire oluşmasına sebep olabilir. Etkin bir metot belirleyip en az sayıda ana malzeme sarfının sağlanması ve fire miktarının yok edilmesi veya en az seviyeye indirilmesi malzeme sarf verimliliği açısından önemlidir. Kullanılan yöntem aynı zamanda makul süre içerisinde çözüm sunması beklenir. Çözüm süresinin aşırı uzaması yöntemi pratik olmaktan uzaklaştırabilir.

4.2. Test Verilerinin Oluşturulması

Bir tersanenin metal kesim atölyesindeki bir boyutlu stok kesme problemi referans alınarak Tablo 1’de gösterilen

on adet test veri seti hazırlanmıştır. Test veri setleri hazırlanırken toplam talep edilen çubuk parçası uzunluğu ve toplam talep sayısı bakımından küçük veri setinden daha büyük veri setine doğru gidilmiştir. Problemlerin çözümünde, veri sayısı büyüklüğüne göre kullanılan yöntemlerin nasıl bir performans göstereceğini tespit etmek amaçlanmıştır.

Tablo 1’in ilk sütunlarında test verilerinin adı ve diğer sütunlarda sırasıyla ana çubuğun uzunluğu (u), talep edilen çubuk parçasının uzunluğu (l), her talep edilen çubuk parçasına karşılık gelen talep sayısı ve toplam talep miktarı gösterilmiştir.

Tablo 1. Test verileri

Test Veri Adı	Ana Çubuk (cm)	Talep Edilen Çubuk Parçasının Uzunluğu (cm)	Her Talep Edilen Çubuk Parçası Uzunluğuna Karşılık Gelen Talep Sayısı (adet)	Toplam Talep (adet)
Test - 1	200	(25, 30, 35, 40, 45, 50)	(27, 17, 8, 14, 15, 19)	100
Test - 2	200	(25, 30, 35, 40, 45, 50, 55, 60)	(41, 37, 18, 27, 29, 17, 12, 19)	200
Test - 3	200	(25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100)	(31, 22, 17, 26, 16, 35, 11, 18, 10, 17, 24, 8, 9, 7, 14, 35)	300
Test - 4	200	(25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 115, 120, 125, 130, 135, 140, 145, 150)	(40, 21, 16, 25, 18, 31, 13, 18, 9, 17, 24, 20, 19, 17, 14, 35, 12, 15, 22, 25, 20, 14, 8, 13, 14, 20)	500
Test - 5	200	(25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 115, 120, 125, 130, 135, 140, 145, 150)	(65, 35, 18, 48, 24, 45, 22, 23, 24, 27, 36, 28, 8, 32, 28, 55, 17, 25, 24, 35, 33, 24, 19, 17, 12, 26)	750
Test - 6	200	(25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 115, 120, 125, 130, 135, 140, 145, 150)	(95, 36, 19, 48, 22, 85, 22, 42, 24, 35, 46, 28, 22, 32, 26, 95, 13, 25, 24, 56, 36, 28, 19, 45, 17, 60)	1.000
Test - 7	200	(25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 115, 120, 125, 130, 135, 140, 145, 150)	(119, 48, 38, 74, 41, 104, 41, 61, 43, 54, 66, 55, 43, 53, 45, 111, 33, 48, 43, 77, 55, 47, 42, 64, 16, 79)	1.500
Test - 8	200	(25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100)	(111, 62, 57, 88, 55, 69, 55, 78, 61, 68, 68, 79, 56, 55, 57, 61, 55, 60, 55, 69, 67, 61, 54, 74, 46, 99, 93, 58, 49, 84, 50, 73, 55, 70, 62, 78, 85, 64, 72, 62, 54, 65, 50, 57, 62, 86, 74, 56, 54, 63, 38, 87, 53, 56, 46, 82, 52, 56, 52, 71, 63, 74, 76, 65, 73, 63, 65, 83, 58, 63, 58, 92, 80, 62, 57, 79)	5.000
Test - 9	200	(25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144)	(138, 79, 75, 104, 82, 96, 102, 95, 76, 85, 95, 86, 73, 82, 74, 78, 72, 77, 72, 96, 84, 78, 71, 95, 75, 106, 107, 75, 66, 101, 87, 90, 67, 87, 69, 85, 92, 81, 69, 79, 68, 82, 59, 84, 69, 103, 83, 73, 74, 93, 55, 104, 70, 73, 63, 99, 69, 63, 69, 88, 70, 81, 93, 82, 70, 80, 72, 100, 65, 80, 75, 109, 87, 79, 74, 96, 49, 114, 109, 83, 73, 109, 80, 121, 76, 96, 78, 89, 101, 90, 78, 85, 77, 92, 75, 86, 70, 103, 81, 83, 78, 90, 57, 105, 100, 74, 74, 98, 67, 113, 68, 88, 80, 81, 83, 82, 80, 80, 72, 77)	10.000
Test- 10	200	(15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150)	(203, 144, 140, 169, 147, 161, 167, 160, 141, 139, 160, 151, 138, 147, 139, 143, 137, 142, 137, 161, 149, 143, 136, 160, 140, 171, 172, 140, 131, 166, 152, 155, 132, 152, 134, 150, 157, 146, 134, 144, 133, 147, 124, 149, 134, 168, 148, 138, 139, 158, 120, 169, 135, 128, 117, 164, 124, 123, 134, 153, 135, 146, 148, 147, 135, 145, 137, 165, 130, 145, 140, 174, 152, 144, 139, 161, 114, 179, 174, 148, 138, 174, 145, 186, 141, 161, 143, 154, 166, 155, 143, 150, 142, 157, 140, 151, 135, 168, 146, 148, 143, 155, 122, 170, 165, 139, 139, 163, 132, 178, 133, 153, 145, 146, 148, 145, 145, 145, 137, 142, 120, 149, 134, 168, 148, 127, 139, 158, 120, 169, 135, 138, 128, 164, 134, 128)	20.000

