

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

İNŞAAT MÜHENDİSLİĞİ ANABİLİM DALI

**YAPI PROJELERİNDE ZAMAN MALİYET OPTİMİZASYONU İÇİN
YENİ BİR MODEL OLARAK NDS-II-Jaya TEKNİĞİ ÖNERİSİ**

YÜKSEK LİSANS TEZİ

Soufiane KAMİL

**ŞUBAT 2021
TRABZON**





KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

İNŞAAT MÜHENDİSLİĞİ ANABİLİM DALI

**YAPI PROJELERİNDE ZAMAN MALİYET OPTİMİZASYONU İÇİN
YENİ BİR MODEL OLARAK NDS-II-Jaya TEKNİĞİ ÖNERİSİ**

Soufiane KAMİL

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünce
"İNŞAAT YÜKSEK MÜHENDİSİ"
Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

Tezin Enstitüye Verildiği Tarih : 09 / 02 / 2021

Tezin Savunma Tarihi : 22 / 02 / 2021

Tez Danışmanı : Prof. Dr. Tayfun DEDE

Trabzon 2021



ÖNSÖZ

Bu çalışma, Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İnşaat Mühendisliği Anabilim Dalı'nda yüksek lisans tezi olarak hazırlanmıştır. Çalışmamın başından sonuna kadar istisnasız her aşamasında beni destekleyen, daha iyi ve güzel çalışmalar için devamlı teşvik eden değerli hocam sayın Prof. Dr. Tayfun DEDE'ye içten teşekkürlerimi sunarım.

Yüksek lisans öğrenimim boyunca desteklerini hiçbir zaman esirgemeyen ve bana olan inançlarını her zaman hissettiren sevgili arkadaşlarıma ve sevgili aileme çok teşekkür ederim.

Soufiane KAMİL
TRABZON 2021



TEZ ETİK BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduğum “Yapı Projelerinde Zaman Maliyet Optimizasyonu İçin Yeni Bir Model Olarak NDS-II-Jaya Tekniği Önerisi” başlıklı bu çalışmayı baştan sona kadar danışmanım Prof. Dr. Tayfun DEDE’nin sorumluluğunda tamamladığımı, verileri/örnekleri kendim topladığımı, deneyleri/analizleri ilgili laboratuarlarda yaptığımı/yaptırdığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 22/02/2021

Soufiane KAMİL



İÇİNDEKİLER

Sayfa No

ÖNSÖZ.....	III
TEZ ETİK BEYANNAMESİ	IV
İÇİNDEKİLER.....	V
ÖZET	VI
SUMMARY	VII
TABLolar DİZİNİ.....	VIII
ŞEKİLLER DİZİNİ	IX
KISALTMALAR DİZİNİ	XII
1. GENEL BİLGİLER.....	1
1.1. Giriş	1
1.2. Tezin Amacı.....	2
1.3. Tez Bölümleri	3
1.4. Literatür Taraması	3
1.4.1. Proje Planlamasında Kritik Yol Metodu (CPM)	3
1.4.2. Zaman Maliyet İlişkisi	5
1.4.3. Zaman Maliyet Ödünleşim Problemlerinin Çözümü.....	7
1.4.4. Zaman Maliyet Ödünleşim Problemleri İçin Optimizasyon Teknikleri	8
2. YAPILAN ÇALIŞMALAR.....	18
2.1. Zaman Maliyet Ödünleşim Optimizasyonu	19
2.2. Baskın Olmayan Sıralama II (NDS-II)	20
2.3. Jaya ile Optimum Çözüm	22
2.4. Zaman Maliyet Ödünleşim Problemleri İçin Baskın Olmayan Sıralama II – Jaya Algoritması	25
3. BULGULAR.....	27
3.1. İnşaat Projelerinde Zaman Maliyet Ödünleşim Problemleri İçin Baskın Olmayan Sıralama II-Jaya Uygulaması.	27
3.2. Çeşitli Faaliyetler için Sayısal Örnekler	28
3.2.1. 7 Faaliyetli Proje Örneği.....	28
3.2.2. Beş Modlu 18 Faaliyetli Proje Örneği	33
3.2.3. Üç Modlu 18 Faaliyetli Proje Örneği	37



3.2.4. 63 Faaliyetli Proje Örneđi.....	42
3.2.5. 630 Faaliyetli Proje Örneđi.....	49
4. SONUÇLAR.....	55
5. ÖNERİLER.....	57
6. KAYNAKLAR	58
ÖZGEÇMİŞ	



Yüksek Lisans Tezi

ÖZET

YAPI PROJELERİNDE ZAMAN MALİYET OPTİMİZASYONU İÇİN YENİ BİR
MODEL OLARAK NDS-II-Jaya TEKNİĞİ ÖNERİSİ

Soufiane KAMİL

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
İnşaat Mühendisliği Anabilim Dalı
Danışman: Prof. Dr. Tayfun DEDE
2021, 63 Sayfa

Gerçek dünyada, birçok amacı karşılayan en optimal çözümü bulmayı içeren birçok problem bulunmaktadır. Bu çözümü elde etmek için çok amaçlı optimizasyon modelleri gerekmektedir. Bu nedenle, bu tarz çok amaçlı optimizasyon prosedürünün gerçekleştirilmesi için bu çalışmada etkili bir Baskın Olmayan Sıralama II-Jaya yaklaşımı kullanılmıştır. Kurulan bu modelin çeşitli inşaat projelerinde verimlilikleri test edilmiştir. Böylece, kullanılan modelin Pareto ön çözümü, önceki çalışmalardan çeşitli optimizasyon problemleri çözülerek doğrulanmıştır. Bundan dolayı, NDSII-Jaya bu çalışmada oluşturulan çok amaçlı optimizasyon modeli, inşaat mühendisliğinde zaman maliyet ödünleşim problemlerini çözmek için başka bir alternatif algoritma olarak önerilebilir. Bu nedenle, bu çalışmanın ana katkısı, zaman maliyet ödünleşim problemlerini (TCTP) çözmek için inşaat yönetimi alanında NDSII-Jaya'nın geliştirilmesi ve uygulanmasıdır.

