



Stokastik Teslim Zamanlı Tek Makine Çizelgeleme Problemi İçin Bir Harmoni Arama Algoritması

A Harmony Search Algorithm for The Single Machine Scheduling Problem with Stochastic Due Dates

Tuğba Saraç ^{*}, Feriştah Özçelik 

Eskişehir Osmangazi Üniversitesi Mühendislik Mimarlık Fakültesi Endüstri Mühendisliği Bölümü, Eskişehir, TÜRKİYE

Sorumlu Yazar / Corresponding Author *: tsarac@ogu.edu.tr

Öz

Bir iş, teslim zamanından sonra tamamlandıysa o iş gecikir. İşlerin gecikmesinin, müşteri memnuniyetsizliği ve müşterilere ödenecek cezalar gibi ağır bedelleri olabileceğinden önlenmesi önemlidir. Eğer teslim zamanlarında bir belirsizlik var ise bir başka değişle müşterilerin ürünleri talep ettikleri tarihi öne çekmeleri ya da ertelemeleri mümkün ise bu durumda işler çizelgelenirken mutlaka bu belirsizliğin göz önünde bulundurulması gerekir. Bu nedenle bu çalışmada, stokastik teslim zamanlı tek makine çizelgeleme problemi ele alınmıştır. Ele alınan problemin çözümü için iki aşamalı bir stokastik programlama modeli önerilmiştir. Ancak problemin NP-zor doğası, büyük boyutlu problemlerin kesin çözüm yaklaşımları ile çözülebilmesine engeldir. Bu nedenle büyük boyutlu problemlerin çözülebilmesi için de bir harmoni arama algoritması önerilmiştir. Önerilen çözüm yaklaşımlarının performansları rassal türetilen test problemleri kullanılarak gösterilmiştir. Elde edilen test sonuçları, teslim zamanlarının stokastik doğasının dikkate alınmasının toplam gecikmeyi %10'a kadar azaltabileceğini ortaya koymuştur.

Anahtar Kelimeler: Tek makine çizelgeleme problemi, Stokastik teslim zamanı, Harmoni arama algoritması, Stokastik programlama

Abstract

If a job is completed after its due date, that job is tardy. It is important to avoid tardiness as it can have heavy costs such as customer dissatisfaction and customer penalties. If there is any uncertainty in the due dates, in other words, if it is possible to change due dates as earlier or later, then this uncertainty must be taken into account when scheduling the jobs. Therefore, in this study, the single machine scheduling problem with stochastic due dates is considered. A two-stage stochastic programming method is proposed for the solution of the considered problem. However, the NP-hard nature of the problem prevents finding a feasible solution for large-scale problems with exact solution approaches. Therefore, a harmony search algorithm is proposed for solving large-size problems. The performances of the proposed solution approaches are demonstrated using randomly generated test problems. The test results revealed that taking into account the stochastic nature of due dates can reduce the total tardiness by up to 10%.

Keywords: Single machine scheduling problem, Stochastic due dates, Harmony search algorithm, Stochastic programming

EXTENDED ABSTRACT

Introduction

The single machine scheduling problem is an important problem because it is a sub-problem of more complex scheduling problems such as parallel machine scheduling and job shop scheduling. The parameters of machine scheduling problems generally have a stochastic structure. Therefore, there are studies in the literature that have addressed the problem stochastically. These studies are summarized in Table 1.

Variability in due dates occurs when customers bring forward or postpone the date they request products. When these changes are not taken into account, some products may be delayed and some may be produced ahead of time. Since production plan changes can occur at a significant frequency, such as several times a month, in some sectors, the uncertainty of due dates must definitely be taken into account. Otherwise, companies may experience major losses. Considering stochastic due dates is a

more realistic approach and makes it possible to reach more successful solutions. Therefore, the due dates of the jobs are considered randomly in this study.

When the studies in the literature that consider stochastic due dates are evaluated, the sequencing rules that can guarantee the optimum solution under some special conditions (such as having a certain distribution or objective function) come to the fore. However, due to their nature, sequencing rules can only produce successful solutions for problems for which they are specifically developed, and they cannot be expected to produce successful solutions when changes are made to the structure of the problem. Therefore, more general solution approaches are needed to solve the stochastic single-machine scheduling problem. In this study, a two-stage stochastic programming approach is proposed. A harmony search algorithm is also proposed for the solution of large-scale problems. Considering the accessible literature, a metaheuristic algorithm is proposed for the first time in this

study for the single-machine scheduling problem where due dates are stochastic (STMCP) and total tardiness is minimized.

Materials and Methods

In STMCP, n jobs are processed on a machine. The processing time of job j (p_j) is known. The setup times of jobs (s_{ij}) are sequence dependent. Jobs are delayed if they are completed after their due dates. Due dates are stochastic. Jobs cannot be splitted and their ready at time zero. Only one job can be processed on a machine at a time. The objective function is to minimize the total tardiness. The developed deterministic and stochastic mathematical models are given in Section 2.

For the solution of large sized problems, a harmony search algorithm is proposed. Harmony search (HA) is a music-inspired population based metaheuristic algorithm. The details of the proposed HA algorithm is given in Section 3.2.

Results and Discussion

The test problems were solved using the DTMC and STMC models with a time limit of 7200 seconds. The results obtained with GAMS/CPLEX are given in Table 2. As can be seen from table, the optimum solutions of only 10-job problems could be obtained with the DTMC and STMC models. No feasible solution could be found for both the s1 and s2 variable type stochastic problems of 150-1-1 within the time limit with the STMC model and GAMS/CPLEX. No feasible solution could be found for the deterministic, s1 and s2 variable type stochastic problems of 200-1-1 within the time limit.

The results of the developed harmony search algorithm are given in Table 3. As can be seen from Table 3, the proposed HA algorithm was able to reach the optimum solutions for all 10-job

problems. HA algorithm could not reach better solutions than GAMS/CPLEX only for 50-1-2 and 50-2-1 deterministic problems. For all the remaining problems, more successful results. With the proposed HA algorithm, the results of deterministic problems were improved by an average of 7%, the results of stochastic problems with s1 variable type by an average of 12.3% and the results of stochastic problems with s2 variable type by an average of 13.7%. When the solution times are compared, the longest solution time of HA algorithm is 118 seconds, while GAMS/CPLEX used the time limit of 7200 seconds for all test problems with dimensions of 20 and above. As a result, the HA algorithm was able to achieve better results in a shorter time than GAMS/CPLEX.

Conclusion

Since job delays directly affect customer satisfaction, many businesses want to prevent delays if possible, or reduce them as much as possible. Therefore, minimizing total tardiness is one of the most important objective functions considered in machine scheduling problems. If a job completed after its due date, tardiness occurs. In general, due dates are in stochastic structure. In this study, the effect of stochastic handling of due dates on total tardiness is investigated. The considered problem is a single machine scheduling problem with stochastic due dates. A two-stage stochastic programming method and a harmony search algorithm are proposed to solve this problem. The performances of the proposed solution approaches are demonstrated using randomly generated test problems. The obtained test results show that stochastic handling of due dates can reduce total tardiness by up to 10%. In the future, the effect of stochastic due dates on total tardiness can be studied for different machine environments.

1. Giriş

Tek makine çizelgeleme problemi, paralel makine çizelgeleme, atölye tipi çizelgeleme gibi daha karmaşık çizelgeleme problemlerinin alt problemi olması ve pek çok işletmede de uygulama alanı bulması nedeniyle önemli ve güncel bir problemidir. Makine çizelgeleme problemlerinin parametreleri genellikle stokastik bir yapıya sahiptir. Bu nedenle literatürde problemi stokastik olarak ele almış çalışmalar mevcuttur. Jang ve Klein [1], Seo vd. [2], Van den Akker ve Hoogeveen [3], Elyasi ve Salmasi [4], Elyasi ve Salmasi [5], Soroush [6], Soroush [7], Salmasnia vd. [8], Chang vd. [9], Atakan vd. [10], Lemos ve Ronconi [11], Iranpoor vd. [12], Baker [13], Qi vd. [14], Qi vd. [15], Wang vd. [16], Lu vd. [17], Zhang vd. [18], Cai vd. [19], Jia [20], Benmansour vd. [21], Cai ve Zhou [22], Cai ve Zhou [23], Cai ve Zhou [24], Cai vd. [25] işlem sürelerini, Zhang vd. [26] hazır olma zamanlarını, Ronconi ve Powell [27], Xu vd. [28], Zhang vd. [18] geliş zamanını, Ertem vd. [29], Ozcelik vd. [30], Lu vd. [17], Roohnavazfar [31] hazırlık sürelerini, Gu vd. [32], Qi vd. [15], Wang vd. [16], Cai ve Zhou [23], Cai ve Zhou [24] arızaları ve Jia [20], Benmansour vd. [21], Cai ve Zhou [22], Cai ve Zhou [23], Cai ve Zhou [24], Cai vd. [19], Cai vd. [25], Yue vd. [33], Wei [34] teslim zamanlarını stokastik ele almışlardır. Bu çalışmalar Tablo 1'de özetlenmiştir.

