

Solution of Resource Constrained Project Scheduling Problem by Scanning the Whole Search Domain

Onder Halis Bettemir*, Derya Cakmak

Inonu University, Engineering Faculty, Department of Civil Engineering, Malatya, 44280, Turkey

*Received Date: Feb11, 2021

*Revised Date: Oct 20, 2021

*Accepted Date: Oct 28, 2021

*Published Online: Nov 16, 2021

Abstract

Resource constrained project scheduling problem is a prevalent problem for manufacturing and construction sectors. Resource allocation is redone from the beginning in the construction sector for each construction project; therefore project planners may not have adequate time to obtain optimum solution. Moreover, employment of civil engineers who are talented in the optimization task is less probable in small-scale construction firms. As a result of this, resource overrun problems are solved without converging optimum solution. In this study, a complete enumeration based algorithm which can solve resource constrained project scheduling problems without implementing complex optimization methods is implemented. The algorithm is programmed on spreadsheet and small-scale problems are solved in order to represent the proposed algorithm can obtain the optimum solution. Search domain expands exponentially and solution time excessively prolongs when the number of activities in the project increases because the resource constrained project scheduling problem is NP-Hard. Therefore in this study small-scale problems are solved. However, elimination of infeasible solutions and parallel computing can be inserted as further study to decrease the computing time. Thence solution of larger scale problems would be possible. It is considered that the construction sector may implement the proposed algorithm since the executed six case studies are solved within one minute and the implementation of the proposed method does not require complex optimization algorithms.

Keywords

Optimization, resource-constrained project scheduling problem, project management.

*Corresponding Author: Onder Halis Bettemir, onder.bettemir@inonu.edu.tr,  [0000-0002-5692-7708](https://orcid.org/0000-0002-5692-7708)

Tüm Arama Uzayını Tarayarak Küçük Ölçekli Kaynak Kısıtlı Proje Çizelgeleme Problemlerinin Çözümü

Önder Halis Bettemir*, Derya Çakmak

İnönü Üniversitesi, Mühendislik Fakültesi, İnşaat Mühendisliği, Malatya, 44280, Türkiye

•Gönderi Tarihi: 11 Şub 2021

•Düzeltilme Tarihi: 20 Eki 2021

•Kabul Tarihi: 28 Eki 2021

•Çevrimiçi Yayın Tarihi: 16 Kas 2021

Özet

Kaynak kısıtlı proje çizelgeleme problemi özellikle imalat ve inşaat sektöründe yaygın olarak karşılaşılan bir problemdir. Proje tabanlı faaliyet gösteren inşaat sektöründe kaynak tahsisi her proje için baştan yapıldığı için planlamacılar kaynak kısıtlarının ihlalini düzenlerken en iyi çözümü bulmak için yeterli vakte sahip olmayabilirler. Ayrıca küçük ölçekli inşaat işlerini gerçekleştiren firmalarda optimizasyon alanında uzman inşaat mühendislerinin istihdam edilmesi düşük olasılıktır. Bunun sonucunda kaynak kısıtlarını ihlal etmeyen fakat optimum çözümden uzak bir çözüm elde edebilirler. Bu çalışmada karmaşık optimizasyon yöntemlerini kullanmadan tüm proje çizelgeleme olasılıklarını deneyerek kaynak kısıtlı proje çizelgeleme problemini çözen bir algoritma geliştirilmiştir. Algoritma hesap tablosu üzerinde programlanmış ve küçük ölçekli projelerde denenerek çalıştığı gösterilmiştir. Kaynak kısıtlı proje çizelgeleme problemleri NP-Zor türü problemler olduğu için arama uzayı aktivite sayısının artması ile üstel biçimde arttığı için büyük problemlerde hesaplama süresi çok artmaktadır. Bu nedenle bu çalışmada küçük ölçekli problemler çözülmüştür. Fakat ileri çalışma olarak yöntemin hızlandırılması, uygun olmayan çözümlerin elenmesi ve paralel hesaplama ile daha hızlı çözümün elde edilmesi amaçlanmaktadır. Böylece daha büyük problemlerin çözümü mümkün olacaktır. Gerçekleştirilen 6 vaka analizinin tam sonuçları 1 dakikadan daha kısa sürede elde edilmesi ve yöntemin uygulanabilmesi için karmaşık optimizasyon yöntemlerinin bilinmesine gerek duyulmaması inşaat sektöründe önerilen algoritmanın uygulanabileceği gösterilmiştir.

Anahtar Kelimeler

Optimizasyon, Kaynak kısıtlı proje çizelgeleme problemi, proje yönetimi.

*Sorumlu Yazar: Önder Halis Bettemir, onder.bettemir@inonu.edu.tr, [ID 0000-0002-5692-7708](https://orcid.org/0000-0002-5692-7708)

1. GİRİŞ

Kaynak kısıtlı proje çizelgeleme problemleri (KKPÇP), kaynaklar üzerindeki kısıtları ihlal etmeden bir projeyi oluşturan tüm faaliyetlerin, öncel ilişkilerini dikkate alarak proje süresi en kısa olacak şekilde çizelgelenmesini amaçlar [1]. Bu sayede temin edilebilen kaynak miktarının sınırlı olduğu durumlarda mümkün olan en kısa sürede projenin tamamlanması sağlanır. Kalabalık şantiye kurmadan uygun sürede projenin tamamlanmasını ve iş makinesi yatırımlarında önemli tasarrufların elde edilmesini sağladığı için KKPÇP inşaat sektörü açısından oldukça önemlidir. Kaynak kısıtlı iş programının oluşturulması üretim süreci, yapının inşasının veya üretim gruplarının programlanması ile ilgilidir. Ayrıca yalın işletme kavramını uygulayabilmek için kapasite azalma yoluna giden sipariş usulü çalışan şirketler açısından da artan öneme sahiptir.

KKPÇP’de $V=\{0,1,\dots,n,n+1\}$ kümesinden oluşan bir projenin çözüleceği varsayılmaktadır. V kümesindeki $1,\dots,n$ ’e kadar olan aktiviteler gerçek, $j=0$, ($j=n+1$) başlangıç ve bitiş aktiviteleri kurgusaldır. Aktivitelerin yarıda kesilip daha sonra tekrar devam etme izni yoktur. Bu problem türünde yenilenebilir kaynaklar sınırlıdır ve tüm veriler tamsayıdır. KKPÇP’de amaç öncelik ilişkilerinin getirdiği kısıtlamalar ve sınırlı kaynak temin edilebilirliğini ihlal etmeden en kısa süreli iş programını bulmaktır.

En kısa proje süresi izin verilen $T_{üst}$ sınırını aşmadan öncelik ilişkileri kullanılarak $[E, G]$ aralığında erken tamamlanma zamanı E ve geç tamamlanma zamanı G ile ileri ve geri tekrarlanan $j \in V$ aktivitelerinin, önce uygulanan tamamlanma süresi devam eder. Benzer olarak $[E, G]$ aralığı E ilk başlangıç zamanı ve G son başlangıç zamanından itibaren sırasıyla aşağıdan ve yukarıdan bağlanır, öncelikle uygulanabilir başlangıç zamanlarını belirtmek için hesaplanabilir.

KKPÇP proje yönetimi uygulanan inşaat sektörü için oldukça önemli bir problemdir. Her inşaat projesi özgün olduğu için kendine özgü iş programı oluşturulacaktır. İlk oluşturulan iş programı kaynakların sonsuz sayıda temin edilebildiği varsayımı ile hazırlanır. Fakat ilk hazırlanan iş programının temin edilebilen kaynak miktarından daha fazla kaynak ihtiyacının olması muhtemeldir. Bu durumda mevcut iş programının hedeflenen sürede gerçekleştirilebilmesi mümkün olmayacak ve inşaatın gecikmesi sonucunda müteahhit taahhüdünü yerine getiremeyecek ve gecikme cezaları ile karşılaşacaktır. Belirtilen maddi yük nedeniyle imalat, hizmet ve endüstri sektörleri için KKPÇP’nin optimum çözümü çok önemlidir. Problemin öneminden dolayı literatürde de KKPÇP probleminin çözümü üzerine çok sayıda çalışma bulunmaktadır.

Zhang vd. parçacık sürüsü optimizasyonu tabanlı algoritmalarla KKPÇP çözümünü gerçekleştirmiştir [2]. Problemin ele alınışında öncelik tabanlı ve permütasyon tabanlı gösterim olmak üzere iki parçacık gösterimini göz önünde bulundurularak, kaynak kısıtlı proje çizelgeleme problemleri için parçacık sürüsü optimizasyonu kalıpları geliştirilmiştir. 480 adet 30 aktiviteli projelerden oluşan deneme problemlerinin çözüm sonucuna göre kaynak kısıtlı proje çizelgeleme problemi çözümünde permütasyon-tabanlı PSO, öncelik-tabanlı PSO'dan daha üstün performans göstermiştir ve PSO-tabanlı yöntem diğer üst sezgisel, genetik algoritma, tavlama benzetimi gibi yöntemlere göre daha kısa proje süreleri sunmuştur.