Tablo 1’den görüleceği gibi test verilerinin toplam talep miktarı, Test-1 için yüz adet talepten başlayıp Test-10 için

yirmi bin talebe kadar gitmektedir. Talep miktarı ve talep edilen çubuk parça çeşit sayısı artmasıyla problem daha

karmaşık hale gelmekte ve zorlaşmaktadır. Problemin amaç fonksiyonu, toplam talep miktarının karşılanması ve bu talep karşılanırken en az sayıda ana malzeme kullanılmasıdır. Ayrıca, iyi bir kesim planı oluşturup fire miktarının oluşmaması veya en küçük oranda kalması hedeflenmiştir.

4.3. Problemin Modellenmesi

Tavlama benzetimi ve geliştirilen sezgisel model algoritmaları stok kesme problemlerine uygulanmıştır.

4.3.1. Tavlama Benzetimi Algoritması

Tavlama benzetimi algoritması yerel minimuma takılmadan en iyi çözüme ulaşabilmesi amacıyla başlangıç parametreleri Tablo 2'de gösterildiği gibi belirlenmiş ve algoritma içerisine tanımlanmıştır.

Tablo 2. Tavlama benzetimi parametreleri

Başlangıç Sıcaklık	1.000
Tekrar Sayısı	10
Soğuma Oranı	0,99
Durma Kriteri	1.000

Başlangıç sıcaklık değeri yeterince yüksek bir değer olan bin değeri seçilirken, soğuma oranı (alfa) bir değerine yakın olan 0,99 değeri seçilmiştir. Her döngünün tekrar sayısı (iterasyon sayısı) on olarak girilmiştir ve durma kriteri olarak ise soğuma oranının bin kere uygulanacağı belirtilmiştir. Bu parametre değerleri istenirse farklı değerlerle değiştirilebilir.

Fire oranı hesaplanırken tüm çözüm metotlarına uygulanan hesaplama yöntemi şu şekilde olmuştur: fire oranı (%) = (toplam kesme kaybı uzunluğu / kullanılan toplam ana çubuk uzunluğu) * 100.

Bir boyutlu stok kesme problemlerine uygulanan tavlama benzetimi algoritmasında kullanılan indisler ve algoritmanın temel adımları aşağıda özetlenmiştir.

Algoritmada kullanılan indisler:

i = Talep edilen çubuk parça uzunluk sayısı, $i = 1 \dots L$

j = Ana çubuk sayısı, $j = 1 \dots N$

L = Talep edilen toplam çubuk parça uzunluk sayısı

N = Kullanılan toplam ana çubuk sayısı

T_0 = Mevcut sıcaklık

P = Boltzmann olasılık durumu

e = Euler sayısı

r = 0 ile 1 arasında rassal değer

l = Talep edilen çubuk parçasının uzunluğu, $l \in L$

d = Her talep edilen çubuk parçası uzunluğuna karşılık gelen talep sayısı, $d \in L$

u = Ana çubuğun uzunluğu

b = Başlangıç çözümün N değeri

m = Mevcut çözümün N değeri

k = Komşu çözümün N değeri

g = En iyi (global) çözümün N değeri

Algoritma temel adımları:

I. Başlangıç tanımlarının yapılması:

Adım 1. Tavlama benzetimi parametrelerini (başlangıç sıcaklık, tekrar sayısı, soğuma oranı, durma kriteri) tanımla.

Adım 2. Başlangıç çözüm listesi, komşu çözüm listesi, mevcut çözüm listesi, en iyi çözüm listesi, rassal değer, ana çubuk uzunluğunu tanımla.

Adım 3. Talep edilen çubuk parça uzunluğu (l) ve talep sayısı (d) listelerini tanımla.

II. Başlangıç çözüm oluşturma:

Adım 4. Mevcut sıcaklığı (T_0) = Başlangıç sıcaklık olarak ata.

Adım 5. (l_i)'leri ve (d_i)'leri l ve ilgili listelerin içine ekle.