Teslim zamanlarındaki değişkenlik, müşterilerin ürünleri talep ettikleri tarihi öne çekmeleri ya da ertelemeleri ile ortaya çıkmaktadır. Bu değişiklikler dikkate alınmadığında, bazı ürünler gecikilmekte bazıları ise zamanından önce üretilebilmektedir. Gecikmelerin önlenmesi, işlerin gecikmesi nedeniyle müşterinin yaşayacağı memnuniyetsizliğin önlenmesi anlamına gelmektedir. Gecikme, bir ürün teslim zamanından sonra tamamlandıysa

ortaya çıkar. Eğer bir müşteri teslim zamanını öne çekerse, daha önce gecikme yaşanmayan bir iş gecikebilir. Bazı sektörlerde üretim planı değişiklikleri ayda birkaç kez gibi önemli bir sıklıkta gerçekleşebildiğinden teslim zamanlarının belirsizliği kesinlikle göz önünde bulundurulmalıdır. Aksi halde firmalar büyük kayıplar yaşayabilirler. Teslim zamanlarının deterministik değil, stokastik ele alınması daha gerçekçi bir yaklaşımdır ve daha başarılı çözümlere ulaşılabilmesini mümkün kılar. Bu nedenle bu çalışmada işlerin teslim zamanları stokastik olarak ele alınmıştır.

Tablo 1'de sunulan teslim zamanlarını stokastik olarak ele alan çalışmalar incelendiğinde; Cai vd. [19] stokastik tek makine çizelgeleme probleminde en büyük gecikme veya enbüyük ağırlıklı gecikmeyi enküçükleyecek optimal sıraları belirlemişlerdir. İlgili çalışmada işlem süreleri, teslim zamanları ve ağırlıklar stokastik olarak ele alınmıştır. Jia [20] hem işlem sürelerinin hem de teslim zamanlarının stokastik olduğu tek makine çizelgeleme problemini ele almıştır. Amaç, tamamlanma zamanlarının teslim zamanlarından sapmalarının beklenen ağırlıklı toplamını enküçüktür. Benmansour vd. [21] işlem sürelerini ve teslim zamanlarını stokastik olarak ele aldıkları çalışmalarında, tamamlanma zamanlarının teslim zamanlarından beklenen toplam ağırlıklı sapmasını enküçükleyecek şekilde optimal çizelgeleri araştırmışlardır. Cai ve Zhou [22] işlem sürelerinin ve teslim zamanlarının stokastik olduğu stokastik tek makine çizelgeleme probleminde, beklenen toplam maliyeti enküçüklenmeyi amaçlamışlardır. Problemin çözümü için bazı özel koşullar altında optimal çizelgeleri elde edebilen bir sıralama kuralı önermişlerdir.

Cai ve Zhou [23], çalışmalarında gecikme ve erken tamamlanma kaynaklı maliyetlerin beklenen ağırlıklı toplamını enküçüklemişlerdir. Teslim zamanlarının ve işlem sürelerinin

stokastik olduğu, ayrıca makinaların arızalarla karşılaşma olasılığının olduğu problemin çözümü için dinamik programlama algoritması önermişlerdir. Cai ve Zhou [24], teslim zamanlarının ve işlem sürelerinin stokastik olduğu, ayrıca makinaların arızalarla karşılaşma olasılığının olduğu stokastik tek makine çizelgeleme probleminde, gecikme ve erken tamamlanma kaynaklı maliyetlerin beklenen ağırlıklı toplamını enküçükleyecek bir algoritma önermişlerdir. Cai vd. [25], genel stokastik maliyet fonksiyonlarının olduğu stokastik tek makine çizelgeleme probleminde beklenen toplam maliyeti enküçükleyecek optimal çizelgeleri araştırmışlardır.

Tablo 1. Stokastik tek makine çizelgeleme literatürü.

Table 1. Stochastic single machine scheduling literature.

Kaynak	İS	HOZ	GZ	HZ	AR	TZ	G	SÇY	KÇY	MS
Jang ve Klein [1]	✓							✓		
Seo vd. [2]	✓								✓	
Van den Akker ve Hoogeveen [3]	✓								✓	
Elyasi ve Salmasi [4]	✓								✓	
Elyasi ve Salmasi [5]	✓						✓		✓	
Sorouh [6]	✓							✓		
Sorouh [7]	✓						✓		✓	
Salmasnia vd. [8]	✓						✓		✓	
Chang vd. [9]	✓								✓	
Atakan vd. [10]	✓						✓		✓	
Lemos ve Ronconi, [11]	✓						✓	✓		
Iranpoor vd., [12]	✓						✓	✓	✓	
Baker, [13]	✓						✓	✓	✓	
Qi vd., [14]	✓						✓		✓	
Zhang vd., [26]	✓							✓		
Gu vd. [32]					✓			✓		
Ronconi ve Powell, [27]			✓				✓		✓	
Xu vd., [28]			✓						✓	
Ertem vd., [29]				✓			✓	✓	✓	
Ozcelik vd., [30]				✓			✓	✓	✓	
Qi vd., [15]	✓				✓		✓		✓	
Wang vd., [16]	✓				✓			✓		
Lu vd., [17]	✓			✓				✓		
Zhang vd., [18]	✓		✓					✓		
Yue vd., [33]						✓		✓		
Cai vd., [19]		✓				✓			✓	
Wei [34]	✓					✓	✓		✓	
Jia, [20]		✓				✓	✓		✓	
Benmansour vd., [21]		✓				✓	✓		✓	
Cai ve Zhou, [22]		✓				✓	✓		✓	
Cai ve Zhou, [23]		✓			✓	✓	✓		✓	
Cai ve Zhou, [24]		✓			✓	✓	✓		✓	
Cai vd., [25]		✓				✓	✓		✓	
Bu çalışma						✓	✓		✓	✓

İS: İşlem Süreleri, HOZ: Hazır Olma Zamanları, GZ: Geliş Zamanı, HZ: Hazırlık Süreleri, AR: Arızalar, TZ: Teslim Zamanları, G: Gecikmeyi Dikkate Alan Amaç Fonksiyonları, SÇY: Sezgisel Çözüm Yaklaşımı, KÇY: Kesin Çözüm Yaklaşımı, MS: Metasezgisel Algoritmalar

Literatürde teslim zamanlarını stokastik ele alan çalışmalar birlikte değerlendirildiğinde bazı özel koşullar (belli bir dağılım ya da amaç fonksiyonuna sahip olmak gibi) altında eniyi çözümü garanti edebilen sıralama kuralları öne çıkmaktadır. Ancak sıralama kuralları doğası gereği sadece özel olarak geliştirildikleri problemler için başarılı çözümler üretebilirler, problemin yapısında değişiklikler yapıldığında ise başarılı çözümler üretmeleri beklenemez. Bu nedenle, stokastik tek makine çizelgeleme probleminin çözümü için daha genel çözüm yaklaşımlarına ihtiyaç vardır. Bu çalışmada iki aşamalı stokastik

Yue vd. [33] teslim zamanlarının stokastik olduğu tek makine çizelgeleme problemi için enbüyük gecikmenin yaşandığı senaryonun gecikmesinin enküçüklendiği bir matematiksel model ve gürbüz baskınlık kuralına dayanan sezgisel bir çözüm yaklaşımı önermişlerdir. Wei [34], enbüyük gecikmenin veya tamamlanma zamanlarının enküçüklendiği stokastik tek makine çizelgeleme probleminde öğrenme etkilerinin dikkate alındığı matematiksel modeller önermiştir.

programlama yaklaşımı önerilmiştir. Ancak problemin NP-zor [29] olması nedeniyle matematiksel model ile sadece küçük boyutlu problemler çözülebilmektedir. Büyük boyutlu problemlerin çözümü için de bir harmoni arama algoritması önerilmiştir. Erişilebilen literatür dikkate alındığında teslim zamanlarının stokastik olduğu ve toplam gecikmenin enküçüklendiği tek makine çizelgeleme problemi için ilk defa bu çalışmada bir metasezgisel algoritma önerilmiştir.