Anahtar Kelimeler: Baskın Olmayan Sıralama II, Çok Amaçlı Optimizasyon, Metasezgisel Algoritmalar, Zaman Maliyet Ödünleşim Problemi, Yapım Yönetimi



Master Thesis

SUMMARY

NDS-II - Jaya AS A NEW MODEL FOR TIME COST TRADE-OFF OPTIMIZATION TECHNIQUES IN BUILDING PROJECTS TECHNIQUE

Soufiane KAMİL

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Civil Engineering Graduate Program
Supervisor: Prof. Tayfun DEDE
2021, 63 Pages

In the real world, there are many problems that include finding the optimal solution for many purposes. To achieve this solution, multiobjective optimization models are required. For this reason, this study used a non-dominated sorting II - Jaya approach to perform this type of multiobjective optimization procedure. This implemented model has been tested for efficiency in a variety of construction projects. Thus, the Pareto front of the model used is verified by resolving several optimization problems from previous studies. The Pareto front output of the version used it has been validated by re-solving the problems of criterion optimization taken from the literature. The multiobjective optimization model created in this study based on NDSII-Jaya can be suggested as another alternative algorithm to solve time cost trade-off problems in construction engineering and management. Therefore, the main contribution of this study is the development and implementation of NDSII-Jaya within the field of construction management to solve Time Cost Trade-off problems (TCTP).

Keywords: Non-Dominated Sorting II (NDSII), Multiobjective Optimization, Time-Cost Trade-Off Problem (TCTP), Construction Management, Metaheuristic Algorithms



ŞEKİLLER DİZİNİ

Sayfa No

Şekil 1.1.	Zaman ve maliyet arasındaki ilişki	6
Şekil 2.1.	Jaya algoritmasının akış şeması.....	24
Şekil 2.2.	NDS II-Jaya algoritmasının akış şeması.....	26
Şekil 3.1.	7 Faaliyetin ayarlanması	29
Şekil 3.2.	NDSII-Jaya tarafından 7 faaliyetli problem için ulaşılan Pareto optimal çözümü.....	31
Şekil 3.3.	NDSII-Jaya kullanarak 7 faaliyet için maliyet ve iterasyon sayısı yakınsaması.....	32
Şekil 3.4.	NDSII-Jaya kullanarak 7 faaliyet için süre ve iterasyon sayısı yakınsaması.....	32
Şekil 3.5.	NDSII-Jaya kullanarak 7 faaliyet için süre, maliyet ve iterasyon sayısı yakınsaması.....	33
Şekil 3.6.	18-faaliyetine örnek projesi için faaliyet ilişkileri.....	34
Şekil 3.7.	NDSII-Jaya tarafından 7 faaliyetli problem için ulaşılan Pareto optimal çözümü.....	36
Şekil 3.8.	NDSII-Jaya kullanarak 5 modlu 18 faaliyet için maliyet ve iterasyon sayısı yakınsaması	36
Şekil 3.9.	NDSII-Jaya kullanarak 5 modlu 18 faaliyet için süre ve iterasyon sayısı yakınsaması.....	37
Şekil 3.10.	NDSII-Jaya kullanarak 7 faaliyet için süre, maliyet ve iterasyon sayısı yakınsaması.....	37
Şekil 3.11.	NDSII-Jaya tarafından 18 faaliyetli problem için ulaşılan Pareto optimal çözümü.....	40
Şekil 3.12.	NDSII-Jaya kullanarak 5 modlu 18 faaliyet için maliyet ve iterasyon sayısı yakınsaması	41
Şekil 3.13.	NDSII - Jaya kullanarak 5 modlu 18 faaliyet için süre ve iterasyon sayısı yakınsaması.....	41
Şekil 3.14.	NDSII-Jaya kullanarak 5 modlu 18 faaliyet için süre, maliyet ve iterasyon sayısı yakınsaması.....	42
Şekil 3.15.	63 faaliyetli ağın ağ temsili.....	43
Şekil 3.16.	Durum 1 için NDSII-Jaya algoritması ile elde edilen 63 faaliyetli problemin Pareto ön çözümleri.....	46
Şekil 3.17.	Durum 2 için NDSII-Jaya algoritması ile elde edilen 63 faaliyetli problemin Pareto ön çözümleri.....	47



Şekil 3.18. NDSII-Jaya kullanılan Durum 1 için 63 faaliyetli problemi analiz karşılaştırması (IC= \$2300)	48
Şekil 3.19. NDSII-Jaya kullanılan Durum2 için 63 faaliyetli problemi analiz karşılaştırması (IC= \$3500)	49
Şekil 3.20. Durum 1 için NDSII-Jaya algoritması ile elde edilen 630 faaliyetli problemin Pareto ön çözümleri.....	50
Şekil 3.21. Durum 2 için NDSII-Jaya algoritması ile elde edilen 630 faaliyetli problemin Pareto ön çözümleri.....	51
Şekil 3.22. NDSII-Jaya kullanılan Durum 1 için 630 faaliyetli problemi analiz karşılaştırması (IC= \$2300)	53
Şekil 3.23. NDSII-Jaya kullanılan Durum 2 için 630 faaliyetli problemi analiz karşılaştırması (IC= \$3500)	53



TABLÖLAR DİZİNİ

Sayfa No

Tablo 3.1.	7 faaliyetli proje için (modlar) seçenekler	29
Tablo 3.2.	7 faaliyetli problem için karşılaştırma	30
Tablo 3.3.	NDS-II-Jaya kullanılarak 7 faaliyetli problem için seçilen seçenekler	31
Tablo 3.4.	Beş modlu 18 faaliyetli proje için seçenekler	34
Tablo 3.5.	NDS-II-Jaya kullanarak 5 modlu 18 faaliyetli problem için karşılaştırma.....	35
Tablo 3.6.	Üç modlu 18 faaliyetli projeler için seçenekler.....	38
Tablo 3.7.	Üç modlu 18 faaliyetli projenin farklı algoritmaları arasında karşılaştırma.....	39
Tablo 3.8.	63 faaliyetli zaman maliyet ödünleşim problemi için veriler	44
Tablo 3.9.	NDS-II-Jaya kullanılan Durum 1 için 63 faaliyetli projenin analiz sonuçları (IC= \$2300).....	46
Tablo 3.10.	NDS-II-Jaya kullanılan Durum 2 için 63 faaliyetli projenin analiz sonuçları (IC= \$3500).....	47
Tablo 3.11.	NDS-II-Jaya kullanarak 63a ve 63b problemleri için yüzde sapma ortalamasını karşılaştırılması	48
Tablo 3.12.	630 faaliyetli proje için en iyi sonuçlar (Durum1: Günlük dolaylı maliyet 2300\$)	50
Tablo 3.13.	630 faaliyetli proje için en iyi sonuçlar (Durum 2: Günlük dolaylı maliyet \$3500)	51
Tablo 3.14.	Durum 1 için 10 koşunun ortalama değerlerinin karşılaştırması (IC = 2300 \$)	52
Tablo 3.15.	Durum 2 için 10 koşunun ortalama değerlerinin karşılaştırması (IC = 3500 \$)	52
Tablo 3.16.	NDS-II-Jaya ile elde edilen modeller için optimal çözümlerden 630 faaliyetli problemin ortalama sapmaları	52