Kanıt vd. kaynak kısıtlarının yapı maliyetine etkisini ele alınmıştır. Yapım süreci boyunca kullanılması gereken kaynaklarda kısıt bulunması halinde, kısıtların proje süresi ve maliyetine etkisini araştırmıştır [3]. Talbot tarafından önerilen kapalı numaralandırma yöntemi daha az bilgisayar belleği kullanımını gerektirmektedir [4]. Stinson tarafından önerilen dal-sınır çözüm yöntemi, bilgisayar belleğinin sınırlamadığı durumlarda minimum hesaplama süresinde çözümler üretmekte ve tercih edilen çözüm yaklaşımı olması muhtemel görünmektedir [5]. Davis ve Khumawala tarafından geliştirilen sınırlı numaralandırma yönteminde minimum hesaplama süresi ile en uygun çözümün elde edilme olasılığı arttırılmıştır. Her bir problem için en iyi beş bağımsız değişken regresyon modelinin kullanıldığı ve her problemi çözmek için gereken hesaplama süresini tahmin etmek için bir ek deney gerçekleştirilmiştir. Elde edilen regresyon sonuçlarına dayanarak, minimum hesaplama süresini veren yöntem 110 test probleminin 98'inde veya bunların yaklaşık yüzde 89'unda doğru olarak tahmin edilmiştir [6].

Kolish seri ve paralel kaynak kısıtlı proje programlama yöntemlerinin, kaynak kısıtlı proje programlama problemlerinde nasıl bir etki oluşturduğu incelenmiştir. Sistematiik olarak oluşturulmuş 360 örneğe dayanan derinlemesine bir hesaplama çalışması yapılmıştır [7]. Merkle ve Middendorf kaynak kısıtlı proje çizelgeleme probleminin (KKPÇP) çözümünü karınca kolonisi optimizasyonu (KKO) ile gerçekleştirmiştir. KKO, 396 örnek problemin 130 en iyi çözümünü bulabilmiştir. İncelenen problemlerin yaklaşık 2/3'ü optimal çözülemese de genetik algoritma, tavlama benzetimi, tabu araması gibi algoritmalara kıyasla daha iyi bir performans gösterdiği sonucuna varılmıştır [8].

Hartmann, KKPÇP'yi genetik algoritma (GA) ile çözmüştür. Çalışmada ProGen tarafından üretilen 30 ve 60 aktiviteli projeler kullanılmıştır. Permütasyona dayalı GA yaklaşımı 30 aktiviteli projelerde en iyi sonucu vermiştir [9]. Patterson birden fazla kısıtlı kaynaklı iş programlama probleminin çözümünü incelenmiştir [10]. Özdamar KKPÇP'yi belirli bir amaç ve kısıtlara göre sınıflandırıp, zaman-maliyet tabanlı amaçlar ile yenilenebilir-yenilenemez

kaynak kısıtlarını ele almıştır [11]. Christofides 40 ile 25 aktiviteli problemleri 3 kaynak türü için [12], Olaguibel ve Goerlich ise; 38, 51 ve 103 aktiviteli problemleri 6 kaynak türü için çözmüşlerdir [13]. Özdamar ve Ulusoy geniş kapsamlı dallanma şemasını kullanarak sezgisel budama yöntemiyle optimum sonuçtan ortalama %2 sapma ile çözüm elde etmiştir [14].

Peteghem ve Vanhoucke çok modlu kaynak kısıtlı proje programlama problemini (ÇMKKPPP) çözmek için literatürde bulunan meta sezgisel çözüm yöntemlerine genel bir bakış sağlayıp bu yöntemler arasında eşit kriterlere dayalı karşılaştırmalar yapmıştır [15]. Bukata vd. kaynak kısıtlı proje programını (KKPPP) makul bir sürede nitelikli çözümler bulmak için paralel tabu arama algoritmasını (PTAA) ele alınmıştır [16]. Tabu arama (TA) meta-sezgisel yöntem ilk olarak 1986 yılında Glover tarafından önerilmiştir [17]. TA üst-sezgisel yöntemi, TA'nin temel terimlerinin bir tabu listesi, aspirasyon kriterleri, çeşitlendirme ve yoğunlaşma olarak tanımlanmıştır.

Bhaskar bulanık aktivite süresiyle KKÇP'yi ele almıştır [18]. Problemi çözmek için tekrarlamalı olmayan performans programlama dizini (PPD) sezgisel yöntemini kullanmıştır. Beşikçi vd. çok modlu kaynak kısıtlı proje programlama problemini çözmüştür [19-20].

Arauzo çoklu faktör sistemi (ÇFS) ile dinamik olarak kaynak tahsis ederek KKPPP çözmüştür [21]. Wang belirsiz faaliyet süreleriyle kaynak kısıtlı çoklu proje programlama problemi (KKÇPPP) için öncelik kurallarının performansı ele alınmıştır. Minimum gecikme, maksimum gecikme, minimum gecikme süresi dâhil olmak üzere 17 hibrit kural uygulamıştır [22]. Brucker makine programlamasında yaygın olarak kabul edilen sınıflandırma ile uyumlu bir sınıflandırma şeması oluşturamamıştır [23]. Zhu vd. ayırık zıt çoklu versiyon optimizasyon algoritması ile çok kaynaklı KKPPP'yi çözmüştür [24]. Lin vd. genetik programlamalı hiper sezgisel yöntemle çok kaynaklı KKÇP'yi çözmüştür [25]. Pellerin vd. KKÇP çözümü için uygulanan hibrit üst sezgisel yöntemleri incelemiş ve öne çıkan yöntemlerin ortak özelliklerini sıralamıştır [26]. Birjandi ve Mousavi stokastik KKÇP'yi sezgisel yöntemle çözmüştür [27]. Çok kaynaklı KKÇP'yi Laszczyk ve Myszkowski çok amaç fonksiyonlu evrimsel gelişim algoritması ile çözmüştür [28]. Chakraborty vd. çok modlu KKÇP'yi modifiye edilmiş değişken komşulu arama sezgisel yöntemi ile [29], Tirkolae vd. çok amaç fonksiyonlu çok modlu KKÇP'yi Pareto tabanlı algoritmalarla çözmüştür [30]. Creemers stokastik KKÇP'yi aktiviteleri istenildiği zaman yarıda kesip bir süre geçtikten sonra devam ettirecek şekilde incelemiştir [31]. Bettemir ve Sönmez (2015) KKÇP'yi tavlama benzetimli genetik algoritma ile çözmüştür [32].

Literatür taramasında değinilen dal-sınır yöntemlerinin modifikasyonları optimizasyon problemlerinin çözüm başarısını önemli ölçüde geliştirmiştir. Geçerli eşitsizliklerin üretimi ve bu eşitsizliklerin bir doğrusal programlama çözücüsü aracılığıyla ilerlemesi devamlı testlere dayalı kısıtların ilerlemesinin etkili bir başlangıcı için düşünülebilir. Programlamadaki başarılı uygulamalar programlama problemlerini ayırmak ve mevcut durumdaki temel kaynak-kısıtlı proje programlama problemini de kapsamaktadır. Böylece uygulanamayan düğümlerin sezgisel ve dal-sınır yöntemlerini erken tespit yoluyla hızlandırmaktır.

Literatür taramasında belirtilen algoritmalar problemin başarılı biçimde çözülmesini sağlamaktadır. Fakat inşaat sektöründe faaliyet gösteren kurumsallaşamamış küçük ve orta ölçekli yükleniciler göz önüne alındığında ve inşaat mühendisliği eğitim müfredatında optimizasyon algoritmalarına yer verilmediği göz önüne alındığında belirtilen yüklenici grubunun KKPCP'yi optimizasyon algoritmaları kullanarak çözebilmesi uygulanabilir değildir. KKPCP'yi çözen paket yazılımlar bulunmakla birlikte paket yazılımlar hem ilave maliyet olmakta hem de optimizasyon yazılımlarının uyguladığı doğrusal programlama ve dal-sınır algoritmalarına uygun veri girişini yapabilecek inşaat mühendisi sayısı yetersizdir. Bu nedenle KKPCP'nin çözümü küçük ve orta ölçekli kurumsallaşamamış yükleniciler tarafından ihmal edilmektedir. Bu çalışmada optimizasyon yöntemi uygulamadan tek modlu kısıtlı kaynaklı iş programı probleminin çözümü incelenmiştir. KKPCP'nin arama uzayının belirlenip tamamının taranması ile garantili biçimde optimizasyon yöntemi uygulanmadan en kısa proje süresi elde edilebilecektir.