Çalışmanın izleyen bölümünde, Stokastik Tek Makine Çizelgeleme Problemi (STMÇP) tanımlanmış ve geliştirilen

matematiksel model verilmiştir. Üçüncü bölümde önerilen çözüm yaklaşımları sunulmuştur. Deneysel sonuçlar dördüncü bölümde, sonuç ve öneriler ise son bölümde tartışılmıştır.

2. Problem Tanımı ve Matematiksel Model

STMÇP'de, n iş, bir makinede işlem görmektedir. İş j 'nin işlem süresi (p_j) bilinmektedir. İşlerin hazırlık süreleri (s_{ij}) sıra bağımlıdır. İşler teslim zamanlarından sonra tamamlanırlarsa gecikmektedirler. Teslim zamanları, müşteriler tarafından güncellenebildiğinden stokastiktir. İşler bölünemez ve hazır olma zamanları sıfırdır. Bir makinede aynı anda sadece bir iş yapılabilir. Amaç fonksiyonu toplam gecikmenin enküçüklenmesidir.

Deterministik teslim zamanlı problemin matematiksel modeli (DTMÇ) ve bu modele ait tanım ve açıklamalar aşağıda verilmiştir;

İndisler:

$i, j \in N = \{1, 2, \dots, n\}$ iş

$k \in N = \{1, 2, \dots, n\}$ sıra

Parametreler:

p_j : iş j 'nin işlem süresi

h_j : iş j 'nin ilk hazırlık süresi

s_{ij} : iş i 'den iş j 'ye geçilirken gerekecek sıra bağımlı hazırlık süresi

d_j^D : iş j 'nin deterministik teslim zamanı

M : pozitif büyük bir sayı

Karar Değişkenleri:

x_{jk} : eğer iş j , sıra k 'ya atanırsa 1, atanmazsa 0

C_j : iş j 'nin tamamlanma zamanı

T_j : iş j 'nin gecikme süresi

(DTMÇ)

Amaç fonksiyonu:

$$enk z_D = \sum_j T_j \quad (1)$$

Kısıtlar:

$$\sum_j x_{jk} = 1 \quad \forall k \quad (2)$$

$$\sum_k x_{jk} = 1 \quad \forall j \quad (3)$$

$$C_j + M(1 - x_{jk}) \geq h_j + p_j \quad \forall j, k = 1 \quad (4)$$

$$C_j + M(2 - x_{jk} - x_{i(k-1)}) \geq C_i + p_j + s_{ij} \quad \forall i \neq j, k > 1 \quad (5)$$

$$T_j \geq C_j - d_j \quad \forall j \quad (6)$$

$$x_{jk} \in \{0,1\} \quad \forall j, k \quad (7)$$

$$C_j \geq 0 \quad \forall j \quad (8)$$

$$T_j \geq 0 \quad \forall j \quad (9)$$

Amaç (1) toplam gecikme süresinin enküçüklenmesidir. Denklem (2) her sıraya mutlaka bir işin atanmasını garanti etmektedir. Denklem (3) her işin mutlaka bir sıraya atanmasını sağlamaktadır. Denklem (4) ilk işin ve denklem (5) ikinci ve

sonraki sıralardaki işlerin tamamlanma zamanlarını hesaplatan kısıtlardır. Denklem (6) iş j 'nin gecikme süresini hesaplatma kısıtıdır. Denklem (7)-(9) karar değişkenlerine ait işaret kısıtlarıdır.

Problemin stokastik modeli (STMÇ) ve bu model için ek tanımlamalar aşağıda verilmiştir.

$w \in W = \{1, 2, \dots, w\}$ senaryo indisi

q_w : w . senaryonun gerçekleşme olasılığı

d_{jw} : iş j 'nin w . senaryodaki teslim zamanı

T_{jw} : iş j 'nin w . senaryodaki gecikme süresi

(STMÇ)

Amaç fonksiyonu:

$$enk z_S = \sum_w q_w \sum_j T_{jw} \quad (10)$$

Kısıtlar:

$$(2)-(5), (7), (8)$$

$$T_{jw} \geq C_j - d_{jw} \quad \forall j, w \quad (11)$$

$$T_{jw} \geq 0 \quad \forall j \quad (12)$$

Stokastik modelde amaç (10) beklenen toplam gecikme süresinin enküçüklenmesidir. Denklem (11) her işin her senaryodaki gecikme sürelerinin hesaplanmasını sağlamaktadır. Denklem (12) T_{jw} karar değişkenine ait işaret kısıtıdır.

3. Çözüm Yaklaşımı

Bu bölümde STMÇP'nin çözümü için geliştirilen iki aşamalı stokastik programlama yaklaşımı ve harmoni arama algoritması ayrıntıları ile açıklanmıştır.

3.1. İki Aşamalı Stokastik Programlama

Deterministik yaklaşımda, her bir parametre için tek bir değer kullanılmaktadır. Stokastik yaklaşımda ise, problem parametrelerindeki belirsizliği dikkate almak için parametreler rassal değişkenler olarak tanımlanmakta [29] ve parametrelerdeki değişkenlik senaryolar aracılığıyla hesaba katılabilmektedir. Bu çalışmada teslim zamanlarındaki değişkenlik senaryolar aracılığı ile ele alınmıştır. Her senaryo için, ilgili parametrenin varsayılan dağılımına uygun olarak rassal bir parametre değeri üretilmiştir.

Stokastik programlamada genellikle, problem bilinen yöntemlerle çözülebilmesi için, deterministik eşdeğerine dönüştürülür. Bu modellerin çözümünde sık kullanılan bir yaklaşım olan iki aşamalı stokastik programlamada, belirsizlik devam ederken ilk aşama kararı verilir, belirsizlik ortadan kalktıktan sonra, ikinci aşamada ise düzeltme kararı verilir [30]. Önerilen modelde, ilk aşama kararları, iş sıraları (x_{jk}) ve her bir işin tamamlanma zamanlarıdır (C_j). İkinci aşamada, teslim zamanları kesinleştikten (belirsizlik ortadan kalktıktan) sonra her bir senaryo için işlerin gecikme süreleri (T_{jw}) (düzeltme kararı) hesaplanır.

Stokastik problemlerin çözümlerinin değerlendirilmesinde kullanılan en önemli başarı kriterlerinden birisi Stokastik Çözümün Değeri (SÇD)'dir. SÇD, problemi stokastik ele almanın

katkısını göstermektedir. Ele alınan problem için SÇD değerini hesaplamakta kullanılan yöntemin adımları aşağıda verilmiştir.

Adım 1: STMÇ modelini tüm senaryoları dikkate alarak çöz ve amaç fonksiyonu değerini (z_S) hesapla.

Adım 2: DTMÇ modelini kullanarak problemi çöz ve iş sıralarını (x_{jk}) belirle.

Adım 3: Adım 2'de bulunan iş sıralarını (x_{jk}) sabitleyerek STMÇ modelini çöz ve amaç fonksiyonu değerini (z_S^D) hesapla.

Adım 4: SÇD değerini hesapla ($SÇD = z_S^D - z_S$).

3.2. Harmoni Arama Algoritması

Geem vd. [35] tarafından önerilen Harmoni Arama (HA) algoritması, popülasyon tabanlı bir metasezgiseldir. HA algoritması, bir müzisyenin tipik davranışını taklit ederek aday çözümleri geliştirir [36]. Müzisyenler harmoniyi oluştururken, genellikle hafızalarında depolanan müzik perdelerinin çeşitli olası kombinasyonlarını denerler. Bu mükemmel uyum arayışı, mühendislik problemlerine en başarılı çözümleri bulma prosedürüne benzer [37].