KISALTMALAR DİZİNİ

ACO:	Karınca Kolonisi Optimizasyonu
ANOVA:	Varians Analizi
AoA:	Ağ Üzerinde Faaliyet (AoA)
AoN:	Düğümmler Üzerinde Faaliyet
APD:	Ortalama Yüzde Sapma
CPM:	Kritik Yol Metodu
DC:	Doğrudan Maliyet
DMRT:	Duncan Çoklu Aralık Testi
DTCTP:	Ayrık Zaman Maliyet Ödünleşim Problemi
EA:	Evrimsel Algoritmalar
EF:	Erken Bitiş
ES:	Erken Başlangıç
FF:	Bitişten Bitişe
FMOPSO:	Bulanık Çok Amaçlı Parçacık Sürüsü Optimizasyonu
FS:	Bitişten-Başlangıç
GA:	Genetik Algoritmalar
HA:	Hibrit algoritma
HMH:	Hibrit Meta Sezgisel
IC:	Dolaylı Maliyet
ICR:	Olaylı Maliyet Oranı
MAWA:	Değiştirilmiş Uyarlanabilir Ağırlık Yaklaşımı
MSFL:	Değiştirilmiş Kurbağa Sıçrama Algoritması
NDS-II:	Baskın Olmayan Sıralama II
NFE:	Fonksiyon Değerlendirme Sayısı
PSO:	Partikül Sürüsü Optimizasyonu
QSA:	Kuantum Simülasyonlu Tavlama
SA:	Benzetilmiş Tavlama
SAM:	Siemens Yaklaşım Yöntemi
SF:	Başlangıçtan Bitişe



SFL:	Kurbağa Sıçrama Algoritması
SS:	Başlangıçtan Başlangıca
TCGA:	Zaman Maliyet Genetik Algoritmalar
TCTP:	Zaman Maliyet Ödünleşim Problemleri
TLBO:	Öğretme-Öğrenme Tabanlı Optimizasyon



1. GENEL BİLGİLER

1.1. Giriş

İnsanlığın gelişmesiyle birlikte insanlar, evler, şehirler, köprüler kurmak için farklı etkinlikleri birleştirerek bir projeye yol açan pek çok faaliyetle yaşam biçimlerini değiştirerek medeni bir yaşam arayışına giriyor. Diğer bir deyişle, proje terimi, bina veya altyapı inşa etmek için iyi yönetilen bir dizi faaliyeti ifade eder. Ne tür bir inşaat projesi olursa olsun, bir yönetici, faaliyetleri hızlı ve verimli bir şekilde yürütmek için grubun önemli bir üyesidir. Proje yöneticisi, projeyi müşteri ile yüklenici arasında planlandığı şekilde, hedeflenen bütçe ve son tarihte bitirme sorumluluğuna sahiptir.

Son dönemlerde proje yönetimi; süre, maliyet ve kaynaklar olmak üzere, üç ana unsurla ilgili olarak hedeflenen tarihe minimum zaman ve maliyetle ulaşma rolü nedeniyle tüm projelerde önemli bir ilgi görüyor. Günümüz inşaat piyasasında, projenin zamanını ve maliyetini minimuma indirebilme kabiliyeti farklı şirketlerin verimliliğini ve hayatta kalmasını belirliyor olmasına rağmen herhangi bir proje süresi ve bütçesinin, planlanması ve yönetilmesinde karşılaşılan zorluklar müşteri ve müteahhitlerin ilgisini çeken asıl problemlerdir. Herhangi bir inşaat projesinde hem zamanı hem de maliyeti azaltmak, proje planlamasında çok önemlidir, inşaat projesinin zamanı ve maliyeti arasındaki ilişki, zaman maliyet ödünleşim kararları olarak bilinir. Proje başarısında büyük ölçüde önemli sorunlardan biri olduğu için proje yöneticileri zaman maliyet ödünleşim problemlerine dikkat çekiyorlar. Zaman maliyet ödünleşim problemleri (TCTP) proje yönetiminde oldukça kritiktir.

Zaman maliyet ödünleşim problemleri genellikle beklenen bir gecikmeyi telafi etmek için bir faaliyetin süresinin kısaltılması gerektiğinde veya projeyi planlanan son tarihten önce bitirmek zorunda olduğunda sıralanır. Herhangi bir projenin süresinin artırılması, yeni kaynakların kullanılmasını gerektiren bazı faaliyetlerin süresinin kısaltılması nedeniyle bütçeyi etkiler ve bu da projenin maliyetini yükseltir. Bu nedenle, proje süresi ve doğrudan maliyet azalan bir eğri ilişkisine sahiptir; proje süresi en aza indirilirse, dolaylı maliyetlerde artışa neden olacağı anlamına gelir. Projenin dolaylı maliyetleri proje süresinden etkilenir. Projenin toplam maliyeti, doğrudan ve dolaylı maliyetlerin toplamıdır. Proje için doğrudan maliyet, malzeme, ekipman, işçilik ve alt yüklenicilerin maliyetlerini



kapsar. Aksine, dolaylı maliyetler, belirli bir faaliyetle ilişkilendirilemeyen zorunlu faaliyetlerin yürütülmesi maliyetleridir. Sonuç olarak, zaman ile doğrudan maliyet arasında bir ödünleşim vardır: Kaynaklar ne kadar az olursa, süre o kadar uzar; herhangi bir faaliyetin süresini kısaltmak, doğrudan maliyetini otomatik olarak artıracaktır. Proje yönetiminde zaman ve maliyet en önemli amaçtır. Zaman maliyet ödünleşim problemlerinin çözümü inşaat sektöründe kullanılan Kritik yol metodu'na (CPM) bağlıdır. Bu yöntemin amacı kritik yolu, toplam süreyi ve hangi faaliyetlerin ertelenebileceğini belirlemektir. Kritik bir yol, proje süresini etkilemeden değiştirilemeyen ağ diyagramı aracılığıyla bir dizi koordineli faaliyettir; ayrıca azami sürenin belirlenmesine yardımcı olur.