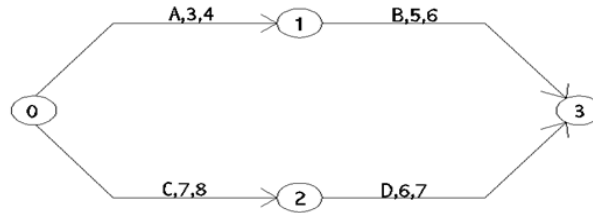
2. YÖNTEM

Tüm arama uzayının taranabilmesi için KKPCP'nin arama uzayının belirlenmesi gerekmektedir. Bu süreç şebekedeki kritik olan ve olmayan tüm aktivitelerin ardıl-öncel ilişkilerin ihlal edilmeden kaç farklı biçimde öncelik atanması yapılabileceğinin hesaplanmasını gerektirmektedir. Arama uzayının belirlenmesinin birinci adımı şebekede kaç adet yol olduğunun belirlenmesidir. Şebekedeki yollar belirlendikten sonra iki adet yol biçimde eşleştirilerek kaç farklı öncelik tayin edilerek tek yol oluşturulabileceği tespit edilir. Elde edilen birleştirilmiş yol, daha önce incelenmemiş yolla eşleştirilerek tekrar kaç farklı öncelik tayin edilmiş biçimde tek yol oluşturulabildiği hesaplanır. Bu şekilde şebekedeki tüm yollar incelenene kadar birleştirme işlemi sürdürülür. Aktivitelerin kaç farklı biçimde dizilebileceği permütasyon ile hesaplanır.

Hesap tablosu uygulamasının oluşturulması ve problemin tanıtılması örnek problem üzerinde açıklanmıştır. Bu sayede inşaat sektöründe faaliyet gösteren tüm firmalar önerilen yöntemi uygulayabileceklerdir.

2.1 Örnek Problem 1

KKPÇP'nin çözümü için hazırlanan hesap cetveli uygulamasının oluşturulması dört aktiviteli bir problem üzerinden anlatılmıştır. Şekil 1'de ok diyagramı verilen projede 4 aktivite yer almakta ve aktiviteler A, B, C ve D olarak adlandırılmıştır. Şebekeyi oluşturan ve toplam dört tane olan düğüm noktaları 0, 1, 2 ve 3 olarak adlandırılmıştır. Aktivite süreleri ve aktivitelerin ihtiyaç duyduğu kaynak miktarı aktiviteyi temsil eden okların üzerindeki etiketlerde belirtilmiştir.



Şekil 1. Örnek proje-1.

Proje A-B ve C-D aktivitelerinin oluşturduğu 2 hattan oluşmaktadır. Belirtilen hatların permütasyonu sonucu Tablo 1'de gösterilen öncelik sıralamaları elde edilmektedir.

Tablo 1. Problem 1'in öncelik sıralama alternatifleri

Alternatif	Öncelik Sıralaması
A1	A-B-C-D
A2	A-C-B-D
A3	A-C-D-B
A4	C-A-B-D
A5	C-A-D-B
A6	C-D-A-B

Projenin analiz edilebilmesi için problemin hesap tablosu uygulamasına tanıtılması gerekmektedir. Aktivite isimleri Şekil 2'de gösterildiği gibi uygulamanın D sütununda girilmektedir. Her bir aktivitenin başlangıç ve bitiş düğüm noktaları sırası ile B ve C sütunlarında tanımlanmıştır. Aktivite süreleri E sütununa, aktivitelerin kaynak ihtiyaçları ise F sütununda her aktivite için ilgili satırlara Şekil 2'de gösterildiği gibi tanımlanmıştır. Bu kısım projede yer alan aktivitelerle ilgili veri girişinin yapıldığı sütunlardır.

Bu aşamada projeyi oluşturan aktivitelerin hangi başlangıç ve bitiş düğüm noktaları arasında olduğu belirtilmiştir. Örneğin A aktivitesinin başlangıç düğüm noktası 0 ve bitiş düğüm

noktasının 1 olduğu sırası ile B3 ve C3 düğüm noktalarına girilen 0 ve 1 değerleri ile tanımlanmıştır. Diğer aktivitelerin başlangıç ve bitiş düğüm noktaları aynı yöntemle tanımlanmıştır.

Aktivite süreleri ve Şekil 1'de sunulan şebeke diyagramı kullanılarak tüm düğüm noktalarının ileri gidişle Erken Olay Zamanı (EOZ) ve Geç Olay Zamanı (GOZ) hesaplanmıştır. Ok diyagramı şebekelerinin çözümünde ilk önce ileri gidişle EOZ hesaplanır. İlk düğüm noktası olan sıfırıncı düğüm noktasının EOZ'si 0'dır. Bir sonraki olay zamanı olan 1. düğüm noktası için EOZ hesaplaması sıfırıncı düğüm noktasının erken olay zamanı olan EOZ₀'a A aktivitesinin süresi eklenerek gerçekleştirilir. Belirtilen işlemin hesap tablosu üzerinde tanımlanması Şekil 3'te gösterilmektedir.

	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Baş	Bit	Aktivite	Süre	Kaynak	EOZ _i	GOZ _i	EOZ _j	GOZ _j	TB	Baş	Bit			Mak işçi	12
0	1	A	3	4	0	0	3	8	5	0	3				
1	3	B	5	6	3	8	13	13	5	3	8				
0	2	C	7	8	0	0	7	7	0	0	7				
2	3	D	6	7	7	7	13	13	0	7	13				
DN	EOZ	GOZ													
0	0	0													
1	3	8													
2	7	7													
3	13	13													

Şekil 2. Veri giriş ekranı.

TOPLA																
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1																
2		Baş	Bit	Aktivite	Süre	Kaynak	EOZ _i	GOZ _i	EOZ _j	GOZ _j	TB	Baş	Bit		Mak işçi	12
3		0	1	A	3	4	0	0	3	8	5	0	3			
4		1	3	B	5	6	3	8	13	13	5	3	8			
5		0	2	C	7	8	0	0	7	7	0	0	7			
6		2	3	D	6	7	7	7	13	13	0	7	13			
7																
8		DN	EOZ	GOZ												
9		0	0	0												
10		1	=C9+E3	8												
11		2	7	7												
12		3	13	13												

Şekil 3. Erken ve Geç Olay Zamanları hesaplamasının hesap tablosunda tanımlanması.

2 numaralı düğüm noktasına sadece 0 numaralı düğüm noktasından geliş vardır ve 0-2 hattında C aktivitesi yer almaktadır. Bu durumda $EOZ_2 = EOZ_0 + SÜRE_C$ biçiminde formüleleştirilebilir. Son düğüm noktası olan 3 numaralı düğüm noktasına hem 1 numaralı hem de 2 numaralı düğüm noktalarından gelinebilmektedir. EOZ hesaplanırken tüm gelişlerin en uzun süreni alınır. Böylece $EOZ_3 = \text{MAKSİMUM}(EOZ_1 + SÜRE_B; EOZ_2 + SÜRE_D)$ formülü ile ifade edilir. Son düğüm noktasının hesaplanması ile ileri gidiş tamamlanmış ve tüm düğüm noktalarının EOZ'leri hesaplanmıştır.

Bu aşamadan sonra düğüm noktalarının GOZ'leri hesaplanmıştır. Son düğüm noktası olan üçüncü düğüm noktasının GOZ'si için $GOZ_3 = EOZ_3$ eşitliği tanımlanır. Bir önceki olay zamanı olan 2. düğüm noktasının GOZ hesabı üçüncü düğüm noktasından D aktivite süresinin çıkarılmasıyla $GOZ_2 = GOZ_3 - SÜRE_D$ olarak hesaplanmıştır. Birinci düğüm noktasının GOZ

hesaplaması 3. düğüm noktasının GOZ değerinden B aktivite süresinin çıkarılmasıyla bulunmuştur ($GOZ_1 = GOZ_3 - SÜRE_B$). Sıfırncı düğüm noktasına iki dönüş vardır ve GOZ hesaplanırken tüm dönüşlerin en kısa süreni alınmaktadır. Sıfırncı düğüm noktası $GOZ_0 = \text{MİNİMUM}(GOZ_2 - SÜRE_C; GOZ_1 - SÜRE_B)$ olarak hesaplanmıştır.