Harmoni arama algoritmasının adımları aşağıda verilmiştir:

Adım 1. Başlangıç Harmoni Arama Hafızası (HAH)'nı rassal oluştur ve değerlendir.

Adım 2. HAH'dan yeni bir çözüm türet ve değerlendir.

Adım 3. HAH'ı güncelle. Yeni çözüm, HAH'daki en kötü çözümden daha başarılı ise, bu çözümü en kötü çözümle değiştir. Aksi takdirde ele.

Adım 4. Durdurma koşulu sağlanmıyorsa Adım 2'ye git.

Çözüm gösterimi. Çözüm olarak işlerin π permütasyonu kullanılmaktadır. Örneğin, tek makine çizelgeleme probleminde sekiz iş olduğunu varsayalım. $\pi_i = [2, 8, 3, 1, 6, 5, 7, 4]$, sekiz işin bir permütasyonudur ve bu vektör, makinedeki işlerin sırasını temsil eder. Örneğin $\pi_4 = 1$, 1 numaralı işin 4. sıraya atıldığını göstermektedir.

HA hafızası. İlk HAH, belirli sayıda rassal oluşturulmuş çözümlerden oluşur. Sekiz işli bir problem için, Harmoni Arama Hafıza Boyutu (HAHB) altı ise, örnek bir HAH Şekil 1'de verildiği gibi gösterilebilir.

$$HAH = \begin{bmatrix} 4 & 1 & 7 & 8 & 2 & 6 & 3 & 5 \\ 1 & 6 & 3 & 8 & 5 & 7 & 2 & 4 \\ 2 & 1 & 6 & 7 & 5 & 4 & 8 & 3 \\ 4 & 2 & 3 & 1 & 6 & 8 & 5 & 7 \\ 7 & 3 & 5 & 2 & 4 & 8 & 6 & 1 \\ 7 & 4 & 5 & 8 & 3 & 2 & 1 & 6 \end{bmatrix}$$

Şekil 1. Örnek bir harmoni arama hafızası.

Figure 1. An example of a harmony search memory.

Yeni bir çözüm türetilmesi. Rassal türetilmiş bir HAH'dan başlayarak, her ardıştırmada yeni bir çözüm oluşturulur. Bu çözümün her bir elemanı, Harmoni Hafızasını Kullanma Oranı (HHKO)'na dayalı olarak elde edilir. HHKO, HAH üyelerinden bir eleman seçme olasılığı olarak tanımlanır ve dolayısıyla 1-HHKO, yeni çözümü rastgele türetme olasılığıdır. Permütasyon gösterimi kullanıldığından, her iş numarası bir çözüm vektörüne sadece bir kez yerleştirilebilir. Bu nedenle, elde edilen bir iş numarası daha önce oluşturulmuşsa, henüz oluşturulmamış bir iş numarası elde edilene kadar Denklem (13) kullanılarak onarılır. Burada π_i , çözüm vektörünün i . elemanıdır ve n , iş sayısıdır.

$$\pi_i = (\pi_i \bmod n) + 1 \quad (13)$$

Çözüm vektörünün elemanı HAH'dan geliyorsa, rastgele bir HAH üyesinin i . boyutundan seçilir ve ton ayarlama oranına (Pitching Adjust Rate) göre ayrıca mutasyona uğratılır. Ton ayarlama oranı, HAH'dan bir adayın mutasyona uğrama olasılığını belirler. Mutasyon için seçilen eleman Denklem (14) kullanılarak mutasyona uğratılır. Bu çalışmada, bant genişliği (bg) $[0.15n]$ formülü kullanılarak hesaplanmıştır. Bu formülde R $[1, bg]$ aralığında rassal bir tamsayıdır.

$$\pi_i = (\pi_i \bmod n) + R \quad (14)$$

Bu süreci açıklamak için kullanılacak bir örnek Şekil 2'de verilmiştir. Şekilde HAH'dan gelen elemanlar kare içine alınmıştır ve mutasyon için seçilen elemanlar ise gri ile işaretlenmiştir. Şekil 2'den de görülebileceği gibi yeni çözümün 3. ve 5. elemanları rassal olarak türetilmiştir. Diğerleri ise HAH'dan rassal olarak seçilmiştir. Yeni çözüm vektörünün 4. elemanı HAH'dan 2 olarak seçilmiştir. Ancak yeni çözüm vektörünün 2. elemanının değeri de 2'dir. Bu nedenle bir tamir mekanizmasına ihtiyaç vardır. 4. elemanın yeni değeri Denklem (13) kullanılarak, 3 olarak tamir edilmiştir. 7. eleman HAH'dan seçilmiştir ve mutasyona uğratılacaktır. Denklem (14) kullanılarak ilgili elemana mutasyon uygulanmış ve değeri 3 olmuştur. 3 sayısı daha önce çözümde yer aldığından tamir edilerek 5 değeri elde edilmiştir.

$$HAH = \begin{bmatrix} 4 & 1 & 7 & 8 & 2 & 6 & 3 & 5 \\ 1 & 6 & 3 & 8 & 5 & 7 & 2 & 4 \\ 2 & 1 & 6 & 7 & 5 & 4 & 8 & 3 \\ 4 & 2 & 3 & 1 & 6 & 8 & 5 & 7 \\ 7 & 3 & 5 & 2 & 4 & 8 & 6 & 1 \\ 7 & 4 & 5 & 8 & 3 & 2 & 1 & 6 \end{bmatrix}$$

$$YÇ = \begin{bmatrix} 7 & 2 & 6 & 3 & 4 & 8 & 5 & 1 \end{bmatrix}$$

Şekil 2. Harmoni arama hafızası kullanılarak yeni bir çözüm türetilmesi.

Figure 2. Generation of a new solution using harmony search memory.

HAH'ın güncellenmesi. Türetilen yeni çözümün amaç fonksiyonu değeri hesaplanır. Eğer yeni çözümün amaç fonksiyonu değeri HAH'n en kötü çözümünden daha iyi ise en kötü çözüm yeni çözümle değiştirilir.

Durdurma kriteri. Belirli bir iterasyon boyunca en başarılı çözüm değişmiyorsa algoritma durdurulur. Son nesildeki en başarılı çözüm nihai çözümdür. Bu çalışmada ilgili iterasyon sayısı 1.000.000 olarak alınmıştır.

4. Deneysel Sonuçlar

Önerilen matematiksel modellerin çözümünde GAMS 24.0.2 CPLEX çözücüsü kullanılmıştır. Geliştirilen HA algoritması ise Delphi'de kodlanmıştır. Test problemlerinin türetilme mekanizması ve elde edilen test sonuçları izleyen alt başlıklarda sunulmuştur.

4.1. Test Problemleri

Test problemlerinde iş sayısı 10, 20, 30, 40 ve 50 olarak belirlenmiştir. p_i , h_j ve s_{ij} parametreleri Saraç ve Tutumlu ([38-39])'nın çalışmalarında olduğu gibi kesikli düzgün dağılıma uygun olarak $[1,100]$ aralığında türetilmiştir. Teslim tarihi (d_j) ise Logendran vd. [40] önerdiğine benzer şekilde Kesikli Düzgün Dağılım (KDD) kullanılarak türetilmiştir. İlgili formül denklem (15)'de verilmiştir.

$$d_j^D \sim KDD \left[\sum_j p_j \left(1 - b_1 - \frac{b_2}{2} \right), \sum_j p_j \left(1 - b_1 + \frac{b_2}{2} \right) \right] \quad (15)$$

Denklem (15)'de yer alan b_1 ve b_2 parametreleri, teslim zamanlarının sıklığını kontrol etmektedir. b_2 değeri küçüldükçe teslim zamanı değeri de sıkılaşmaktadır. Bu çalışmada iki tip sıklık tanımlanmıştır. Birinci tipte $b_1 = 0,8$ ve $b_2 = 0,4$ olarak ikinci tipte $b_1 = 0,8$ ve $b_2 = 0,8$ olarak alınmıştır. Birinci tipte b_2 değeri daha küçük olduğu için birinci tip problemlerin teslim zamanları daha sıkıdır.