Günümüzde, proje yöneticileri zaman ve maliyet proje planlamasında asıl amaç olarak dikkate alırlar. Bu rekabetçi sektörde yetenekli yönetici, projeyi en uygun zaman ve maliyetle bitirebilen kişidir. Bu nedenle, bu gibi durumlarda, hedeflenen maliyet ve süreye etkinlik ve verimlilikle ulaşmak için proje yönetiminde iki amaç arasında denge kurmak için optimizasyonun kullanılması gerekmektedir. Son zamanlarda optimizasyon kullanımı; plan, üretim vb. konularda mükemmel sonucu bulmaya yardımcı olduğu için sadece inşaat sektöründe değil, aynı zamanda birçok endüstri alanında da çok yüksektir.

1.2. Tezin Amacı

Bu çalışmada, inşaat yönetimi alanında zaman maliyet ödünleşim problemlerinin çözümünde Pareto ön performansını göstermek için çok amaçlı baskın olmayan sıralama II (NDS-II) tekniği güncel bir optimizasyon algoritması olan Jaya algoritma ile birleştirilmiştir. Zaman maliyet ödünleşim (TCT) modelini geliştirmek için, kritik yol yöntemi (CPM) bir iş programlaması olarak dikkate alınarak gerekli kodlamalar MATLAB bilgisayar programlama dilinde oluşturulmuştur.

Projenin zaman ve maliyetinin en aza indirilmesi amaç fonksiyonları olarak dikkate alınmıştır. İnşaat yönetimi optimizasyon problemlerinde performans değerlendirme kriterlerini yerine getirmek için, baskın olmayan sıralama II-Jaya algoritmasının pareto sonuçları literatürde verilen diğer optimizasyon problemlerinin pareto sonuçları ile kıyaslanmıştır. Böylelikle, baskın olmayan sıralama II-Jaya (NDSII-Jaya) algoritmasının etkin bir şekilde çalışır ve zaman maliyet problemlerinin optimizasyonu için kullanılanbilen etkin bir yöntem olduğu ortaya koyulmuştur.



1.3. Tez Bölümleri

Bölüm 1’de, kritik yol metodu (CPM) hakkında kısa bir giriş ve ardından literatür taraması sunulmaktadır. Daha sonra zaman maliyet ödünleşim (TCT) analizi hakkında bilgi verilmektedir. Zaman maliyet ödünleşim problemleri için çözüm zorlukları, çeşitli kategoriler ve kullanılan yöntemler uygulanmaktadır.

Bölüm 2’de, baskın olmayan sıralama II algoritm ve Jaya yöntemi sunulmaktadır. Ayrıca optimizasyon problemlerini çözmek için uygulaması açıklanmaktadır.

Bölüm 3’de, baskın olmayan sıralama II- Jaya algoritmasının önceki çalışmalardan alınan örneklerle uygulanması sunulmuştur. Elde edilen sonuçlar tartışılır ve önceki araştırmalarla karşılaştırılmaktadır.

1.4. Literatür Taraması

Bu bölümde, proje planlamasında kritik yol metodu, zaman maliyet ilişkisi, zaman maliyet ödünleşim problemlerinin çözümü ve zaman maliyet ödünleşim problemleri için optimizasyon teknikleri alt başlıklar altında detaylı bir şekilde incelenmiştir.

1.4.1. Proje Planlamasında Kritik Yol Metodu (CPM)

Tüm projelerin planlanan zamanda bitirilmesi için iyi bir programa ihtiyacı vardır. Faaliyetlerin sırası ve süreleri programdaki önemli hedeflerdir. Kritik yol metodu, basitliği ve verimliliği nedeniyle inşaat sektöründe en çok kullanılan tekniklerden biridir. Kritik yol metodu, projenin grafiksel bir görünümünü oluşturmayı amaçlar ve her bir faaliyeti gerçekleştirmek için gereken minimum tamamlanma süresini ve kaynakları hesaplar. Ayrıca projenin gecikmeden sonuçlanması için dikkate alınması gereken kritik faaliyetleri belirler. Dahası, CPM, ekipmanı, malzemeleri, yöntemleri ve çalışma saatlerini seçme zamanı ve maliyeti için en iyi planlamayı belirler.

Kritik yol metodunun en göze çarpan özelliklerinden biri, proje süresinin kısaltılması gerektiğinde uygun proje faaliyetlerini kısaltarak toplam proje maliyetini optimize etme girişimidir.



Kritik yol metodunda, ağ üzerindeki faaliyet ve düğüm üzerindeki faaliyet olarak belirtilen iki tür ağ vardır. İlki, faaliyetlerin düğümleri birbirine bağlayan oklar üzerinde olduğunu gösterir. İkincisi, faaliyetlerin düğümlerde görüldüğünü gösterir (Hegazy, 2010). CPM'de iki türü şebeke vardır, ok üzerinde faaliyet (AoA) ve düğüm üzerinde faaliyet (AoN). AoA'da bağlantılar faaliyetleri gösterir ve düğümler faaliyetlerin başlama ve bitişini gösterir. AoN'da noktalar faaliyetleri gösterir ve bağlantılar faaliyetler arasındaki öncelik ilişkilerini gösterir. Ağ üzerinde faaliyet yöntemi (AoA) avantajları nedeniyle daha çok kullanılan yöntemdir.