TOPLA													
=DÜŞEYARA(B3:\$B\$9:\$D\$12;2;YANLIŞ)													
	A	B	C	D	DÜŞEYARA(aranan_değer; tablo_dizisi; sütun_indis_sayısı; [aralık_bak])				K	L	M		
1													
2		Baş	Bit	Aktivite	Süre	Kaynak	EOZ _i	GOZ _i	EOZ _j	GOZ _j	TB	Baş	Bit
3		0		1 A	3	4	=DÜŞEYARA	0	3	8	5	0	3
4		1		3 B	5	6	3	8	13	13	5	3	8
5		0		2 C	7	8	0	0	7	7	0	0	7
6		2		3 D	6	7	7	7	13	13	0	7	13
7													
8		DN	EOZ	GOZ									
9		0	0	0									
10		1	3	8									
11		2	7	7									
12		3	13	13									

Şekil 4. Aktivitelerin Erken ve Geç Başlangıç Zamanlarının hesaplanması

Olay zamanlarının gösteriminde i indisi aktivitenin başlangıç düğüm noktası, j indisi ise aktivitenin bitiş düğüm noktasını ifade etmek için kullanılmıştır. Aktivitelerin EOZ ve GOZ hesaplandıktan sonra aktivitelerin I, J, K ve L sütunlarındaki EOZ_i ve GOZ_j verileri Şekil 4'te formülü açık şekilde belirtilen düşeyara komutu ile hesaplanmıştır. Düşeyara tablonun en sol sütunundaki değeri arar ve daha sonra aynı satırda belirtilen sütundan bir değer vermektedir. Şekil 4'te A aktivitesinin EOZ_i değerini veren ifade gösterilmektedir. Formülde belirtilen B3 satırı aranan değerdir, yani başlangıç düğüm noktasıdır. Tablonun alt satırında girilen düğüm noktası (DN), EOZ ve GOZ sütunları düşey ara formülünde girilen tablo dizisini tanımlamaktadır. Formülde 2 verisi ikinci sütun olan EOZ sütununu belirtmektedir. Formülde belirtilen YANLIŞ verisi tam eşleşme istendiği için girilmiştir.

Formül tüm aktiviteler için tanımlanmış ve EOZ_i, GOZ_i, EOZ_j ve GOZ_j değerleri sırası ile I, J, K ve L sütunlarında gösterilmiştir. Aktivitelerin başlangıç zamanı EOZ_i değerine eşit kabul edilerek erken başlangıç zamanları belirlenmiştir. Başlangıç zamanlarına aktivite süreleri eklenerek bitiş süreleri hesaplanmıştır. M sütununda TB olarak belirtilen toplam bolluk süreleri $GOZ_j - EOZ_i - \text{Aktivite Süresi}$ ifadesi ile hesaplanmıştır. Şekil 5'de A aktivitesinin toplam bolluğunun hesaplanması gösterilmektedir. Formülde L3 hücresi A aktivitesinin GOZ_j 'si, I3 hücresi EOZ_i ve son olarak E3 hücresi ise A aktivitesinin süresidir. Aynı formül tüm aktiviteler için uygulanarak B, C ve D aktivitelerinin de toplam bolluğu bulunmuştur.

N ve O sütununda belirtilen Başlangıç ve Bitiş zamanları Şekil 6 ve 7'de belirtildiği gibi hesaplanmıştır. A aktivitesinin başlangıç zamanı, F sütununda Gecikme olarak tanımlanan A aktivitesinin gecikmesi süresi ile EOZ_i değerini toplanması ile hesaplanmıştır. Belirtilen ifade

$BAŞLANGIÇ_A = EOZ_{iA} + GECİKME_A$ ile formüleleştirilebilir. Aynı işlem diğer aktiviteler içinde yapılarak tüm aktivitelerin başlangıç zamanı hesaplaması yapılmıştır.

O sütununda Bitiş olarak tanımlanan bitiş zamanları Şekil 7’de belirtildiği gibi aktivitenin başlangıç zamanına aktivite süresinin eklenmesi ile hesaplanmıştır. Aynı işlem B, C ve D aktiviteleri içinde uygulanarak tüm aktivitelerin bitiş süreleri bulunmuştur.

TOPLA													
A	B	C	D	E	F	G	H	I	J	K	L	M	
1													
2	Baş	Bit	Aktivite	Süre	Kaynak	EOZ _i	GOZ _i	EOZ _j	GOZ _j	TB	Baş	Bit	
3	0	1	A	3	4	0	0	0	3	8	0	3	
4	1	3	B	5	6	3	8	13	13	5	3	8	
5	0	2	C	7	8	0	0	7	7	0	0	7	
6	2	3	D	6	7	7	7	13	13	0	7	13	
7													
8	DN	EOZ	GOZ										
9	0	0	0										
10	1	3	8										
11	2	7	7										
12	3	13	13										

Şekil 5. Toplam bolluk sürelerinin hesaplanması.

TOPLA														
B	C	D	E	F	G	H	I	J	K	L	M	N	O	
Baş	Bit	Aktivite	Süre	gecikme	yalın süre	Kaynak	EOZ _i	GOZ _i	EOZ _j	GOZ _j	TB	Başlangıç	Bitiş	
0	1	A	3	0	3	4	0	0	3	8	5	0	3	
1	3	B	5	0	5	6	3	8	13	13	5	3	8	
0	2	C	7	0	7	8	0	0	7	7	0	0	7	
2	3	D	6	0	6	7	7	7	13	13	0	7	13	

Şekil 6. Başlangıç sürelerinin hesaplanması.

TOPLA														
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1														
2	Baş	Bit	Aktivite	Süre	gecikme	yalın süre	Kaynak	EOZ _i	GOZ _i	EOZ _j	GOZ _j	TB	Başlangıç	Bitiş
3	0	1	A	3	0	3	4	0	0	3	8	5	0	3
4	1	3	B	5	0	5	6	3	8	13	13	5	3	8
5	0	2	C	7	0	7	8	0	0	7	7	0	0	7
6	2	3	D	6	0	6	7	7	7	13	13	0	7	13
7														
8	DN	EOZ	GOZ											
9	0	0	0											
10	1	3	8											
11	2	7	7											
12	3	13	13											

Şekil 7. Bitiş sürelerinin hesaplanması.

TOPLA																					
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Baş	Bit	Aktivite	Süre	gecikme	yalın süre	Kaynak	EOZ _i	GOZ _i	EOZ _j	GOZ _j	TB	Başlangıç	Bitiş	Mak İşçi	12	aktivite	öncelik				
0	1	A	3	0	3	4	0	0	3	8	5	0	3			A	1	A			
1	3	B	5	0	5	6	3	8	13	13	5	3	8			B	2	B			
0	2	C	7	0	7	8	0	0	7	7	0	0	7			C	3	C			
2	3	D	6	0	6	7	7	7	13	13	0	7	13			D	4	D			
DN	EOZ	GOZ																			
0	0	0																			
1	3	8																			
2	7	7																			
3	13	13																			
Aktivite/Süre	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
A	3	4	4																	12	
B	5	6	6	6	6	6	6													30	
C	7	8	8	8	8	8	8													58	
D	6	7	7	7	7	7	7	7	7	7	7										

Şekil 8. Aktivite/Süre hesaplanması ve Gantt şeması oluşturulması.

Şekil 8’de Gantt şemasının hazırlanışı gösterilmektedir. C14 hücresinden başlayarak sağ taraftaki diğer hücelere iş gününü temsil eden rakamlar girilmiştir. Her satır bir aktiviteye ait

olacak şekilde aktivitenin hangi iş günlerinde yürütüldüğünü belirleyen formül tanımlanmıştır. Şekil 8’de tanımlanan EĞER(VE.....) formülü iş gününün aktivitenin başlangıç zamanından büyük, tamamlanma zamanına küçük eşit olması durumunda ilgili hücreye aktivitenin kaynak talebini, aksi durumda ise 0 değerini yazdırmaktadır. Koşullu biçimlendirme ile sıfırdan büyük değere sahip hücrelere yeşil renk verilerek Şekil 8’de gösterilen Gantt Şema görüntüsü elde edilmiştir.

Gantt şemasının oluşturulmasından sonra günlük kaynak talebi Şekil 9’da 20. satırda gösterildiği gibi hesaplanmıştır. Temin edilebilen kaynak miktarının aşılması halinde o gün yürütülen aktiviteleri belirleyen formül Şekil 10’da belirtildiği gibi 23 ve 26. Satırlardaki tüm hücrelere tanıtılır. Şekil 9’da 4, 5, 6 ve 7. Günlerde B ve C aktivitelerinin 8. günde ise B ve D aktivitelerinin yürütülmesi nedeniyle kaynak çakışmasına yol açmıştır.