5 farklı iş sayısı ve iki sıklık tipi olmak üzere 10 farklı tip deterministik problem türetilmiştir. Her problem tipinden iki örnek olmak üzere toplam 20 ($5 \times 2 \times 2$) deterministik problem türetilmiştir. Stokastik problem setleri, deterministik problem parametreleri kullanılarak oluşturulmuştur: stokastik teslim zamanları, %10 ve %30 değişkenlik içerecek şekilde sırasıyla $[0,9d_j^D; 1,1d_j^D]$ ve $[0,7d_j^D; 1,3d_j^D]$ aralıklarında rassal olarak türetilmiş ve sırasıyla s1 ve s2 olarak kodlanmıştır. Toplamda 20 adet stokastik problem tipi olmak üzere her bir problem tipinden iki örnek türetilmiştir. Toplamda 40 ($5 \times 2 \times 2 \times 2$) adet stokastik problem oluşturulmuştur. Stokastik test problemleri, sırasıyla iş sayısı, teslim zamanı sıklık parametresi, örnek numarası ve değişkenlik dikkate alınarak kodlanmıştır. Örneğin, 10-1-2-s1, 10 işin olduğu, teslim zamanlarının sıkı ve değişkenlik tipinin s1 olduğu problemin ikinci örneğini göstermektedir. Deterministik test problemlerinin gösteriminde değişkenlik tipi yerine problemin deterministik olduğunu göstermek amacıyla 'd' kodu kullanılmıştır.

Tablo 2. DTMÇ ve STMÇ modellerinin GAMS/CPLEX ile çözülmesi ile elde edilen sonuçlar.

Table 2. Results obtained by solving DTMÇ and STMÇ models with GAMS/CPLEX.

problem	d		s1		s2	
	z_D	t_D	z_{s1}	t_{s1}	z_{s2}	t_{s2}
10-1-1	2132	379	2138,73	1330	2110,30	1285
10-1-2	2089	478	2089,67	3200	2085,60	2043
10-2-1	2396	441	2389,23	2151	2382,07	2102
10-2-2	2292	630	2294,90	1593	2277,43	1887
20-1-1	8118	7200	8509,50	7200	8457,23	7200
20-1-2	7422	7200	7313,20	7200	8419,30	7200
20-2-1	6840	7200	6439,90	7200	7252,00	7200
20-2-2	5992	7200	6842,27	7200	6274,10	7200
30-1-1	16556	7200	17509,40	7200	17873,20	7200
30-1-2	17199	7200	15920,67	7200	16052,90	7200
30-2-1	13283	7200	14163,50	7200	14141,73	7200
30-2-2	15844	7200	17062,73	7200	16932,47	7200
40-1-1	32492	7200	33413,00	7200	37815,23	7200
40-1-2	28473	7200	28036,17	7200	30557,10	7200
40-2-1	25250	7200	27756,90	7200	26511,27	7200
40-2-2	26731	7200	26917,50	7200	27245,93	7200
50-1-1	53263	7200	56124,30	7200	53501,53	7200
50-1-2	39312	7200	40182,30	7200	50252,97	7200
50-2-1	44303	7200	55393,73	7200	54453,30	7200
50-2-2	44055	7200	48151,20	7200	54328,43	7200
100-1-1	284641	7200	361133,7	7200	385656,53	7200
150-1-1	966263	7200	-	7200	-	7200
200-1-1	-	7200	-	7200	-	7200

4.3. Geliştirilen HA Algoritmasının Performansı

Tüm test problemlerine GAMS/CPLEX ile süre limiti içinde uygun çözümler bulunamadığından, geliştirilen HA algoritmasının performansının değerlendirilebilmesi için ikinci bir çözüm

Ayrıca geliştirilen matematiksel modelin ve HA algoritmasının daha büyük boyutlu problemleri çözümedeki performansını görebilmek amacıyla 100, 150 ve 200 iş içeren büyük boyutlu problemler türetilmiştir. Büyük boyutlu problemlerin tümünde, sıklık parametresi 1 olarak alınmıştır. Her problem için iki tip değişkenlik olmak üzere toplam 3 adet deterministik ve 6 adet stokastik büyük boyutlu problem türetilmiştir.

4.2. Önerilen Matematiksel Modellerin Performansı

Test problemleri DTMÇ ve STMÇ modelleri kullanılarak 7200 saniye süre limiti ile çözülmüştür. GAMS/CPLEX ile elde edilen sonuçlar Tablo 2'de verilmiştir. Tablo 2 dört bölümden oluşmaktadır. İlk bölümde test problemlerinin adı, ikinci bölümde DTMÇ modelinin GAMS/CPLEX ile çözülmesiyle elde edilen amaç fonksiyonu değerleri (z_D) ve çözüm süreleri (t_D), üçüncü ve dördüncü bölümde ise sırasıyla s1 ve s2 değişkenlik tipli stokastik problemlerin STMÇ ile elde edilen amaç fonksiyonu değerleri (z_{s1} , z_{s2}) ve çözüm süreleri (t_{s1} , t_{s2}) verilmiştir.

Tablo 2'den de görülebileceği gibi, DTMÇ ve STMÇ modelleri ile sadece 10 boyutlu problemlerin eniyi çözümleri elde edilebilmiştir. 150-1-1'in hem s1 hem de s2 değişkenlik tipli stokastik problemlerine STMÇ modeli ve GAMS/CPLEX ile süre limiti içinde uygun çözüm bulunamamıştır. 200-1-1'in deterministik, s1 ve s2 değişkenlik tipli stokastik problemlerine de süre limiti içinde uygun çözüm bulunamamıştır.

yöntemine gereksinim duyulmuştur. Bu amaçla, tek makine çizelgeleme problemlerinde yaygın olarak kullanılan EDD (Earliest Due Date) sıralama kuralı, problemin teslim zamanlarının senaryo bazlı olması nedeniyle beklenen teslim zamanlarını dikkate alacak şekilde EEDD (Expected Earliest Due

Date) olarak kullanılmıştır. Ancak EEDD ile çok başarılı çözümler elde edilemediğinden bu sıralama kuralı uygulandıktan sonra 2-opt iyileştirme algoritması kullanılarak EEDD ile elde edilen çözümler iyileştirilmiştir. Deterministik, s1 ve s2 değişkenlik tipli stokastik problemlere ait sonuçlar, üç bölüm halinde Tablo 3'te verilmiştir.

GAMS/CPLEX ya da EEDD+2-opt ile elde edilen amaç fonksiyonlarından hangisi daha başarılı ise bu değerler her bölümün ilk sütununda (z_D , z_{s1} , z_{s2}) verilmiştir. İkinci sütunlarda ise test problemlerinin HA algoritması ile 20 tekrarlı olarak çözülmesiyle elde edilen en başarılı sonuçlar (z_D^{HA} , z_{s1}^{HA} , z_{s2}^{HA}) sunulmuştur. Geliştirilen HA algoritmasının matematiksel model ya da EEDD+2-opt algoritmasından en başarılısını ne kadar iyileştirdiğini gösterebilmek amacıyla iyileştirme oranları (% $\bar{i}$) hesaplanmıştır. Denklem (16)'de verilen formül ile hesaplanan % $\bar{i}$ değerleri bölümlerin son sütunlarında verilmiştir.

$$\% \bar{i} = 100 \frac{z_D - z_D^{HA}}{z_D} \quad (16)$$

Tablo 3. Harmoni arama algoritması sonuçlarının karşılaştırılması.