Bir projeyi CPM yöntemine göre planlarken, kullanılabilecek bazı ifadeler şunlardır (Siemens,1971):

- Faaliyet: Projeyi tamamlamak için gereken görevin performansı
- Süre: Bir etkinliği gerçekleştirmek için tahmini süre
- Yol: Başlangıç olayından bitiş olayına giden bağlantılı bir faaliyet dizisi
- Kritik Yol: Projenin süresini temsil eden, proje için en uzun süreyi gösteren bir dizi faaliyet
- Erken Başlama Zamanı: Bir faaliyetin başlayabileceği en erken zaman. Erken Başlama Zamanı hemen öncekinin maksimum erken bitişine eşittir.
- Erken Bitirme Süresi: Bir faaliyetin bitebileceği en erken zaman. Erken Bitirme Süresi, en erken başlangıç zamanı artı faaliyet süresine eşittir.
- Geç Başlama Zamanı: Kritik yolların süresini etkilemeden bir faaliyetin başlayabileceği en son zaman.
- Geç bitiş Tarihi: Bir faaliyetin kritik yolların süresini geciktirmeden bitirebileceği en son zaman.
- Serbest Bolluk: Bir faaliyetin, sonraki faaliyete herhangi bir gecikmeye neden olmadan ertelenebileceği süre.
- Toplam Bolluk: Bir faaliyetin kritik hale gelmeden önce geciktirilebileceği maksimum süre.
- İleri Geçiş: Tüm ağ etkinliklerinin tamamlanmamış kısımları için erken başlangıç ve erken bitiş tarihlerinin hesaplanması.
- Geriye Geçiş: Tüm program etkinliklerinin tamamlanmamış kısımları için geç bitiş tarihlerinin ve geç başlangıç tarihlerinin hesaplanması.



Kritik yol yönteminde ağların kurulmasında dört mantıksal ilişki ele alınır, Bitişten-Başlangıca, Başlangıçtan-Başlangıca, Bitişten-Bitişe, Başlangıçtan-Bitişe, daha fazla ayrıntı;

- En yaygın mantıksal ilişki Bitişten Başlangıca'dır (FS). Örneğin, inşaata başlamadan önce şantiyeyi hazırlamanız gerekir.
- Bitişten Bitişe (FF) bağımlılık, paralel olarak gerçekleşen ancak tamamlanamayan faaliyetler için kullanışlıdır. Bir faaliyet olarak bir dizi elektrik tesisatı yapılıyorsa ve aynı zamanda bir klima kuruluyorsa, elektrik işi yapılan kadar o klima çalışmayacak.
- Faaliyetler başka bir faaliyetin başlangıcına bağlı olduğunda, bu bir Başlangıçtan Başlangıca (SS) ilişkidir. Küçük bir gecikme süresine sahip bir Başlangıçtan Başlangıca ilişkiniz vardır.
- En seyrek kullanılan mantıksal ilişki Başlangıçtan Bitişe'dir (SF). Bu normalde bir devir teslim görevidir. Değişiminiz gelene kadar vardiyanızı tamamlayamazsınız, bu da bir sonraki vardiyanın başladığını gösterir.

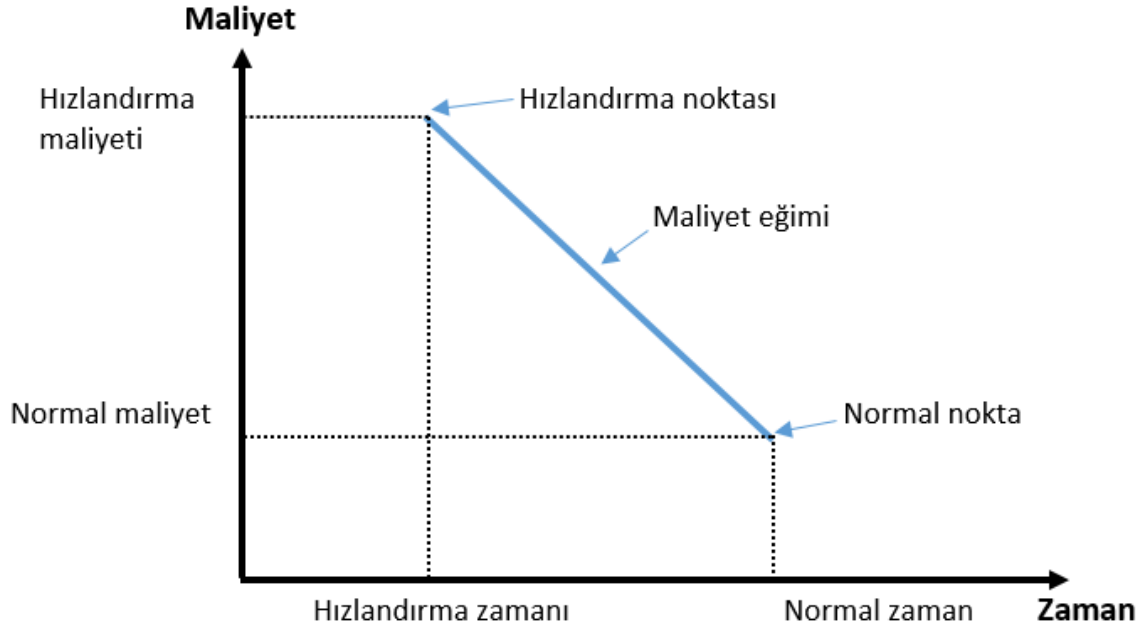
Zaman maliyet ödünleşim analizinin amacı, proje süresini en düşük maliyetle belirli bir son teslim tarihine ulaşmak için düşürmektir. Kritik yoldaki bazı faaliyetler zamanı korumak için uygun modla değiştirildiğinden ve diğerleri maliyetten tasarruf etmek için yumuşatıldığından, sınırlı süreli projede zaman maliyet ödünleşim analizi tercih edilir. Bununla birlikte, kritik yol yöntemi (CPM), proje son tarihi sabit olmadığında uygun bir zamanlama tekniğidir. Bazen erken gecikmeleri telafi etmek için projeyi belirli bir zamanda bitirmek, projeyi erken tamamlama faydaları elde etmek için önceden belirlenmiş bir son tarihte tamamlamak ve proje nakit akışını iyileştirmek gerekebilir. Proje süresinin kısaltılması; faaliyetler arasındaki örtüşmeleri ayarlayarak, faaliyetlerin sürelerini birden çok vardiyalı çalışma vasıtasıyla azaltarak, fazla mesai yaparak veya ek kaynaklar kullanarak gerçekleştirilebilir.