C23		=EĞER(VE(C\$20>\$R\$2:C15>0);\$B23;"")																			
	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	Baş	Bit	Aktivite	Süre	gecikme	yalın süre	Kaynak	EOZİ	GOZİ	EOZJ	GOZJ	TB	Başlangıç	Bitiş		Mak İşçi	12		aktivite	öncelik	
2	0	1 A	3	0	3	4	0	0	0	3	8	5	0	3					A	1 A	
3	1	3 B	5	0	5	6	3	8	13	13	13	5	3	8					B	2 B	
4	0	2 C	7	0	7	8	0	0	7	7	0	0	0	7					C	3 C	
5	2	3 D	6	0	6	7	7	7	7	13	13	0	7	13					D	4 D	
6																					
7																					
8	DN	EOZ	GOZ																		
9	0	0	0																		
10	1	3	8																		
11	2	7	7																		
12	3	13	13																		
13																					
14	Aktivite/Süre	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
15	A	4	4	4																	
16	B				6	6	6	6	6	8											
17	C	8	8	8	8	8	8	8	8												
18	D								7	7	7	7	7	7							
19																					
20		12	12	12	14	14	14	14	13	7	7	7	7	7							
21																					
22	çakışan akt	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
23	A																				
24	B				B	B	B	B	B												
25	C				C	C	C	C	C												
26	D								D												

Şekil 9. Çakışan Aktivitelerin belirlenmesi.

TOPLA		=EĞER(C23<>"";DÜŞEYARA(C23;T\$3:S\$6;2;YANLIŞ;""))																			
	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
2	Baş	Bit	Aktivite	Süre	gecikme	yalın süre	Kaynak	EOZİ	GOZİ	EOZJ	GOZJ	TB	Başlangıç	Bitiş		Mak İşçi	12		aktivite	öncelik	
3	0	1 A	3	0	3	4	0	0	0	3	8	5	0	3					A	1 A	
4	1	3 B	5	0	5	6	3	8	13	13	13	5	3	8					B	2 B	
5	0	2 C	7	0	7	8	0	0	7	7	0	0	0	7					C	3 C	
6	2	3 D	6	0	6	7	7	7	7	13	13	0	7	13					D	4 D	
7																					
8	DN	EOZ	GOZ																		
9	0	0	0																		
10	1	3	8																		
11	2	7	7																		
12	3	13	13																		
13																					
14	Aktivite/Süre	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
15	A	4	4	4																	
16	B				6	6	6	6	6	8											
17	C	8	8	8	8	8	8	8	8												
18	D								7	7	7	7	7	7							
19																					
20		12	12	12	14	14	14	14	13	7	7	7	7	7							
21																					
22	çakışan akt	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
23	A																				
24	B				B	B	B	B	B												
25	C				C	C	C	C	C												
26	D								D												
27																					
28	Ç. önceliği																				
29	A																				
30	B				2	2	2	2	2												
31	C				3	3	3	3	3												
32	D								4												

Şekil 10. Çakışma durumunda ertelenecek aktivitenin tespit edilmesi.

Şekil 10’da kaynak çakışması görülen aktivitelerden hangisine öncelik verileceği gösterilmiştir. Formül, hücrenin boş olmaması halinde hangi aktivite adının olduğunu bulup ilgili aktiviteye hangi önceliğin atandığını tabloda arama yaparak bulmakta ve öncelik değerini ilgili hücreye yazmaktadır. Kaynak önceliklerinin T sütununa aktivite isimlerinin U sütununa kullanıcı

tarafından girilmesi gerekmektedir. Örneğin A aktivitesinin önceliği 1 rakamı ile birinci öncelik olarak B aktivitesinin önceliği ise 2 olarak numaralandırıldığı durum için 2 aktivite arasında kaynak çakışmasının olması halinde B aktivitesinin önceliği daha düşük olduğu için B aktivitesi ertelenecektir. Hesapların sonucu aşağıdaki hesap cetvelinde gösterildiği gibi 4, 5, 6 ve 7. günlerde çakışan B ve C aktivitelerinin öncelik sırası, 8. günde de B ve D aktivitelerinin öncelik sırası 29 ve 32. satırlarda verilmiştir.

Şekil 10’da gösterildiği üzere kaynak aşımı 4. ve 8. iş günleri dâhil olacak şekilde 5 günlük zaman dilimi içindedir ve bunun ilk 4 gününde C Aktivitesi en düşük önceliğe sahiptir. Bu nedenle ilk denemede C aktivitesi 4 gün geciktirilmiş ve oluşan iş programı Şekil 11’de sunulmuştur. Ertelenecek aktivite için E sütununda aktivitenin bulunduğu ilgili satıra gecikme süresi girilir. D sütununda aktivite süresi $Süre = Süre' + Gecikme$ olarak tanımlanmıştır. Burada Süre’ olarak tanımlanan aktivitenin kaynak kullanımını içeren süresi, Gecikme aktivitenin geciktirileceği süre ve Süre ise aktivitenin toplam süresini ifade etmektedir. Aktivitenin kaynak kullanımına ne zaman başlayıp ne zaman tamamlanacağını belirten süreler M ve N sütunda Başlangıç ve Bitiş başlıkları altında hesaplanmıştır. E4 hüccesine 4 rakamı girilerek C aktivitesi 4 gün geciktirilmiştir. C aktivitesi 4 gün geciktirilince M ve N sütunundaki Başlangıç ve Bitiş süreleri de değişmektedir. C aktivitesinin geciktirilmemiş hali olan $GECİKME_C = 0$ durumunda başlama ve tamamlanma zamanı sırası ile 0 ile 7. günler olmaktadır. C aktivitesi 4 gün geciktirilince aktivitenin başlama ve tamamlanma zamanı sırası ile 4 ile 11. günler olmaktadır. C aktivitesinin bolluğu olmadığı için ertelendikçe projenin tamamlanma süresi de uzamaktadır. Mevcut durumda projenin tamamlanma süresi 13 günden 17 güne uzamış ve kaynak kısıtlarının ihlali önlenemediği için aktivitelerin geciktirme sürecine devam edilecektir.

1	Baş	Bit	Aktivite	Süre	Gecikme	Yalın Süre	Kaynak	EOZ1	GOZ1	EOZ2	GOZ2	TB	Başlangıç	Bitiş	Mak İyri	12	Aktivite	Öncelik										
2	0	1	A	3	0	3	4	0	0	0	0	12	9	0	3		A	1 A										
3	1	3	B	5	0	5	6	3	12	17	17	9	3	8			B	2 B										
4	0	2	C	11	4	7	8	0	0	11	11	0	4	11			C	3 C										
5	2	3	D	5	0	5	6	7	11	11	17	17	0	11	17		D	4 D										
7	DN	EOZ	GOZ				12	12																				
8	0	0	0				39	30																				
9	1	3	12				56	56																				
10	2	11	11				42	42																				
11	3	17	17																									
12																												
13																												
14	Gün	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
15	A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
16	B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
17	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
18	D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
19		4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	
20																												
21	Çakışan Aktivite	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
22	A																											
23	B																											
24	C																											
25	D																											
26	Çakışma Önceliği																											
27	A																											
28	B																											
29	C																											
30	D																											
31	Ertelenme	0	0	0	0	0	3	3	3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
32	Aktivite																											

Şekil 11. C aktivitesinin 4 gün ertelenmesi sonucunda oluşan iş programı.

Şekil 11’de sunulan iş programında kaynak kısıtlarının aşılmadığı dönemin 4 gün olduğu ve C aktivitesinin en düşük önceliğe sahip olduğu görülmektedir. Bu durumda ikinci denemede C aktivitesi 4 gün daha ertelenip C aktivitesinin ertelenme süresinin 4 günden 8 güne çıkarılması

gerekmektedir. C aktivitesinin 8 gün ertelenmesi ile elde edilen iş programı Şekil 12’de sunulmuştur.

Bit	Aktivite	Süre	Gecikme	Yalın Süre	Kaynak	EOZ ₁	GOZ ₁	EOZ ₂	GOZ ₂	TB	Başlangıç	Bitiş	Mak İşçi	12	Aktivite	Öncelik									
1	A	3	0	3	4	0	0	3	16	13	0	3			A	1 A									
3	B	5	0	5	6	3	16	21	21	13	3	8			B	2 B									
2	C	15	8	7	8	0	0	15	15	0	8	15			C	3 C									
3	D	6	0	6	7	15	15	21	21	0	15	21			D	4 D									
EOZ		GOZ				12	12																		
0	0					30	30																		
3	15					55	55																		
15	15					42	42																		
21	21																								
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
4	5	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	5	5	5	5	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	4	4	6	6	6	6	6	8	8	8	8	8	8	8	7	7	7	7	7	7	7	7	7	7	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Şekil 12. A-B-C-D öncelik sıralaması için oluşan iş programı.