Table 3. Comparison of harmony search algorithm results.

problem	z_D	z_D^{HA}	% $\bar{i}$	z_{s1}	z_{s1}^{HA}	% $\bar{i}$	z_{s2}	z_{s2}^{HA}	% $\bar{i}$
10-1-1	2132	2132	0,0	2138,7	2138,7	0,0	2110,3	2110,3	0,0
10-1-2	2089	2089	0,0	2089,7	2089,7	0,0	2085,6	2085,6	0,0
10-2-1	2396	2396	0,0	2389,2	2389,2	0,0	2382,1	2382,1	0,0
10-2-2	2292	2292	0,0	2294,9	2294,9	0,0	2277,4	2277,4	0,0
20-1-1	8118	7257	10,6	8509,5	7203,5	15,3	8457,2	7253,4	14,2
20-1-2	7422	6987	5,9	7313,2	6900,2	5,6	8419,3	6760,6	19,7
20-2-1	6840	6533	4,5	6439,9	6205,5	3,6	7252,0	6528,2	10,0
20-2-2	5992	5657	5,6	6842,3	5653,8	17,4	6274,1	5643,5	10,1
30-1-1	16556	14265	13,8	17509,4	13957,7	20,3	17873,2	14042,3	21,4
30-1-2	17199	14222	17,3	15920,7	13398,3	15,8	16052,9	13856,3	13,7
30-2-1	13283	12289	7,5	14163,5	11581,3	18,2	14141,7	11941,1	15,6
30-2-2	15844	14887	6,0	17062,7	14757,5	13,5	16932,5	14509,8	14,3
40-1-1	32492	30869	5,0	33413,0	29569,7	11,5	37815,2	30646,6	19,0
40-1-2	28473	25093	11,9	28036,2	24736,8	11,8	30557,1	24037,5	21,3
40-2-1	25250	24036	4,8	27756,9	23457,0	15,5	26511,3	23682,0	10,7
40-2-2	26731	24341	8,9	26917,5	22835,5	15,2	27245,9	21895,9	19,6
50-1-1	53263	50665	4,9	56124,3	48247,0	14,0	53501,5	48451,6	9,4
50-1-2	39312	39669	-0,9	40182,3	37390,7	6,9	50253,0	38685,6	23,0
50-2-1	44303	46349	-4,6	55393,7	43446,9	21,6	54453,3	45509,6	16,4
50-2-2	44055	43344	1,6	48151,2	41396,9	14,0	54328,4	41659,6	23,3
100-1-1	367962	276469	24,9	381192,3	270137,1	29,1	363537,2	271114,5	25,4
150-1-1	889688	705663	20,7	875831,4	698909,3	20,2	817704,0	700686,8	14,3
200-1-1	1517660	1319468	13,1	1504887,5	1317471,0	12,5	1482616,3	1263950,6	14,7
		ort.	7,0		ort.	12,3		ort.	13,7

4.4. Problemi Stokastik Ele Almanın Katkısı

Problemi stokastik ele almanın katkısını gösterebilmek amacıyla SÇD değerleri hesaplanmıştır. Matematiksel modellerden daha başarılı çözümler elde ettiği ve tüm problemleri çözebildiği için SÇD değerlerinin hesaplanmasında HA algoritması kullanılmıştır.

Problemi stokastik ele almanın katkısını gösteren SÇD değerlerini aynı ölçeğe getirmek için her bir problem için iyileşme yüzdesi (Δ_{si}) hesaplanmıştır. İyileşme yüzdesi formülü (17) numaralı denklemde verilmiştir.

$$\Delta_{si} = 100 \frac{SÇD_{si}}{z_{si}^D} \quad (17)$$

Tüm test problemleri için stokastik çözümün değeri ve iyileşme yüzdesi Tablo 4'de verilmiştir. Tablo, dört bölüme ayrılmıştır. İlk bölümde test problemlerinin adı, ikinci bölümde deterministik problemin HA algoritması ile

Tablo 3'ten de görülebileceği gibi, önerilen HA algoritması 10 boyutlu problemlerin tümünde eniyi çözümlere erişebilmiştir. HA algoritması sadece 50-1-2 ve 50-2-1 deterministik problemleri için GAMS/CPLEX'ten daha iyi çözümlere ulaşamamıştır. Geri kalan tüm problemler için GAMS/CPLEX ya da EEDD+2-opt. ile elde edilmiş olan çözümlerden daha başarılı sonuçlara ulaşılmıştır. Önerilen HA algoritması ile deterministik problemlerin sonuçları ortalama %7, s1 değişkenlik tipli stokastik problemlerin sonuçları ortalama %12,3 ve s2 değişkenlik tipli stokastik problemlerin sonuçları ortalama %13,7 oranında iyileştirilmiştir.

Çözüm süreleri açısından karşılaştırma yapıldığında ise, HA algoritmasının en uzun çözüm süresi 118 saniye iken GAMS/CPLEX, 20 ve daha büyük boyutlu test problemlerinin hepsinde süre limiti olan 7200 saniyeyi kullanmıştır. Sonuç olarak HA algoritması GAMS/CPLEX'ten daha kısa sürede daha iyi sonuçlara ulaşabilmiştir.

çözülmesiyle elde edilen amaç fonksiyonu değerleri (z_D) ve çözüm süreleri (t_D), üçüncü ve dördüncü bölümde ise sırasıyla s1 ve s2 değişkenlik tipli stokastik problemlerin HA algoritması ile elde edilen amaç fonksiyonu değerleri (z_{s1} , z_{s2}), çözüm süreleri (t_{s1} , t_{s2}), SÇD değerleri ve iyileşme yüzdesi (Δ_{s1} , Δ_{s2}) verilmiştir.

Tablo 4, elde edilen iyileşme yüzdesi açısından incelendiğinde, teslim zamanlarını stokastik olarak ele almanın toplam gecikmeyi %10'a kadar azaltabildiği görülmüştür. Sonuçlar problemlerin değişkenlik tipleri açısından incelendiğinde, s1 değişkenlik tipli problemlerde ortalama iyileşme oranı %2,67 iken bu oran s2 değişkenlik tipli problemlerde %2,42 olarak gerçekleşmiştir. Elde edilen sonuçlar, problemi stokastik ele almanın amaç fonksiyonunun değerini iyileştirdiğini ortaya koymuştur.

Tablo 4. Stokastik çözümün değeri ve iyileşme yüzdeleri.**Table 4.** Value of stochastic solution and improvement percentages.

problem	d		s1				s2			
	z_D	t_D	z_{s1}	t_{s1}	SCD_{s1}	Δ_{s1}	z_{s2}	t_{s2}	SCD_{s2}	Δ_{s2}
10-1-1	2132	1	2138,73	1	0,00	0,00	2110,30	1	0,00	0,00
10-1-2	2089	1	2089,67	1	0,00	0,00	2085,60	1	0,00	0,00
10-2-1	2396	1	2389,23	1	0,00	0,00	2382,07	1	0,00	0,00
10-2-2	2292	1	2294,90	1	0,00	0,00	2277,43	1	0,00	0,00
20-1-1	7257	1	7203,53	1	36,00	0,50	7253,40	1	43,93	0,60
20-1-2	6987	1	6900,20	1	100,23	1,43	6760,60	1	277,13	3,94
20-2-1	6533	1	6205,47	1	341,56	5,22	6528,17	1	55,13	0,84
20-2-2	5657	1	5653,80	2	0	0	5643,50	1	50,30	0,88
30-1-1	14265	4	13957,70	10	314,23	2,20	14042,27	12	204,56	1,44
30-1-2	14222	7	13398,30	13	827,87	5,82	13856,27	8	340,23	2,40
30-2-1	12289	3	11581,30	7	724,07	5,88	11941,13	15	430,67	3,48
30-2-2	14887	7	14757,47	12	175,90	1,18	14509,77	13	449,93	3,01
40-1-1	30869	11	29569,73	5	1310,84	4,24	30646,63	20	219,67	0,71
40-1-2	25093	10	24736,77	16	323,20	1,29	24037,50	33	1109,30	4,41
40-2-1	24036	13	23457,03	16	572,57	2,38	23681,97	23	622,93	2,56
40-2-2	24341	5	22835,47	21	1528,13	6,27	21895,93	39	2488,37	10,20
50-1-1	50665	17	48246,97	36	2424,86	4,79	48451,57	31	2356,50	4,64
50-1-2	39669	18	37390,70	17	2293,73	5,78	38685,60	43	1083,33	2,72
50-2-1	46349	14	43446,87	55	2889,20	6,24	45509,63	42	795,17	1,72
50-2-2	43344	10	41396,90	41	2017,10	4,65	41659,60	49	2170,97	4,95
100-1-1	276469	47	270137,13	102	6284,84	2,27	271114,53	92	5475,87	1,98
150-1-1	705663	16	698909,30	118	6950,67	0,98	700686,80	24	6481,67	0,92
200-1-1	1319468	3	1317471,00	87	2410,37	0,18	1263950,57	87	55690,20	4,22
					ort.	2,67			ort.	2,42

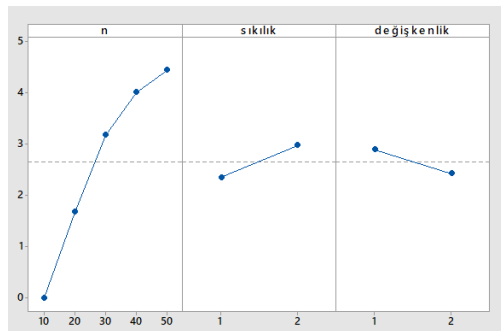
İş sayısı (n), teslim zamanlarının sıklığı (sıklık) ve değişkenliği (değişkenlik) faktörlerinin iyileşme yüzdesi üzerindeki etkisini inceleyebilmek amacıyla varyans analizi yapılmıştır. Varyans analizi tablosu Tablo 5'te, ana etkiler grafiği ise Şekil 3'te verilmiştir.