1.4.2. Zaman Maliyet İlişkisi

Genel olarak, bir faaliyeti tamamlamak için zaman ve maliyet, özellikle de doğrudan maliyet arasında bir ödünleşim ilişkisi vardır. Bir faaliyeti tamamlamak için ne kadar uzun süre gerekirse, kaynakların fiyatı da o kadar düşük olacaktır. Herhangi bir faaliyetin



süresinin azaltılması, doğrudan maliyetini artıracaktır. Bu, dağıtılan kaynakların miktarı ve görev süresi ile orantılıdır (Eirgash, 2018). Örneğin bir faaliyetin bir işçiyle 5 günde yapılmasını talep etmesi, beş işçinin bir günde o faaliyeti bitireceği anlamına gelmez. Bir faaliyetin süresi ile doğrudan maliyetleri arasındaki olası ilişkinin basit bir temsili Şekil 1.1’de görülmektedir. Yalnızca bu faaliyeti tek başına ve proje tamamlama son tarihine atıfta bulunmadan dikkate alarak, bir yönetici, minimum doğrudan maliyet anlamına gelen ve normal süre adı verilen bir süre seçecektir. Diğer taraftan, bir yönetici, faaliyeti hızlandırma süresini mümkün olan minimum sürede, ancak maksimum maliyetle tamamlamayı seçebilir (Elbeltagi, 2006).



Şekil 1.1. Zaman ve maliyet arasındaki ilişki

Bazı ara noktalar, faaliyet için zaman ve maliyet arasındaki ideal veya optimum ödünleşimi temsil edebilir. Normal nokta ile çarpışma noktasını birbirine bağlayan çizginin eğimi, faaliyetin maliyet eğimi olarak adlandırılır.

Her faaliyet, bu durumda minimum doğrudan maliyetle sonuçlanan süre için planlandysa, tüm projeyi tamamlama süresi çok uzun olabilir ve projenin geç tamamlanmasına bağlı olarak önemli cezalar ortaya çıkabilir. Tüm bunlardan kaçınmak için, planlamacılar proje süresini kısaltmak için zaman maliyet ödünleşim analizi olarak bilinen şeyi gerçekleştirirler. Hem müteahhit hem de müşteri projeyi planlandığı gibi veya



mümkün olan minimum bütçe ile beklenenden erken bitirmek istiyorsa, optimizasyon her ikisi için de en iyi çözümdür.

Optimizasyon, farklı olası çözümlere sahip soruna mümkün olan en uygun çözümü bulmak için kullanılan prosedürdür. Mevcut tüm çözümler arasından en uygun çözümü bulmak önemli miktarda zaman alabilir. Önerilen tüm çözümlerin analizi, kritik yol yöntemi (CPM) kullanılarak programın yeniden hesaplanmasını ve toplam proje maliyetinin yeniden değerlendirilmesini içerir. Bu nedenle kapsamlı sayım, çok hızlı bilgisayarlarla bile uygulanabilir ve pratik bir çözüm değildir (Hegazy,1999). Aslında bu süreç, en iyi zaman ve maliyet kombinasyonunu bulmak için mevcut optimizasyon yöntemleriyle kısaltılabilir.

1.4.3. Zaman Maliyet Ödünleşim Problemlerinin Çözümü

Zaman maliyet ödünleşim problemlerini çözmek için iki alana ayrılmış farklı teknikler kullanılmıştır: Matematiksel programlama yöntemi ve sezgisel yöntemler. Matematiksel programlama yöntemi, zaman maliyet ödünleşim problemlerini çözmek için, doğrusal programlama, tamsayı programlama ve dinamik programlama kullanır. Doğrusal programlama yaklaşımı ayrık zaman maliyet ilişkisi için uygun değildir (Feng vd., 1997). Tamsayı programlama, büyük ve karmaşık ağlar için daha fazla hesaplama çabası gerektirir. Dinamik programlama, saf seri veya paralel alt ağlara ayrıştırılabilen ağlar için zaman maliyet ödünleşim problemlerini çözmek için kullanılır. Sezgisel yöntemler, optimal çözümü garanti etmez, ancak iyi çözümler sağlar. Sezgisel yaklaşım, optimal çözümleri garanti etmeyen belirli seçim kriterlerine göre kısaltılacak veya genişletilecek faaliyetleri seçer (Mohammadi, 2011). Genel olarak yöntemler, faaliyetlerle doğrusal zaman maliyet ilişkisini dikkate alır; ancak olası çözümler yelpazesi sağlamaz. Zaman maliyet ödünleşim problemi, aynı anda optimize edilmesi gereken iki hedefli zaman ve maliyete sahip çok amaçlı bir optimizasyon problemidir. Çoklu optimizasyon probleminde tek bir optimum çözüm yerine bir dizi optimizasyon çözümü elde edilir.

Optimum çözümler kümesi baskın olmayan veya pareto optimal olarak adlandırılır (Pathak ve Srivastava, 2007). Çözüm alanında, pareto cephesinin üyelerine diğer üyeler baskın değildir. Çok amaçlı optimizasyon problemini çözmek için evrimsel algoritmalar (EA) tercih edilir. EA, olası çözümler dizisiyle ilgilenir ve tek bir algoritma çalışmasında en uygun pareto çözümlerinin tamamını bulmayı sağlar (Mohammadi, 2011). Farklı



uygulamalar için farklı evrimsel algoritmalar parametrelerinin ayarlanması çok zaman alabilir. Evrimsel algoritmalar (EA), kullanıcının faaliyetlere kaynak tahsis etme sırasını belirleyen optimum öncelik değerinin bulmasını sağlar (Jaskowski ve Sobotka, 2006). Evrimsel algoritmalar, verilen problem üzerinde aynı algoritmanın farklı çalışmaları için farklı sonuçlar üretir; bu nedenle, bu problemdeki performanslarını açıklamak için birden fazla çalışma gerekir.

1.4.4. Zaman Maliyet Ödünleşim Problemleri için Optimizasyon Teknikleri

İnşaat projelerinde zaman ve maliyet arasındaki dengeyi optimize etmek için birçok optimizasyon tekniği önerilmiştir. Doğrusal programlama, tamsayı programlama ve dinamik programlama gibi matematiksel teoriye dayalı optimizasyon yöntemleri, zaman maliyet ödünleşim problemlerini çözmek için kullanılan ilk optimizasyon yöntemidir. Üstelik sezgisel yöntemler, anlaşılması kolay ancak iyi çözümler sunamayan basit bir pratik kuraldır, ayrıca büyük ölçekli problemlere uygulanabilen meta-sezgisel algoritmalar vardır.