Örnek problem 1 için 6 farklı öncelik tayin etme seçeneği vardır. Birinci alternatif olan A-B-C-D öncelik sıralamasının çözümü Şekil 12’de belirtildiği gibi hesaplanmıştır. İkinci çözüm alternatifinde 4 aktivitenin öncelik sırası değiştirilmiştir. Tablo 1’de belirtildiği gibi bu alternatifte B aktivitesinin öncelik sırası C aktivitesinden sonra gelmektedir. Burada B aktivitesi ertelenecektir. Alternatifler arasında en kısa süren öncelik sırası aranan en iyi çözüm olacaktır. Şekil 13’te görüldüğü gibi 5 günlük periyot içerisinde kaynak aşımı vardır. İlk denemede B aktivitesi 5 gün ertelenmiştir. İlk deneme sonunda B aktivitesinin 5 gün geciktirilmiş olmasına rağmen kaynak kısıtlarının aşıldığı Şekil 13’te sunulan iş programında görülmektedir. Kaynak kısıtlarının aşılma süresi 5 gündür. İlk denemede B aktivitesi 5 gün ertelenmiştir. İlk deneme sonunda B aktivitesinin 5 gün geciktirilmiş olmasına rağmen kaynak kısıtlarının aşıldığı Şekil 13’te sunulan iş programında görülmektedir. Kaynak kısıtlarının aşılma süresi 5 gündür.

1	Baş	Bit	Aktivite	Süre	Gecikme	Yalın Süre	Kaynak	EOZ ₁	GOZ ₁	EOZ ₂	GOZ ₂	TB	Başlangıç	Bitiş	Mak İşçi	12	Aktivite	Öncelik										
2	0	1	A	3	0	3	4	0	0	3	16	13	0	3			A	1 A										
3	1	3	B	10	5	5	6	3	3	13	13	0	8	13			B	3 B										
4	0	2	C	7	0	7	8	0	0	7	7	0	0	7			C	2 C										
5	2	3	D	6	0	6	7	7	7	13	13	0	7	13			D	4 D										
7	DN	EOZ	GOZ					12	12																			
8	0	0	0					30	30																			
9	1	3	3					55	55																			
10	2	7	7					42	42																			
11	3	13	13																									
14	Gün	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
15	A	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
16	B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
17	C	5	5	5	5	5	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
18	D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
19	0	12	12	12	8	8	8	7	7	13	13	13	13	13	0	0	0	0	0	0	0	0	0	0	0	0		
21	Çakışan	Aktivite	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
22	A																											
23	B									B	B	B	B	B														
24	C																											
25	D									D	D	D	D	D														
26	Çakışma	Önceliği																										
27	A																											
28	B									3	3	3	3	3														
29	C																											
30	D									4	4	4	4	4														
31	Entelelce	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Şekil 13. A-B-C-D alternatifinin birinci erteleme sonucunda oluşan iş programı.

Bu durumda ikinci denemede B aktivitesi 5 gün daha ertelenmesi gerekmekte olduğu için B aktivitesinin ertelenme süresi 5 günden 10 güne çıkarılması gerekmektedir. B aktivitesinin toplam 10 gün ertelenmesi ile elde edilen iş programı Şekil 14’de sunulmuştur.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1	Baş	Bit	Aktivite	Süre	Gecikme	Yalın	Süre	Kaynak	EOZİ	GOZİ	EOZİ	GOZİ	TB	Başlangıç	Bitiş	Mak İleri	12	Aktivite	Öncelik								
2	0	1	A	3	0	3	4	0	0	3	3	0	0	3				A	1	A							
3	1	3	B	15	10	5	6	3	3	18	18	0	13	18				B	3	B							
4	0	2	C	7	0	7	8	0	0	7	12	5	0	7				C	2	C							
5	2	3	D	6	0	6	7	7	12	18	18	5	7	13				D	4	D							
6																											
7	DN	EOZ	GOZ			12	12																				
8	0	0	0			30	30																				
9	1	3	3			56	56																				
10	2	7	12			42	42																				
11	3	18	18																								
12																											
13																											
14	Gün	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
15	A	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	B	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	5	5	5	0	0	0	0	0	0	0	0
17	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19		12	12	12	8	8	8	8	7	7	7	7	7	7	7	7	7	7	7	6	6	6	6	6	6	6	6
20																											
21	Çalışan																										
22	Aktiviyele	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
23	A																										
24	B																										
25	C																										
26	D																										
27	Çalışma																										
28	Önceliği																										
29	A																										
30	B																										
31	C																										
32	D																										
33	Erteleme	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Şekil 14. A-C-B-D alternatifinin ikinci erteleme sonucunda oluşan iş programı.

Alternatif 2'nin tam çözümünün elde edilmesinin ardından üçüncü alternatife geçilmiştir. İlk aşamada aktivitelere öncelik sırası yeniden tayin edilmiştir. Kaynak aşımı 5 günlük dönem içerisinde olduğu için ve en düşük öncelik B aktivitesinde bulunduğu için ilk erteleme serisinde B aktivitesi 5 gün ertelenmiştir. Kaynak kısıtlarının aşımı hala devam ettiği için ikinci denemede B aktivitesi 5 gün daha ertelenerek B aktivitesinin ertelenme süresi 5 günden 10 güne çıkarılmıştır. B aktivitesinin 10 gün ertelenmesi ile elde edilen iş programı Şekil 15'te sunulmuştur.

13																										
14	Gün	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21				
15	A	4	4	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19		12	12	12	8	8	8	8	7	7	7	7	7	7	7	7	7	7	7	6	6	6	6	6	6	6
20																										

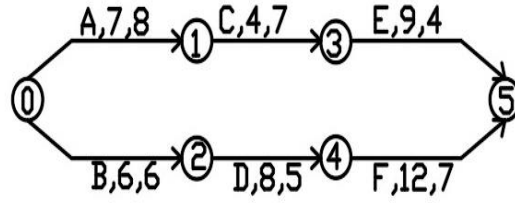
Şekil 15. A-C-D-B öncelik sıralamasının çözümü sonucunda oluşan iş programı.

Dördüncü öncelik sıralamasında ilk önce B aktivitesi 5 gün ertelenmiştir. Kaynak kısıtlarının aşımı hala devam ettiği için ikinci denemede B aktivitesi 5 gün daha ertelenerek ertelenme süresi 10 güne çıkarılmış ve bir önceki öncelik sıralaması ile aynı çözüm olan Şekil 15'de sunulan tam çözüm elde edilmiştir.

Beşinci ve altıncı aktivite öncelik alternatiflerinin çözümünde de B aktivitesi 2 aşamada toplam 10 gün ertelenmiştir. Tablo 1'de belirtilen 6 öncelik sıralamasının tamamı denenmiş ve kaynak kısıtları dikkate alınarak oluşturulan çözümde en kısa proje süresi 18 gün olarak hesaplanmıştır. Uygulanabilir tüm alternatifler denendiği için elde edilen çözüm kesin optimum çözümdür.

2.2 Örnek Problem 2

Şebeke diyagramı Şekil 16'da sunulan 6 aktiviteli, 5 düğüm noktalı şebekenin sonsuz kaynak tanımı ile gerçekleştirilen çözümü 26 gündür.



Şekil 16. Problem 2'nin şebeke diyagramı.

Şebeke 2 A-C-E ve B-D-F hatlarının oluşturduğu 2 hatta sahiptir. Şebekedeki aktivite sayısı 6 olduğu için aktiviteler arasındaki öncelik sıralaması ihmal edildiğinde toplamda 6! kadar öncelik sıralaması oluşturulabilir. A-C-E hattındaki aktiviteler ile B-D-F hattındaki aktivitelere farklı öncelik sıralaması verilemeyeceği için her hat için 3! sayısı kadar öncelik sırası elenir ve toplamda $\frac{6!}{3! \cdot 3!}$ adet öncelik sırası oluşturulur.

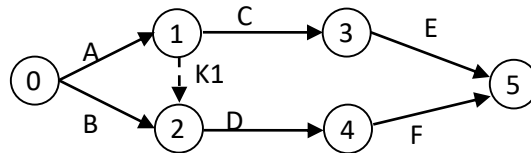
Şebekenin çözümü için Tablo 2'de sunulan öncelikler denenmiş ve elde edilen proje süreleri aynı tabloda sunulmuştur. Projenin en kısa tamamlanma süresi 37 gündür.

Tablo 2. Aktivite önceliklerinin oluşturulması.

Aktivite Öncelik Sırası	Proje Süresi	Aktivite Öncelik Sırası	Proje Süresi
ACEBDF	37	BADC FE	46
ACBEDF	37	BACDFE	46
ACBDEF	37	BACDEF	38
ACBDFE	37	BADCEF	46
ABCDFE	46	BDACEF	46
ABCDEF	38	BDACFE	46
ABDCEF	46	BDFACE	46
ABDCFE	46	BDAFCE	46
ABDFCE	46	BADFCE	46
ABCEDF	38	BACEDF	38

2.3 Örnek Problem 3

Problem 3'ün şebeke diyagramı Şekil 17'de verilmiştir. Problem 7 aktivite ve 6 düğüm noktalı bir şebekeden oluşmaktadır. Aktivite süreleri ve kaynak talebi Tablo 3'te gösterilmiştir. Şebekenin sonsuz kaynak tanımı ile gerçekleştirilen çözümü 27 gündür.