Tablo 5. Varyans analizi**Table 5.** Analysis of variance

Source	DF	Adj SS	Adj MS	F-value	P-value
n	4	106,246	26,5614	6,16	0,001
sıklık	1	3,938	3,9376	0,91	0,349
değişkenlik	1	2,195	2,1949	0,51	0,482
n*sıklık	4	10,936	2,7341	0,63	0,643
n*değişkenlik	4	9,336	2,3341	0,54	0,707
sıklık*değişkenlik	1	0,026	0,0255	0,01	0,939
Error	24	103,516	4,3132		
Lack-of-Fit	4	8,951	2,2378	0,47	0,755
Pure Error	20	94,565	4,7282		
Total	39	236,192			

Tablo 5'ten de görülebileceği gibi sadece iş sayısı faktörü kritiktir (p-değeri(0,001)<0,05). Şekil 3 incelendiğinde iş sayısı arttıkça problemi stokastik ele almanın katkısının da ciddi bir şekilde arttığı görülmektedir.

Yapılan deneysel çalışmalar, stokastik teslim zamanlarına sahip tek makine çizelgeleme problemini stokastik ele almanın, özellikle problemin boyutu arttıkça daha çok katkı sağladığını ortaya koymuştur.

**Şekil 3.** Ana etkiler grafiği.**Figure 3.** Main effects plot.

5. Sonuç ve Öneriler

İş gecikmeleri müşteri memnuniyetini doğrudan olumsuz etkilediği için, pek çok işletme gecikmelerini mümkünse önlemek mümkün değilse de olabildiğince azaltmak istemektedir. Bu nedenle toplam gecikmelerin en küçüklenmesi amacı makine çizelgeleme problemlerinde en çok ele alınan önemli amaç fonksiyonlarından birisidir. Bir işin gecikmesi, teslim zamanından sonra tamamlandığında ortaya çıkar. Gecikmelerin doğru hesaplanabilmesi için eğer varsa teslim zamanlarındaki değişkenliğin dikkate alınması gerekir. Değişkenlik dikkate alınmadığında, bazı ürünler gecikebilmekte bazıları ise zamanından önce üretilebilmektedir. Bu çalışmada, teslim zamanlarının stokastik ele alınmasının toplam gecikme üzerindeki etkisi araştırılmıştır.

Ele alınan problem stokastik teslim zamanlı tek makine çizelgeleme problemidir. Bu problemin çözümü için iki aşamalı bir stokastik programlama yöntemi ve bir harmoni arama algoritması önerilmiştir. Önerilen çözüm yaklaşımlarının performansları rassal türetilen test problemleri kullanılarak gösterilmiştir. Elde edilen test sonuçları, teslim zamanlarını stokastik ele almanın toplam gecikmeyi %10'a kadar azaltabileceğini ortaya koymuştur. Gelecekte stokastik teslim zamanlarının toplam gecikme üzerindeki etkisi farklı makine ortamları için de incelenebilir ayrıca belirsizlik, stokastik yapı yerine bulanık, olabirlikli, etkileşimli bulanık ve/veya etkileşimli olabirlikli olarak dikkate alınabilir.

Etik kurul onayı ve çıkar çatışması beyanı

Hazırlanan makalede etik kurul izni alınmasına gerek yoktur. Hazırlanan makalede herhangi bir kişi/kurum ile çıkar çatışması bulunmamaktadır.

Yazar katkılarının beyanı

Tuğba Saraç, makale fikrinin oluşturulması, literatür taraması, matematiksel modelin ve harmoni arama algoritmasının geliştirilmesi, test problemlerinin türetilmesi, analizlerin gerçekleştirilmesi ve makalenin yazımı aşamalarında katkı sağlamıştır. Feriştah Özçelik, makale fikrinin oluşturulması, literatür taraması, matematiksel modelin ve harmoni arama

algoritmasının geliştirilmesi, test problemlerinin türetilmesi, analizlerin gerçekleştirilmesi ve makalenin yazımı aşamalarında katkı sağlamıştır.