Doğrusal programlama, doğrusal zamanlı maliyet ilişkilerinde sorunları çözmek için uygun bir yöntemdir ancak ayrı zaman maliyetli ilişkilerde sorunları çözemez (Feng vd., 1997). Tamsayı programlama ve dinamik programlama, daha karmaşık proje ağlarını çözmek veya çok sayıda faaliyetle projeleri çözmek için çok fazla bilgi işlem gücü gerektirir. Büyük TCT problemlerini çözmek için matematiksel optimizasyon yöntemlerinin sınırlamaları geçmek için meta-sezgisel optimizasyon yöntemleri alternatif olarak tanıtıldı. Zaman maliyet ödünleşim problemlerini çözmek için, çok amaçlı optimizasyon modellerinin en uygun çözümü geliştirilmiştir. Ek olarak, bilgisayarlardaki ve sayısal yöntemlerdeki gelişmelerin dışında, sezgisel, meta-sezgisel ve kesin algoritmalar, zaman maliyet ödünleşim problemlerini çözmek için kullanılan yaygın optimizasyon algoritma yöntemleridir.

Optimizasyon algoritmaları çok fazla hesap hacmi kullandıkları için yüksek kaliteli bilgisayarlar ve profesyonel kodlama teknikleri gerektirmektedirler. Ayrıca bu algoritmalar elde ettikleri sonuçların optimal olduğunu göstermeleri gerekmektedir. Doğrusal programlama, dinamik programlama ve karma tamsayılı programlama, zaman maliyet ödünleşim problemlerini çözmek için kesin algoritmalara bazı örneklerdir.



Meyer ve Shaffer, (1965) zaman maliyet ödünleşim problemlerini çözmek için Karma tamsayı programlamayı ilk uygulayanlardır. Daha sonra zaman maliyet hedef fonksiyonunu azaltmak için esnek bir karışık tamsayı modeli önerildi (Moussourakis ve Haksever, 2004). Modelleri, doğrusal, parçalı doğrusal veya ayırık dahil olmak üzere farklı türdeki amaç işlevleriyle başa çıkabilir. Buna ek olarak, bu model, optimum toplam maliyeti elde etmek için son tamamlanma tarihini kısıtlama olarak alır.

Prabuddha vd., (1997), ayrı zaman maliyet ödünleşim probleminin çözümü için önceden geliştirilen modellerin dezavantajını tartışmaktadır. Merkezileştirilmiş bir yaklaşım ve modüler ayrıştırma yaklaşımı olarak gösterilen dinamik programlamaya dayanan iki çözüm modeli, zaman maliyet ödünleşim problemlerinin çözümünü belirlemek için Prabuddha, vd. (1997) tarafından gerçekleştirilmiştir. Ayrıca optimizasyon süreçlerinde, ikinci çözüm modellerinde de paralel modüller kullanmışlar.

Demeulemeester vd., (1998) Visual C ++ platformunda ayırık TCT problemi için kesin bir çözüm modeli geliştirdi. Modelleri, ufuk değiştiren bir yaklaşımla geliştirilmiş dal ve sınır optimizasyon modeline dayanıyor. Belirledikleri iki farklı kuralı kullanarak ayırık zaman maliyet ödünleşim eğrileri için hesaplanan dışbükey parçalı doğrusal tahminlerin niteliklerini incelediler. Modelleriyle yapılan sayısal deneylerden elde edilen sonuçlar, faktöriyel deney vasıtasıyla doğrulandı ve Demeulemeester, vd., (1996) tarafından bildirilen sonuçlarla karşılaştırıldı.

Vanhoucke (2005), Yang ve Chen (2000) tarafından önceden incelenen zaman değişim kısıtlı ayırık zaman maliyet ödünleşim problemlerini değerlendirmiştir. Bu kısıtlamalar, faaliyetlerin gündüz, gece ve hafta sonu vardiyalarına uygulanan belirli başlangıç zamanını ve etkin olmayan zaman aralıklarını ifade eder. Demeulemeester vd. (1998) tarafından geliştirilen alt sınır hesaplama yaklaşımının ışığında dal ve sınır algoritmasının yeni bir çeşidini sundular.

Sezgisel algoritmalar, incelenen problemlerin çözümlerini ortaya koymak için basit kurallar uyguladılar, bu yüzden karmaşık problemlerde çaba harcamadan kullanılmaları kolaydır. Bununla birlikte, sezgisel algoritmalar için, elde edilen sonucun küreselliği her zaman sorgulanabilir, çünkü bunlar genellikle yerel veya yakın küresel çözümleri bulabilirler. Bu yöntemler, uygun çözümü üretmek için bir algoritma kullanır. Genel olarak, yordamın sonucuna ulaşılmadığı sürece, buluşsal yöntemlerin gelişim süresi boyunca uygun bir çözüm elde edilmez. Sezgisel yaklaşımlar ve matematiksel programlama modelleri olarak iki ilişkili çözüm türü kullanılmıştır. Matematiksel



programlama modellerine yönelik temel eleştiriler; karmaşık formülasyonları, sayısal yoğunlukları, küçük ölçekli problemlere uygulanabilirliği ve yerel minimum çözümleri içermektedir (Feng vd., 1997; Li ve Love, 1997; Moselhi, 1993).

Siemens (1971) Siemens yaklaşım yöntemi (SAM) olarak adlandırılan bir yöntem geliştirdi. Bu yöntem sezgisel mantık ve analize dayalı mantıksal sistematik bir prosedürdür. SAM, sezgisel bir yöntem olarak adlandırılır. Model, dışbükey doğrusal olmayan zaman maliyet ödünleşim problemlerini birden fazla eğrisel parçayla yaklaşık olarak yaklaştıran doğrusal hale getirir. Siemens yaklaşım yöntemi (SAM), proje ağının kurumu ile başlar, ardından en az maliyetle faaliyetleri hızlandırmak için bir dizi kural izler. Bu modelin sonuçları, kesin algoritmalarla elde edilenlerle iyi bir düzenlilik göstermektedir. Sonuç olarak, Siemens (1971) modelin faaliyetlerin ait olduğu farklı yolların sayısını göz ardı etmesi ve minimum maliyet eğimi ile çalıştığı için, hedeflenen miktarın dışında çok daha kısa bir proje süresi üretebileceğini belirtmiştir.