Adım 6. (l_i)'leri en uzundan en kısaya doğru sırala ve bu listenin karşısında (d_i) listesini sırala.

Adım 7. Bir başlangıç çözüm oluşturmak için, sırayla (l_i)'leri, tam sayı(kalan çubuk boyu/ l_i) * l_i kadar ana çubuk (u_j)'lere ata.

Adım 8. Başlangıç çözümü, mevcut çözüm olarak ata ($m = b$).

Adım 9. Mevcut çözümü, en iyi (global) çözüm olarak ata ($g = m$).

III. Komşu çözüm oluşturma:

Adım 10. Komşu çözümü bulmak için, iki adet birbirinden farklı rassal tam sayı ($1 \leq \text{rassal sayı} \leq L$ koşuluna göre) üret.

Adım 11. Üretilen rassal sayılara göre, ilk rassal sayıya karşılık gelen birinci (l_i) ile ikinci rassal sayıya karşılık gelen ikinci (l_j)'yi aralarında yer değiştir.

Adım 12. Üretilen rassal sayılara göre, ilk rassal sayıya karşılık gelen birinci (d_i) ile ikinci rassal sayıya karşılık gelen ikinci (d_j)'yi aralarında yer değiştir.

Adım 13. Adım 7'deki prosedürü tekrarla.

Adım 14. Komşu çözüm, mevcut çözümden iyi mi?

Evet ise (eğer $k \leq m$ koşuluna göre) komşu çözümü kabul et, $m = k$ olarak ata ve Adım 17'ye git. Hayır ise Adım 15'e git.

Adım 15. Komşu çözümü, kabul koşulunu göre ($e^{-(k-m)/T_0} > r$) kabul et veya ($e^{-(k-m)/T_0} < r$) ret et.

Adım 16. Sonuç; kabul ise $m = k$ olarak ata ve Adım 18'e git, ret ise Adım 18'e git.

Adım 17. Mevcut çözüm, şimdiye kadar elde ettiğimiz en optimal sonuç mu? Evet ise $g = m$ olarak ata ve Adım 18'e git. Hayır ise Adım 18'e git.

Adım 18. Tekrar sayısına ulaşıldı mı? Evet ise Adım 19'a git. Hayır ise tekrar sayısını bir arttır ve Adım 10'a git.

Adım 19. Sıcaklık soğuma oranına göre sıcaklık azalt, yeni $(T_0) = \text{alfa} * (T_0)$ olarak ata.

Adım 20. Durma kriteri koşuluna ulaşıldı mı? Evet ise Adım 21'e git. Hayır ise Adım 10'a git.

IV. Sonucun yazdırılması ve algoritmanın durması:

Adım 21. En iyi çözüm olan (g) değerini, fire hesabını ve algoritma çalışma süresini ekrana yaz.

Adım 22. Algoritma dur.

4.3.2. Geliştirilen Sezgisel Algoritma

Bir boyutlu stok kesme problemlerini çözmek için ilk uygun azalan sezgiselinden yola çıkılarak bir sezgisel algoritma geliştirilmiştir. İlk uygun azalan sezgiselinde, talep edilen çubuk parçaları uzunluklarına göre azalan sırada sıralanır ve ana malzemeye atama işlemine en uzun parçadan başlanır. Atama işlemine sırası gelen parçadan devam edilir ve tüm parçaların yerleştirme işlemi bitene kadar atama devam eder. Geliştirilen sezgisel algoritmada, ilk uygun azalan sezgiselinde olduğu gibi çubuk parçaları azalan sırada sıralanır ve ana malzemeye atama işlemine en uzun parçadan başlanır. Ana malzemeye sırası gelen parçanın mümkün olan ataması yapıldığında, ana malzemeden geriye kalan uzunluk var mı kontrol edilir. Eğer ana malzemeden geriye kalan uzunluk varsa, sırayla henüz ataması yapılmamış çubuk parçaları kontrol edilir ve kalan uzunluğa yerleşebilecek çubuk parçası varsa ataması yapılır. Böylece ana malzeme mümkün olduğunca fazla atama yapılarak en fazla yerleşimin yapılması amaçlanır. Talep edilen çubuk parçalarının ataması bitene kadar yerleştirme işlemleri devam eder.

Geliştirilen sezgisel algoritmanın temel adımları aşağıda özetlenmiştir.

Algoritmanın temel adımları:

Adım 1. Başla.

Adım 2. Talep edilen çubuk parçalarını büyükten küçüğe sırala.

Adım 3. En uzun çubuk parçasından başlayarak, çubuk parçasını ana malzemeye ata.

Adım 4. Ana malzemede kullanılmayan uzunluk varsa uygun çubuk parçasından ata.

Adım 5. Talep edilen miktarlar karşılana kadar döngüyü tekrar et.

Adım 6. Kullanılan toplam ana çubuk sayısını, fire hesabını ve algoritma çalışma süresini ekrana yaz.

Adım 7. Bitiş.