Şekil 17. Problem 3'ün şebeke diyagramı.

Örnek problem 3'ün çözümünde Tablo 4'te sunulan öncelikler denenmiştir. Kaynak kısıtları dikkate alındığında en kısa proje süresinin 37 gün olduğu tespit edilmiştir.

Tablo 3. Problem 3'ün kaynak kısıtı olmadan çözümü.

Aktivite	Süre	Kaynak
A	7	8
B	6	6
K1	0	0
C	4	7
D	8	5
E	9	4
F	12	7

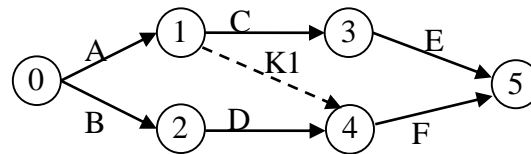
Tablo 4. Aktivite öncelikleri ve proje süreleri.

Aktivite Öncelik Sırası	Proje Süresi	Aktivite Öncelik Sırası	Proje Süresi
ABCDEF	38	ABDCEF	46
ABCDFE	46	ABDCFE	46
BACDEF	38	BADCFE	46
BACDFE	46	BADCEF	46
ABCEDF	38	ABDFCE	46
BACEDF	38	BADFCE	46
ACBDEF	37	ACBDFE	37
ACEBDF	37	ACBEDF	37

2.4. Örnek problem 4

Şebeke diyagramı Şekil 18'de verilen 7 aktiviteli ve 6 düğüm noktalı şebekenin sonsuz kaynak tanımı ile gerçekleştirilen çözümü 26 gündür. Tablo 5'te gösterilen hesap cetvelinde bazı veriler girilerek en optimum proje süresi bulunması amaçlanmıştır.

Örnek problem 4'ün çözümünde Tablo 6'da sunulan öncelikler denenmiştir. En kısa proje süresinin 37 gün olduğu tespit edilmiştir.



Şekil 18. Problem 4'ün şebeke diyagramı.

Tablo 5. Problem 4'ün kaynak kısıtı olmadan çözümü.

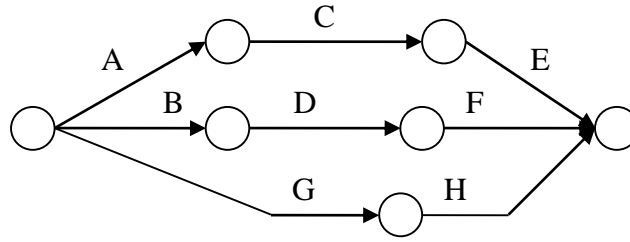
Aktivite	Süre	Kaynak
A	7	8
B	6	6
K1	0	0
C	4	7
D	8	5
E	9	4
F	12	7

Tablo 6. Aktivite önceliklerinin oluşturulması.

Aktivite Öncelik Sırası	Proje Süresi	Aktivite Öncelik Sırası	Proje Süresi
ABDCEF	46	BDAFCE	46
ABDCFE	46	ACEBDF	37
ABDFCE	46	ACBDEF	37
BADCEF	46	ABCDEF	38
BADCFE	46	BACDEF	38
BDACEF	46	ACBDFE	37
BDACFE	46	ABCDFE	46
ACBEDF	37	BACDFE	46
ABCEDF	38	BADFCE	46
BACEDF	38		

2.5. Örnek problem 5 ve 6

Şebeke diyagramları Şekil 19 ve 20’de sunulan 8 aktiviteli, düğüm noktalı şebekenin kaynak kullanımı ve aktivite süreleri sırası ile Tablo 7 ve 9’da gösterilmiştir.

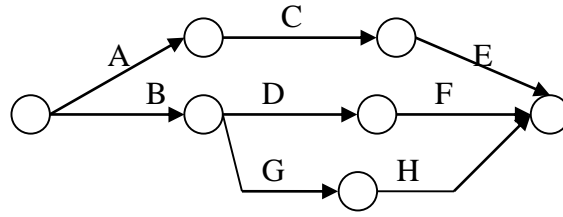


Şekil 19. Problem 5’in şebeke diyagramı.

Tablo 7. Problem 5’in aktivite süreleri ve kaynak ihtiyaçları

Aktivite	Süre	Kaynak
A	3	5
B	4	7
C	7	6
D	5	8
E	8	10
F	9	9
G	11	11
H	10	12

Örnek problem 5’in çözümünde Tablo 8’de sunulan öncelikler denenmiştir. En kısa proje süresinin 53 gün olduğu tespit edilmiştir.



Şekil 20. Problem 6'nın şebeke diyagramı

Tablo 8. Problem 5'in aktivite öncelikleri ve süreleri.

Alternatif	Gün	Alternatif	Gün	Alternatif	Gün
BDFGHACE	57	BDACFGHE	54	BDACFGEH	54
BDFGAHCE	57	BADCFGHE	54	BADCFGEH	54
BDFAGHCE	54	ABDCFGHE	54	ABDCFGEH	54
BDAFGHCE	54	BACDFGHE	53	BACDFGEH	53
BADFGHCE	54	ABCDFGHE	53	ABCDFGEH	53
ABDFGHCE	54	ACBDFGHE	53	ACBDFGEH	53
BDFGACHE	57	BDFGACEH	57	BDFACEGH	54
BDFAGCHE	54	BDFAGCEH	54	BDAFCEGH	54
BDAFGCHE	54	BDAFGCEH	54	BADFCEGH	54
BADFGHCE	54	BADFGCEH	54	ABDFCEGH	54
ABDFGCHE	54	ABDFGCEH	54	BDACFEGH	54
BDFACGHE	54	BDFACGEH	54	BADCFEGH	54
BDAFCGHE	54	BDAFCGEH	54	ABDCFEGH	54
BADFCGHE	54	BADFCGEH	54	BACDFEGH	53
ABDFCGHE	54	ABDFCGEH	54	ABCDFEGH	53
BACEDFGH	53	ABCEDFGH	53	ACBEDFGH	53
ACEBDFGH	53				

Tablo 9. Problem 6'nın aktivite süreleri ve kaynak ihtiyaçları.

Aktivite	Süre	Kaynak
A	3	5
B	4	7
C	7	6
D	5	8
E	8	10
F	9	9
G	11	11
H	10	12

Örnek problem 6'nın çözümünde Tablo 10'da sunulan öncelikler denenmiştir. En kısa proje süresinin 53 gün olduğu tespit edilmiştir.

Şekil 19 ve 20'de verilen şebekede aktivitelerin ardıl-öncel ilişkileri dikkate alınarak sırası ile 336 ve 560 adet aktivite öncelik sıralaması üretilmiştir. Tablo 8 ve 10'de sırası ile ilk 49 ve 56 alternatif hesaplanıp gösterilmiştir. Kesin çözüme 560 ve 336 aktivite sıralama alternatiflerinin oluşturulup çözülmesi ile ulaşılmıştır.

3. TARTIŞMA VE SONUÇ

Bu çalışmada tek modlu kısıtlı kaynak proje çizelgeleme problemi tüm arama uzayı taranarak çözülmüştür. Tüm arama uzayının taranması hesap yükünü arttırmakla birlikte problemin çözümü için yapılması gereken kodlama işlemlerini önemli ölçüde azaltmaktadır. Bu sayede problem önemli ölçüde optimizasyon bilgisine ve kodlama becerisine sahip olmayan mühendisler tarafından da çözülebilir hale gelmektedir.

Tablo 10. Problem 6'nın aktivite öncelikleri ve süreleri.