Kaynakça

- [1] Jang, W., Klein, C.M., 2002. Minimizing the expected number of tardy jobs when processing times are normally distributed. *Operations Research Letters*, Vol.30, pp.100–106. DOI: [https://doi.org/10.1016/S0167-6377\(02\)00110-4](https://doi.org/10.1016/S0167-6377(02)00110-4).
- [2] Seo, D.K., Klein, C.M., Jang, W., 2005. Single machine stochastic scheduling to minimize the expected number of tardy jobs using mathematical programming models. *Computers & Industrial Engineering*, Vol.48, pp.153–161. DOI: <https://doi.org/10.1016/j.cie.2005.01.002>.
- [3] Van den Akker, M., Hoogeveen, H., 2008. Minimizing the number of late jobs in a stochastic setting using a chance constraint. *Journal of Scheduling*, Vol.11, pp.59–69. DOI: <https://doi.org/10.1007/s10951-007-0034-8>.
- [4] Elyasi, A., Salmasi, N., 2013. Stochastic scheduling with minimizing the number of tardy jobs using chance constrained programming. *Mathematical and Computer Modelling*, Vol.57, pp.1154–1164. DOI: <https://doi.org/10.1016/j.mcm.2012.10.017>.
- [5] Elyasi, A., Salmasi, N., 2013. Due date assignment in single machine with stochastic processing times. *International Journal of Production Research*, Vol.51(8), pp.2352–2362. DOI: <https://doi.org/10.1080/00207543.2012.737945>.
- [6] Soroush, H.M., 2007. Minimizing the weighted number of early and tardy jobs in a stochastic single machine scheduling problem. *European Journal of Operational Research*, Vol.181, pp.266–287. DOI: <https://doi.org/10.1016/j.ejor.2006.05.036>.
- [7] Soroush, H.M., 2010. Solving a stochastic single machine problem with initial idle time and quadratic objective. *Computers & Operations Research*, Vol.37, pp.1328–1347. DOI: <https://doi.org/10.1016/j.cor.2009.10.002>.
- [8] Salmasnia, A., Khatami, M., Kazemzadeh, R.B., Zegordi, S.H., 2015. Bi-objective single machine scheduling problem with stochastic processing times. *TOP*, Vol.23, pp.275–297. DOI: <https://doi.org/10.1007/s11750-014-0337-9>.
- [9] Chang, Z., Song, S., Zhang, Y., Ding, J.Y., Zhang, R., Chiong, R., 2017. Distributionally robust single machine scheduling with risk aversion. *European Journal of Operational Research*, Vol.256, pp.261–274. DOI: <https://doi.org/10.1016/j.ejor.2016.06.025>.
- [10] Atakan, S., Bülbül, K., Noyan, N., 2017. Minimizing value-at-risk in single-machine scheduling. *Annals of Operations Research*, Vol.248, pp.25–73. DOI: <https://doi.org/10.1007/s10479-016-2251-z>.
- [11] Lemos, R.F., Ronconi, D.P., 2015. Heuristics for the stochastic single-machine problem with E/T costs. *International Journal of Production Economics*, Vol.168, pp.131–142. DOI: <https://doi.org/10.1016/j.ijpe.2015.06.014>.
- [12] Iranpoor, M., Fatemi Ghomi, S.M.T., Zandieh, M., 2013. Due-date assignment and machine scheduling in a low machine-rate situation with stochastic processing times. *Computers & Operations Research*, Vol.40, pp.1100–1108. DOI: <https://doi.org/10.1016/j.cor.2012.11.013>.
- [13] Baker, K.R., 2014. Minimizing earliness and tardiness costs in stochastic scheduling. *European Journal of Operational Research*, Vol.236, pp.445–452. DOI: <https://doi.org/10.1016/j.ejor.2013.12.011>.
- [14] Qi, X.D., Yin, G., Birge, J.R., 2000. Scheduling problems with random processing times under expected earliness/tardiness costs. *Stochastic Analysis and Applications*, Vol.18(3), pp.453–473. DOI: <https://doi.org/10.1080/07362990008809680>.
- [15] Qi, X.D., Yin, G., Birge, J.R., 2000. Single-machine scheduling with random machine breakdowns and randomly compressible processing times. *Stochastic Analysis and Applications*, Vol.18(4), pp.635–653. DOI: <https://doi.org/10.1080/07362990008809689>.
- [16] Wang, D.J., Liu, F., Wang, Y.Z., Jin, Y., 2015. A knowledge-based evolutionary proactive scheduling approach in the presence of machine breakdown and deterioration effect. *Knowledge-Based Systems*, Vol.90, pp.70–80. DOI: <https://doi.org/10.1016/j.knsys.2015.09.032>.
- [17] Lu, C.C., Lin, S.W., Ying, K.C., 2012. Robust scheduling on a single machine to minimize total flow time. *Computers & Operations Research*, Vol.39, pp.1682–1691. DOI: <https://doi.org/10.1016/j.cor.2011.10.003>.
- [18] Zhang, L., Lin, Y., Xiao, Y., Zhang, X., 2018. Stochastic single-machine scheduling with random resource arrival times. *International Journal of Machine Learning and Cybernetics*, Vol.9(7), pp.1101–1107. DOI: <https://doi.org/10.1007/s13042-016-0631-y>.
- [19] Cai, X., Wang, L., Zhou, X., 2007. Single-machine scheduling to stochastically minimize maximum lateness. *Journal of Scheduling*, Vol.10, pp.293–301. DOI: <https://doi.org/10.1007/s10951-007-0026-8>.
- [20] Jia, C., 2001. Stochastic single machine scheduling with an exponentially distributed due date. *Operations Research Letters*, Vol.28, pp.199–203. DOI: [https://doi.org/10.1016/S0167-6377\(01\)00065-7](https://doi.org/10.1016/S0167-6377(01)00065-7).
- [21] Benmansour, R., Allaoui, H., Artiba, A., 2012. Stochastic single machine scheduling with random common due date. *International Journal of Production Research*, Vol.50(13), pp.3560–3571. DOI: <https://doi.org/10.1080/00207543.2012.671589>.
- [22] Cai, X., Zhou, X., 2005. Single-machine scheduling with exponential processing times and general stochastic cost functions. *Journal of Global Optimization*, Vol.31, pp.317–332. DOI: <https://doi.org/10.1007/s10898-004-5702-z>.
- [23] Cai, X., Zhou, X., 2000. Asymmetric earliness and tardiness scheduling with exponential processing times on an unreliable machine. *Annals of Operations Research*, Vol.98(1–4), pp.313–331. DOI: <https://doi.org/10.1023/A:1019220826984>.
- [24] Cai, X., Zhou, X., 2006. Stochastic scheduling with asymmetric earliness and tardiness penalties under random machine breakdowns. *Probability in the Engineering and Informational Sciences*, Vol.20, pp.635–654. DOI: <https://doi.org/10.1017/S0269964806060396>.
- [25] Cai, X., Wu, X., Zhou, X., 2007. Single-machine scheduling with general costs under compound-type distributions. *Journal of Scheduling*, Vol.10(1), pp.77–84. DOI: <https://doi.org/10.1007/s10951-006-0327-3>.
- [26] Zhang, J., Yang, W., Tu, Y., 2013. Scheduling with compressible and stochastic release dates. *Computers & Operations Research*, Vol.40, pp.1758–1765. DOI: <https://doi.org/10.1016/j.cor.2013.01.011>.
- [27] Ronconi, D.P., Powell, W.B., 2010. Minimizing total tardiness in a stochastic single machine scheduling problem using approximate dynamic programming. *Journal of Scheduling*, Vol.13, pp.597–607. DOI: <https://doi.org/10.1007/s10951-009-0160-6>.
- [28] Xu, L., Wang, Q., Huang, S., 2015. Dynamic order acceptance and scheduling problem with sequence-dependent setup time. *International Journal of Production Research*, Vol.53(19), pp.5797–5808. DOI: <https://doi.org/10.1080/00207543.2015.1005768>.
- [29] Ertem, M., Ozcelik, F., Saraç, T., 2019. Single machine scheduling problem with stochastic sequence-dependent setup times. *International Journal of Production Research*, Vol.57(10), pp.3273–3289. DOI: <https://doi.org/10.1080/00207543.2019.1581383>.
- [30] Ozcelik, F., Ertem, M., Saraç, T., 2022. A stochastic approach for the single-machine scheduling problem to minimize total expected cost with client-dependent tardiness costs. *Engineering Optimization*, Vol.54(7), pp.1178–1192. DOI: <https://doi.org/10.1080/0305215X.2021.1919098>.
- [31] Roohnavazfar, M., Manerba, D., Fotio Tiotop, L., Reza Pasandideh, S.H., Tadei, R., 2021. Stochastic single machine scheduling problem as a multi-stage dynamic random decision process. *Computational Management Science*, Vol.18, pp.267–297. DOI: <https://doi.org/10.1007/s10287-020-00386-1>.
- [32] Gu, J., Gu, M., Gu, X., 2014. Optimal rules for single machine scheduling with stochastic breakdowns. *Mathematical Problems in Engineering*, Vol.2014, Article ID: 260415. DOI: <https://doi.org/10.1155/2014/260415>.
- [33] Yue, F., Song, S.J., Jia, P., Wu, G.P., Zhao, H., 2020. Robust single machine scheduling problem with uncertain job due dates for industrial mass production. *Journal of Systems Engineering and Electronics*, Vol.31(2), pp.350–358. DOI: <https://doi.org/10.23919/JSEE.2020.000012>.
- [34] Wei, W., 2019. Single machine scheduling with stochastically dependent times. *Journal of Scheduling*, Vol.22, pp.677–689. DOI: <https://doi.org/10.1007/s10951-019-00600-2>.
- [35] Geem, Z.W., Kim, J.H., Loganathan, G.V., 2001. A new heuristic optimization algorithm: Harmony search. *Simulation*, Vol.76(2), pp.60–68. DOI: <https://doi.org/10.1177/003754970107600201>.
- [36] Zammori, F., Braglia, M., Castellano, D., 2014. Harmony search algorithm for single-machine scheduling problem with planned maintenance. *Computers & Industrial Engineering*, Vol.76, pp.333–346. DOI: <https://doi.org/10.1016/j.cie.2014.08.001>.
- [37] Gao, K.Z., Suganthan, P.N., Pan, Q.K., Tasgetiren, M.F., 2015. An effective discrete harmony search algorithm for flexible job shop scheduling problem with fuzzy processing time. *International Journal of Production Research*, Vol.53, pp.5896–5911. DOI: <https://doi.org/10.1080/00207543.2015.1020174>.
- [38] Saraç, T., Tutumlu, B., 2022. A bi-objective mathematical model for an unrelated parallel machine scheduling problem with job-splitting. *Journal of the Faculty of Engineering and Architecture of Gazi University*, Vol.37(4), pp.2293–2308. DOI: <https://doi.org/10.17341/gazimmfd.967343>.
- [39] Saraç, T., Tutumlu, B., 2022. A mixed-integer programming model and solution approach to determine the optimum machine number in the unrelated parallel machine scheduling problem. *Journal of the Faculty of*

- Engineering and Architecture of Gazi University, Vol.37(1), pp.329–345.
DOI: <https://doi.org/10.17341/gazimmfd.686683>.
- [40] Logendran, R., McDonell, B., Smucker, B., 2006. Scheduling unrelated parallel machines with sequence-dependent setups. *Applied Mathematics and Computation*, Vol.181(2), pp.1008–1017. DOI: <https://doi.org/10.1016/j.amc.2006.02.048>.