5. Bulgular ve Tartışma

Hesaplama denemeleri, 64-bit Windows işletim sistemine sahip bir bilgisayar kullanılarak gerçekleştirilmiş olup işlemci olarak AMD Ryzen 5 4600H Radeon Graphics 3.00 GHz ve 8 GB RAM kullanılmıştır. Tavlama benzetimi ve geliştirilen sezgisel model algoritmaları Python [23] programlama dili ile kodlanıp çözümlerken, tam sayılı matematiksel model Gurobi Python API 9.5 [24] sürümü ile çözülmüştür. Dördüncü yöntem olarak, literatürde sıkça kullanılan ve stok kesme problemlerinde iyi sonuçlar veren hazır paket programlarından biri olan Real Cut 1D (sürüm 11.7.3.1) [25] kullanılmıştır. Hesaplama sonuçları, dört çözüm yolu içinde Tablo 3'te gösterilmiştir. Tablo 3'ün ilk sütununda test verilerinin adı, diğer sütunlarda sırasıyla kullanılan yöntemler belirtilmiştir. Her bir yöntemin altında sonuç (kullanılan toplam ana çubuk sayısı), fire oranı ve süre gösterilmiştir.

Tablo 3. Hesaplama sonuçları

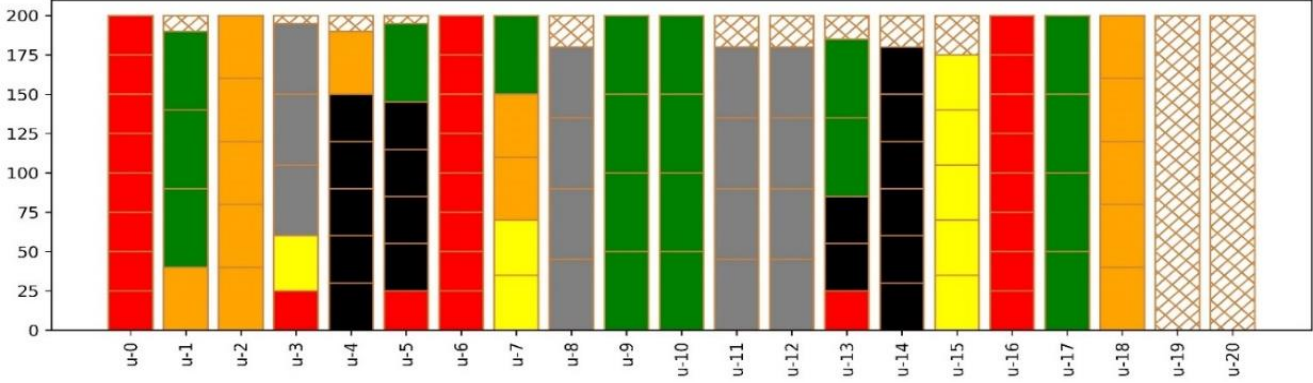
Test Verileri	Tavlama Benzetimi			Geliştirilen Sezgisel Yöntem			Tam Sayılı Matematiksel Algoritma			Real Cut 1D		
	Sonuç (adet)	Fire (%)	Süre (sn.)	Sonuç (adet)	Fire (%)	Süre (sn.)	Sonuç (adet)	Fire (%)	Süre (sn.)	Sonuç (adet)	Fire (%)	Süre (sn.)
Test - 1	19	3,95	0,75	19	3,95	0,02	19	3,95	0,03	19	3,95	0,17
Test - 2	41	4,76	1,54	41	4,88	0,02	39	0,00	0,25	39	0,00	0,63
Test - 3	91	2,58	4,63	91	2,58	0,03	89	0,39	12,53	89	0,39	0,66
Test - 4	207	0,50	17,03	208	1,00	0,03	206	0,04	32,49	206	0,04	0,47
Test - 5	305	0,68	24,12	306	1,01	0,03	303	0,03	45,81	303	0,03	0,48
Test - 6	419	0,36	33,65	421	0,84	0,03	418	0,13	24,64	418	0,13	0,18
Test - 7	632	0,23	51,61	634	0,55	0,03	631	0,08	335,62	631	0,08	0,36
Test - 8	1.604	2,73	233,02	1.605	2,79	0,04	1.571	0,68	1800,31	1.562	0,11	0,70
Test - 9	4.237	0,52	980,17	4.240	0,60	0,13	4225	0,25	1801,22	4.216	0,03	2,95
Test - 10	8.227	0,20	1.967,86	8.229	0,23	0,19	8229	0,23	1800,69	8.211	0,01	4,11