Alternatif	Gün	Alternatif	Gün	Alternatif	Gün
BDFGHACE	54	BDACFGHE	54	BDACFGEH	54
BDFGAHCE	54	BADCFGHE	54	BADCFGEH	54
BDFAGHCE	54	ABDCFGHE	54	ABDCFGEH	54
BDAFGHCE	54	BACDFGHE	54	BACDFGEH	54
BADFGHCE	54	ABCDFGHE	54	ABCDFGEH	54
ABDFGHCE	54	ACBDFGHE	54	ACBDFGEH	54
BDFGACHE	54	BDFGACEH	54	BDFACEGH	54
BDFAGCHE	54	BDFAGCEH	54	BDAFCEGH	54
BDAFGCHE	54	BDAFGCEH	54	BADFCEGH	54
BADFGHCE	54	BADFGCEH	54	ABDFCEGH	54
ABDFGCHE	54	ABDFGCHE	54	BDACFEGH	54
BDFACGHE	54	BDFACGEH	54	BADCFEGH	54
BDAFCGHE	54	BDAFCGEH	54	ABDCFEGH	54
BADFCGHE	54	BADFCGEH	54	BACDFEGH	54
ABDFCGHE	54	ABDFCGHE	54	ABCDFEGH	54
ACBDFEGH	54	BDACEFGH	54	BADCEFGH	54
ABDCEFGH	54	BACDEFEGH	54	ABCDEFEGH	54
ACBDEFEGH	54	ABCDEFEGH	54	ACEBDFEGH	54
BACDEFEGH	54	ACBEDFGH	54		

Tüm arama uzayının taranması hesap yükünü çok arttıracığı için bu çalışmada küçük ölçekli 6 problemin çözümü gerçekleştirilmiştir. Hesap tablosu uygulamasında hesaplamalar derleyicide derlenmiş kaynak kodlarına göre daha yavaş yürütülmesine rağmen en büyük problemde dahi 1 dakika içinde garantili en iyi çözüme ulaşılmıştır. Gerçekleştirilen örnek çalışmalar küçük ölçekli inşaat projeleri için tek modlu kısıtlı kaynak proje çizelgeleme probleminin önerilen yöntemle çözülebildiğini göstermektedir.

İnşaat sektöründe belirtilen yöntem aktivitelerin detay düzeyi azaltılarak birleştirilmesi ile az sayıda aktiviteden oluşan iş programlarında geliştirilen yöntem optimizasyon konusunda deneyimi ve teorik bilgisi bulunmayan mühendisler tarafından da uygulanabilmektedir. Bu sayede KKÇP kurumsallaşamamış küçük ve orta ölçekli yükleniciler tarafından da çözülebilecektir.

Problemin arama uzayının belirlenmesi önemli bir zorluk olarak ortaya çıkmaktadır. Çözülen tüm problemlerde arama uzayı hesap tablosuna kullanıcı tarafından tanımlanmıştır. Bu durum kullanıcının arama uzayını tanımlamasını zorunlu kılmaktadır. Özellikle büyük ölçekli problemler için sıkıntılı olabilecek bu durumun çözümü ileri çalışma olarak arama uzayını otomatik belirleyen bir algoritmanın geliştirilmesi olacaktır. Ayrıca daha hızlı çalışması için hesap tablosu yerine makro ile ara işlemlerin yapılması hesap süresini kısaltacaktır. Yöntemin eksik kısımlarına rağmen kısıtlı kaynak proje çizelgeleme probleminin çözümünde uygulanma potansiyelinin yüksek olduğu görülmektedir.

KAYNAKLAR

- [1] G. Ulusoy, L. Ozdamar, *A constrained-based perspective in resource constrained project scheduling*. International Journal of Production Research. 32 (1994) 693-705.
- [2] H. Zhang, H. Li, C.M. Tam, *Particle swarm optimization for resource-constrained project scheduling*. International Journal of Project Management. 24(1) (2006) 83-92.
- [3] R. Kanit, U.N. Baykan, M. Erdal, *Kısıtlı Kaynak Koşullarının Yapı Maliyetine Etkisinin İncelenmesi*. Politeknik Dergisi. 8(2) (2005) 209-221.
- [4] F.B. Talbot, *An Integer Programming Algorithm for the Resource-Constrained Project Scheduling Problem*. Ph.D. Dissertation, Pennsylvania State University, 1976.
- [5] P. Stinson Joel, *A Branch and Bound Algorithm for a General Class of Resource-Constrained Scheduling Problems*. Ph.D. Dissertation, University of North Carolina at Chapel Hill, 1976.
- [6] E.D. Davis, B.M. Khumawala, *Multiple Resource-Constrained Scheduling Using Branch and Bound*. AIIE Trans. 10(3) (1978) 252-259.
- [7] R. Kolisch, *Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation*. European Journal of Operational Research. 90(2) (1996) 320-333.
- [8] D. Merkle, M. Middendorf, H. Schmeck, *Ant colony optimization for resource-constrained project scheduling*. IEEE transactions on evolutionary computation. 6(4) (2002) 333-346.
- [9] S. Hartmann, *A competitive genetic algorithm for resource constrained project scheduling*. Naval Research Logistics (NRL). 45(7) (1998) 733-750.
- [10] J.H. Patterson, *Project scheduling: The effect of problem structure on heuristic performance*. Naval Research Logistics Quarterly. 23 (1976) 95-124.
- [11] L. Ozdamar, G. Ulusoy, *A survey on the resource constrained project scheduling problem*. IIE Transactions. 27 (1995) 574-586.
- [12] N. Christofides, R. Alvarez-Valdes, J.M. Tamarit, *Project scheduling with resource constraints: a branch and bound approach*. European Journal of Operational Research. 29 (1987) 262-273.
- [13] R.A.V. Olaguibel, J.M.T. Goerlich, *Heuristic algorithms for resource-constrained project scheduling: A review and an empirical analysis*. Advances in project scheduling. (1989) 113-134.
- [14] L. Özdamar, G. Ulusoy, *A survey on the resource-constrained project scheduling problem*. IIE transactions. 27(5) (1995) 574-586.

- [15] V. Van Peteghem, M. Vanhoucke, *An experimental investigation of metaheuristics for the multi-mode resource-constrained project scheduling problem on new dataset instances*. European Journal of Operational Research. 235(1) (2014) 62-72.
- [16] L. Bukata, P. Šůcha, Z. Hanzálek, *Solving the resource constrained project scheduling problem using the parallel tabu search designed for the CUDA platform*. Journal of Parallel and Distributed Computing. 77 (2015) 58-68.
- [17] F. Glover, *Future paths for integer programming and links to artificial intelligence*. Comput. Oper. Res. 13(5) (1986) 533-549.
- [18] T. Bhaskar, M.N. Pal, A.K. Pal, *A heuristic method for RCPSP with fuzzy activity times*. European Journal of Operational Research. 208(1) (2011) 57-66.
- [19] U. Beşikci, Ü. Bilge, G. Ulusoy, *Multi-mode resource constrained multi-project scheduling and resource portfolio problem*. European Journal of Operational Research. 240(1) (2015) 22-31.
- [20] U. Beşikci, Ü. Bilge, G. Ulusoy, *Resource dedication problem in a multi-project environment*. Flexible Services and Manufacturing Journal. 25(1) (2013) 206-229.
- [21] J.A. Arauzo, J.M. Galan, J. Pajares, A. Lopez-Paredes, *Efficient project portfolio management. An intelligent decision support system for engineering and consultancy firms*. Dyna. 84(9) (2009) 761-772.
- [22] X. Wang, Q. Chen, N. Mao, X. Chen, Z. Li, *Proactive approach for stochastic RCMPSP based on multi-priority rule combinations*. International Journal of Production Research. 53(4) (2015) 1098-1110.
- [23] P. Brucker, S. Knust, A. Schoo, O. Thiele, *A branch and bound algorithm for the resource-constrained project scheduling problem*. European Journal of Operational Research. 107 (1998) 272-288.
- [24] L. Zhu, J. Lin, Z.J. Wang, *A discrete oppositional multi-verse optimization algorithm for multi-skill resource constrained project scheduling problem*. Applied Soft Computing. 85 (2019) 105805.
- [25] J. Lin, L. Zhu, K. Gao, *A genetic programming hyper-heuristic approach for the multi-skill resource constrained project scheduling problem*. Expert Systems with Applications. 140 (2020) 112915.
- [26] R. Pellerin, N. Perrier, F. Berthaut, *A survey of hybrid metaheuristics for the resource-constrained project scheduling problem*. European Journal of Operational Research. 280(2) (2020) 395-416.
- [27] A. Birjandi, S.M. Mousavi, *Fuzzy resource-constrained project scheduling with multiple routes: A heuristic solution*. Automation in Construction. 100 (2019) 84-102.
- [28] M. Laszczyk, P.B. Myszkowski, *Improved selection in evolutionary multi-objective optimization of multi-skill resource-constrained project scheduling problem*. Information Sciences. 481 (2019) 412-431.
- [29] R.K. Chakraborty, A. Abbasi, M.J. Ryan, *Multi-mode resource-constrained project scheduling using modified variable neighborhood search heuristic*. International Transactions in Operational Research. 27(1) (2020) 138-167.
- [30] E.B. Tirkolaee, A. Goli, M. Hematian, A.K. Sangaiah, T. Han, *Multi-objective multi-mode resource constrained project scheduling problem using Pareto-based algorithms*. Computing. 101(6) (2019) 547-570.

- [31] S. Creemers, *The preemptive stochastic resource-constrained project scheduling problem*. European Journal of Operational Research. 277(1) (2019) 238-247.
- [32] Ö.H. Bettemir, R. Sonmez, *Hybrid genetic algorithm with simulated annealing for resource-constrained project scheduling*. Journal of Management in Engineering. 31(5) (2015) 04014082.