Tablo 3'ten görüleceği gibi problem veri setlerinin çözümünden kullanılan dört farklı metod içinde sarf edilen ana çubuk ve fire oranları açısından yüksek verimliliğe sahiptir. Çünkü tüm test verilerinin çözümünden ortaya çıkan sonuca göre, ana malzeme sarfı düşük seviyede ve

ideal seviyeye yakın olmuştur. Tüm hesaplama sonuçlarına göre en yüksek fire oranı %4,88 olarak gerçekleşmiştir. Bu fire oranı bir işletmenin operasyonel süreçleri için kabul edilebilir düzey olup etkin bir verimlilik göstergesi olarak değerlendirilebilir. Test verilerinin çözümleri tek tek

incelenirse, Test-1'in çözüm neticesi dört yöntem içinde sarf edilen ana çubuk sayısı ve bunun sonucunda fire oranları aynı seviyede olmuştur. Gurobi programında Test-1'in çözüm sonucu görsel olarak Şekil.2'deki gibi gerçekleşmiş ve şekilde toplam 19 adet ana çubuğa atama yapıldığı görülmektedir. Şekil 2 çözüm grafiğinde, her bir

dikey sütun ana çubuğu göstermekte ve bu sütunların renkli bölümleri ana çubuğun kullanılan kısımlarını simgelemektedir. Ana çubuğun renksiz kısımlar ise çubuğun kullanılmayan veya kayıp uzunluklarını ifade etmektedir.



Şekil 2. Tam sayılı matematiksel programlamanın test-1 çözüm grafiği

Test-2 veri setinin çözümü için tavlama benzetimi ve geliştirilen sezgisel algoritma, kullanılan ana çubuk sayısını 41 adet, fire yüzdelere ise sırayla 4,76 ve 4,88 olarak hesaplamıştır. Kullanılan ana çubuk sayısının 39, toplam fire miktarının sıfır olduğu çözümün mümkün olmasına rağmen metasezgisel ve sezgisel yöntemlerin 41 adet ana malzeme kullanımı ve %4,76, %4,88 fire oranlarıyla hesaplama yapmaları bu algoritmaların çalışma temelinde en iyi veya en iyiye yakın çözüm üretebilmesinden kaynaklanmıştır. Yine de, tavlama benzetimi ve önerilen sezgisel algoritmanın mümkün olan en iyi çözümden 2 adet fazla ana çubuk kullanmaları ve %4,76, %4,88 fire oranlarıyla hesaplama yapmış olmaları iyi seviyede çözüm olarak kabul edilebilir.

Test-2 için, Real Cut 1D ve tam sayılı matematiksel algoritma yüzde sıfır fireyle çözüm sağlamıştır. Real Cut 1D programının Test-2 veri setinden elde edilen çözüm raporu Tablo 4.'te gösterilmiştir.

Tablo 4. Real Cut 1D test-2 çözümü

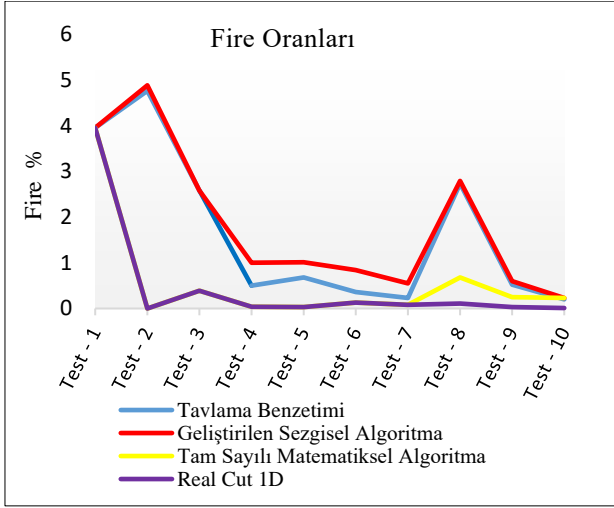
#	Uzunluk (cm)	Miktar (adet)	Atık (cm)	Çözüm
1	200	4	0	4 x 50 Atık = 0
2	200	6	0	2 x 55, 2 x 45, Atık = 0
3	200	1	0	1 x 60, 1 x 50, 2 x 45, Atık = 0
4	200	9	0	2 x 60, 2 x 40, Atık = 0
5	200	1	0	5 x 40, Atık = 0
6	200	1	0	1 x 45, 3 x 40, 1 x 35, Atık = 0
7	200	1	0	2 x 45, 1 x 40, 2 x 35, Atık = 0
8	200	4	0	3 x 45, 1 x 35, 1 x 30, Atık = 0
9	200	2	0	4 x 35, 2 x 30, Atık = 0
10	200	5	0	5 x 30, 2 x 25, Atık = 0
11	200	1	0	1 x 35, 3 x 30, 3 x 25, Atık = 0
12	200	1	0	2 x 35, 1 x 30, 4 x 25, Atık = 0
13	200	3	0	8 x 25, Atık = 0

Sarf edilen ana malzeme ve fire oranları bakımından Test-3, Test-4, Test-5, Test-6 ve Test-7'nin çözümleri Real Cut 1D ve tam sayılı matematiksel programlama aynı değerleri vermiş ve bu yöntemlerin hesaplama sonuçları tavlama benzetimi ve geliştirilen sezgisel algoritmanın sonuçlarından daha iyi olduğu görülmüştür.

Tavlama benzetimi algoritmasının Test-4'ten Test-10'a kadar olan çözümleri, toplam ana malzeme kullanımı açısından geliştirilen sezgisel algoritmadan daha iyi ve en iyi çözümlere yakın değerlerdedir. Stok kesme problemlerinin çözümünde tavlama benzetimi algoritmasının başarılı sonuçlar vermesi ve en iyi veya en iyiye yakın çözümler üretmesi, tavlama benzetimi algoritmasının stok kesme problemleri gibi karmaşık problemlerin çözümünde etkili bir araç olarak kullanılabileceğine işaret etmektedir.

Real Cut 1D ile çözülen Test-8, Test-9 ve Test-10'un, sarf edilen ana malzeme sayısı ve fire oranı açısından en iyi çözümleri üretirken, tam sayılı matematiksel algoritma ise ikinci en iyi çözümleri üretmiştir. Test-8, Test-9 ve Test-10'un çözümünde, tam sayılı matematiksel algoritmanın çözüm süreleri daha fazla uzayacağından program durdurulmuş ve geçen süre zarfında elde edilen sonuçlar Tablo 3'te gösterilmiştir. En iyi çözümden uzaklığını ifade eden (MIPGap %) değeri Test-8, Test-9 ve Test-10 için sırayla 0,6365, 0,2367 ve 0,2187 olarak gerçekleşmiştir. Tam sayılı matematiksel algoritma (Gurobi programı) tam çözüme ulaşabilmesi için yeterli süre beklenirse daha iyi sonuçlara erişmek mümkün olabilir.

Tüm problemlerin, fire yüzdesi bakımından sonuçlarını gösteren grafik Şekil 3'te verilmiştir.



Şekil 3. Problemlerin fire oranı grafiği

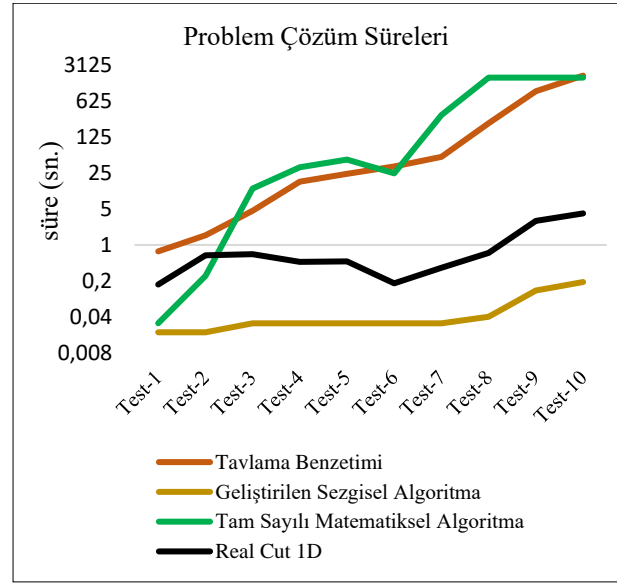
Test-1'den Test-10'a doğru gittikçe test veri büyüklüklerinin artmasına bağlı olarak dört yöntemin de hesaplama sürelerinde genelde artış görülmüştür. Bu durum, talep edilen çubuk parça çeşitliliğinin ve talep miktarlarının artmasının bir sonucu olarak problemin zorluğunun, başka bir ifadeyle karmaşıklığın artmasından kaynaklanmıştır. Çünkü çubuk parça çeşitliliğinin ve talep sayısının artması sonucunda, çubuk parçalarının ana çubuğa atanması açısından daha fazla alternatif yaratmıştır. Alternatiflerin artması problemleri zorlaştırmış ve problemlerin zorlaşması özellikle tam sayılı matematiksel algoritmanın ve tavlama benzetimi algoritmasının çözüm sürelerinde ciddi artışa neden olmuştur. Daha büyük veri ve çok amaçlı problemlerle çalışılırsa tavlama benzetimi algoritmasının ve tam sayılı matematiksel algoritmanın çözüm süreleri açısından performansları azalabilir ve bunun sonucunda çözüm süreleri daha fazla uzayabilir.

Tablo 5. Çözüm Sonuçları

Deney Verileri	Geliştirilen Sezgisel Yöntem		Sezgisel Yöntem (P)		Sezgisel Yöntem (P) ve Matematiksel Yöntem (P1) Birlikte Kullanımı	
	Sonuç (adet)	Süre (sn.)	Sonuç (adet)	Süre (sn.)	Sonuç (adet)	Süre (sn.)
Experiment - 1	22	0,01	21	<1	21	1
Experiment - 2	45	0,02	45	<1	45	1
Experiment - 3	72	0,02	72	2	71	50000.44

Tablo 5'te görüleceği gibi sezgisel yöntem (P) ve matematiksel yöntem (P1) birlikte kullanımı, ana malzeme sarfi açısından üç deney içinde en iyi sonuçlara ulaşmıştır. Matematiksel yöntemin Experiment-3'ün çözüm süresi incelendiğinde, problemin zorluğunun artmasının da etkisiyle diğer yöntemlere göre bariz şekilde uzun sürdüğü görülmektedir. Matematiksel yöntemin hesaplama süresinin sezgisel yöntemlere göre daha uzun sürmesi beklenen bir sonuç olmuştur.

Problemlerinin çözümünden kullanılan yöntemlerin çözüm süreleri görsel olarak Şekil 4'teki grafikte gösterilmiştir.



Şekil 4. Problemlerin çözüm süre grafiği

Geliştirilen sezgisel algoritmanın performansını karşılaştırmak amacıyla literatürde yer alan Kasımbeyli ve diğerlerinin [26] çalışmalarında yer alan ilk üç deney (Experiment 1, 2 ve 3) geliştirilen sezgisel algoritma ile çözülmüş ve çözümler Tablo 5'te gösterilmiştir. Tablo 5'in ilk sütununda deney verilerinin adı, diğer sütunlarda kullanılan yöntemler belirtilmiştir. Yöntemler sırayla; bu çalışmada geliştirilen sezgisel yöntem, Kasımbeyli ve diğerleri tarafından çalışmalarında önerdikleri sezgisel yöntem (P) ve sezgisel yöntemin matematiksel yöntem (P1) ile birlikte kullanımı gösterilmiştir.

6. Sonuçlar

Bu çalışmada, bir boyutlu stok kesme problemlerinin çözümü için farklı büyüklükteki test veri setleri hazırlanıp geliştirilen sezgisel algoritma, tavlama benzetimi algoritması, tam sayılı matematiksel programlama ve literatür çalışmalarında sıklıkla kullanılan Real Cut 1D paket programı aracılığıyla çözülmüştür. Ele alınan çözüm yöntemlerinin hesaplama bulguları karşılaştırmalı olarak değerlendirilmiştir. Yapılan karşılaştırmalar neticesinde, tüm yöntemlerin sarf edilen ana malzeme ve fire yüzdesi bakımından etkin metotlar oldukları görülmüştür. Ayrıca, literatürde yer alan bir çalışmadan alınan deney verileri geliştirilen sezgisel algoritma ile çözülmüş ve iyi sonuçlar elde edilmiştir.

Geliştirilen sezgisel algoritma, diğer yöntemler içerisinde hesaplama süresi açısından en iyi performansa sahip olmuştur. Test veri büyüklüğünün artması sezgisel algoritmanın çözüm süresini bariz şekilde etkilememiştir. Ancak veri büyüklüğünün artması ve bunun doğal sonucu olarak problemin karmaşıklığının artmasına bağlı olarak tavlama benzetimi algoritmasının ve tam sayılı matematiksel programlamanın çözüm süreleri uzamıştır. Daha büyük veriyle veya çok amaçlı problemler ile çalışılması durumunda, tavlama benzetimi algoritmasının ve matematiksel programlamanın çözüm sürelerinin daha fazla uzamasına neden olabilir ve bu durum yöntemleri pratik olmaktan uzaklaştırabilir.

Geliştirilen sezgisel algoritmanın sade, anlaşılır ve birkaç adımdan oluşan algoritma yapısı sayesinde, kolayca kodlaması yapılarak kullanımı sağlanabilir. Çubuk parçası, talep miktarı, ana malzeme uzunluğu gibi verilerin girişi son kullanıcı tarafından rahatlıkla yapılabilir. Hazır paket programlarında genellikle kullanıcı bazlı lisans maliyetlerinin olması işletmelere ayrıca bir maliyet kalemi olarak yansımaktadır. Kullanılacak etkin bir sezgisel yöntem hem kesme planlarının daha etkin ve verimli şekilde yapılmasına olanak sağlayabilir hem de lisans maliyetleri gibi ek maliyetlerin önüne geçilebilir.

Çıkar Çatışması Beyanı:

Yazarlar, bu makalede bildirilen çalışmayı etkilemiş gibi görünebilecek, bilinen rakip mali çıkarları veya kişisel ilişkileri olmadığını beyan ederler.

Etik Standartlar Beyanı:

Bu çalışmada kullanılan materyal ve yöntemler etik kurul izni ve yasal-özel izin gerektirmemektedir.

Kaynaklar

- [1] Poldi, K. C., de Araujo, S. A., 2016. Mathematical models and a heuristic method for the multiperiod one-dimensional cutting stock problem. *Annals of Operations Research*, 238(1), 497-520.
- [2] Onursal, F. S., Fıglalı, A., & Kayman, Y., 2015. A Model For Optimizing Material Assortment. *Eurasian Business & Economics Journal*, 2(2), 76-92.
- [3] Söke, A., Bingül, Z., 2005. İki Boyutlu Giyotinsiz Kesme Problemlerinin Benzetilmiş Tavlama Algoritması ile Çözümlerinin İncelenmesi. *Politeknik Dergisi*, 8(1), 25-36.
- [4] Fırat, H., Alpaslan, N., 2019. Sezgisel Algoritmalar Kullanılarak İki Boyutlu Dikdörtgen Şerit Paketleme Probleminin Çözümü. *Avrupa Bilim ve Teknoloji Dergisi*, 17, 315-322.
- [5] Pierini, L. M., Poldi, K. C., 2021. An analysis of the integrated lot-sizing and cutting-stock problem formulation. *Applied Mathematical Modelling*, 99, 155-165.
- [6] Sanchez, I. A. L., Vargas, J. M., Santos, C. A., Mendoza, M. G., & Moctezuma, C. J. M., 2018. Solving binary cutting stock with metaheuristics using particle swarm optimization and simulated annealing. *Soft Computing*, 22(18), 6111-6119.
- [7] Baykasog, A., Özbel K., B., Baykasog, A., Özbel K., B., 2021. Modeling and solving a real-world cutting stock problem in the marble industry via mathematical programming and stochastic diffusion search approaches. *Computers and Operations Research*, 28, 105-173.
- [8] Saraç, T., Sağır, M., 2021. Mixed-integer programming models for 1.5-dimensional cutting problem with technical constraints. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 36(1), 291-302.
- [9] Velasco, A. S., Uchoa, E., 2019. Improved state space relaxation for constrained two-dimensional guillotine cutting problems. *European Journal of Operational Research*, 272, 106-120.
- [10] Tung, K. T., Chen, C. Y., Hung, Y. F., 2016. Solving cutting scheduling problem by simulated annealing search method. In 2016 IEEE International Conference on Industrial Engineering and Engineering Management, Bali, Indonesia, 04-07 December, 907-911.

- [11] Kasimbeyli, N., Demirci, D., 2018. Bir Boyutlu Kesme Problemi İçin Üç Amaçlı Bir Matematiksel Model ve Çözüm Algoritması. *Endüstri Mühendisliği*, 29(3-4), 42-50.
- [12] Mutur, İ., Sezer, Z., 2018. Algorithms for the one-dimensional two-stage cutting stock problem. *European Journal of Operational Research*, 271, 20–32.
- [13] Tanır, D., Uğurlu, O., Nuriyev, U., Kapar, M., 2018. Birleşik Stok Kesme ve Patern Sıralama Problemi için Bir Sezgisel Algoritma. *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 22(1), 300-305.
- [14] Vishwakarma, R., Powar, P. L., 2021. An efficient mathematical model for solving one-dimensional cutting stock problem using sustainable trim. *Advances in Industrial and Manufacturing Engineering*, 3, 100046.
- [15] Parreño, F., Alvarez-Valdes, R., 2021. Mathematical models for a cutting problem in the glass manufacturing industry. *Omega* 103, 102432.
- [16] Wuttke, A. D., Heese, S. H., 2018. Two-dimensional cutting stock problem with sequence dependent setup times. *European Journal of Operational Research*, 265, 303–315.
- [17] Şık, E., Tosun, S., 2021. Benzetilmiş Tavlama Algoritması İle Fazör Ölçüm Birimlerinin Optimal Yerleşimi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 9(2), 746-758.
- [18] Temiz, İ., 2010. Çok Kriterli Permütasyon Akış Tipi Çizelgeleme Problemi için Bir Tavlama Benzetimi Yaklaşımı. *Cankaya University Journal of Science and Engineering*, 7(2), 141-153.
- [19] Ayan, T. Y., 2009. Kaynak Kısıtlı Çoklu Proje Programlama Problemi İçin Tavlama Benzetimi Algoritması. *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi*, 23(2), 101-118.
- [20] Rahimian M., Soltani R., Tahmasbi-birgani A., 2012. Multi-criteria sequence-dependent single machine scheduling using genetic and simulated annealing algorithms. *International Journal of Emerging Technology and Advanced Engineering*, 2, 532-537.
- [21] Uysal, M., Özcan, U., 2019. Süpermarket Yerleşim Problemi İçin Tavlama Benzetimi Algoritması Yaklaşımı. *Karadeniz Fen Bilimleri Dergisi*, 9(1), 58-69.
- [22] Burke, E. K., Kendall, G. ve Whitewell, G., 2004. A New Placement Heuristic for the Orthogonal Stock-Cutting Problem. *Operations Research*, 52(4), 655-671.
- [23] Python Programlama Dili. <https://www.python.org/>. [Erişim tarihi: 30 Kasım 2023]
- [24] Gurobi Yazılım Programı. <https://www.gurobi.com>. [Erişim tarihi: 30 Kasım 2023]
- [25] Real Cut 1D Yazılım Programı. <http://www.optimalprograms.com/realcut1d.htm>. [Erişim tarihi: 30 Kasım 2023]
- [26] Kasimbeyli, N., Sarac, T., Kasimbeyli, R., 2011. A two-objective mathematical model without cutting patterns for one-dimensional assortment problems. *Journal of Computational and Applied Mathematics*, 235(16), 4663-4674.