Maliyet eğimi yöntemi olarak da bilinen Siemens yaklaşım yöntemi, zaman maliyet ödünleşim problemlerini çözmek için basit bir sezgisel yaklaşımdır. Zaman ve maliyet arasındaki ilişkinin doğrusal olduğunu varsayarsak, bu yöntem proje süresini kısaltır. Bu varsayıma göre, bir faaliyetin maliyet eğimi, süresi birim zaman kısaltıldığında doğrudan maliyetin arttığı oran olarak tanımlanır. Maliyet eğimi yönteminin ayrıntılı adımları aşağıdaki gibidir:

1. Proje ağı çizilir.
2. Kritik Yol Metodu (CPM) hesaplamaları yapıp ve kritik yol belirlenerek, tüm faaliyetler için normal süreleri ve maliyetleri kullanılır.
3. Her faaliyet için maliyet eğimini aşağıdaki denklemden hesaplanır:

$$\text{Maliyet eğimi} = \frac{\text{Hızlandırılmış maliyet} - \text{Normal maliyet}}{\text{Hızlandırılmış süre} - \text{Normal süre}}$$

4. Kritik yoldaki en düşük maliyet eğimine sahip ve hızlandırma süresine kadar kısaltılmamış faaliyet süresini kısaltarak başlanır.
5. Kritik faaliyetlerin süresini, hızlandırma süresine ulaşılan kadar veya kritik yol değişene kadar en düşük maliyet eğimiyle azaltılır.
6. Birden fazla kritik yol söz konusu olduğunda, kısaltılacak faaliyet(ler), (varsa) tüm kritik yollar üzerinde bulunan faaliyetin maliyet eğimi ile her biri bir kritik



yol üzerinde bulunan bir faaliyet grubunun toplam maliyet eğimini karşılaştırarak belirlenir.

7. Kritik bir yolu kısalttıktan sonra, faaliyet zamanlamalarını ve bollukları ayarlanır.
8. Faaliyet kısılmasına bağlı maliyet artışı, maliyet eğiminin kısaltılan zaman birimlerinin süresi ile çarpılmasıyla hesaplanır.
9. Daha fazla kısaltma mümkün olmayana kadar ve hızlandırma noktasına ulaşana kadar devam edilir.
10. Sonuçlar, kümülatif maliyet artışına karşı proje tamamlanma süresi çizilerek, grafiksel olarak gösterilebilir. Bu, projenin doğrudan maliyet zaman ilişkisidir. Proje zaman-maliyet eğrisini elde etmek için bu eğriye proje dolaylı maliyeti eklenebilir. Bu eğri, optimum süreyi ve buna karşılık gelen minimum maliyeti verir.

Demeulemeester (1996), ayrı zaman maliyet ödünleşim probleminin üç genişlemesini araştırıyor; birincisi zaman değişim kısıtlamaları (Prabuddha vd., 1997), ikincisi iş sürekliliği kısıtlamaları (Vanhoutte, 2005) ve sonuncusu net mevcut değer maksimizasyonudur (Yang ve Chen, 2000). Visual C ++'da programlanmış ağ üzerinde faaliyet yöntemini (AoA) dikkate alarak farklı meta sezgisel algoritmalar verirler. Uygulanan algoritmanın sezgisel bölümü, muhit avı ve çeşitliliği sürdürme girişimi üzerine bir onaylama içerir. Algoritmalarının ikinci kısmı, kritik olmayan faaliyetlerin zaman aralığını artırırken, uygun bitiş tarihini elde eden dinamik bir programlamayı birleştirir. Karşılaştırılan sonuçlar, önerilen algoritmanın farklı zaman maliyet ödünleşim probleminin net mevcut değer versiyonlarında uygulanabilir olduğunu ortaya koymaktadır.

Son yıllarda, zaman maliyet ödünleşim problemlerini çözmek için genetik algoritmalar (GA), benzetilmiş tavlama (SA), parçacık sürüsü optimizasyonu (PSO), karınca kolonisi optimizasyonu (ACO) ve karıştırılmış kurbağa sıçrama (SFL) optimizasyonu gibi çeşitli modern meta-sezgisel optimizasyon yöntemleri uygulandı. Bu alternatif optimizasyon yöntemlerinin matematiksel optimizasyon yöntemlerine göre bazı avantajları olmasına rağmen, çoğu zaman maliyet ödünleşim probleminin optimizasyonu için başarı ile uygulanmıştır.

Meta sezgisel algoritmalar, geçmişte sıklıkla sezgisel olarak anılan stokastik bileşenlere sahip bir algoritmadır. Ancak son literatür bunlardan metasezgisel olarak bahsetme eğilimindedir. Meta sezgisel, normalde yerel bir iyimserlik arayışında



üretilemlerin ötesinde çözümler üretmek için diğler buluşsal yöntemleri yönlendiren ve değıştiren bir ana strateji olarak düşünölebilir (Glover ve Laguna, 1999). Ek olarak, tüm meta sezgisel algoritmalar, rastgele seçim ve belirli bir yerel arama ödünleşimi kullanır. Zor optimizasyon sorunlarına kaliteli çözümler, makul bir süre içinde bulunabilir. Ancak optimum çözümlere ulaşılacağıının garantisi yoktur. Bu algoritmaların çoğu zaman çalıştığı, ancak her zaman çalışmadığı umulmaktadır. Neredeyse tüm meta sezgisel algoritmalar, global optimizasyona uygun olma eğilimindedir. Makul bir süre içinde, zor optimizasyon sorunlarına kaliteli çözümler sunarlar, ancak optimum çözümlere ulaşılacağıının garantisi yoktur. Aynı zamanda, global çözümler yerine optimuma yakın çözümleri buldukları için, global çözümlere ulaşmanın çok zor olduğu durumlar için çok yararlı bir araçtır. Diğlerlerinin yanı sıra, genetik algoritmalar (GA), karınca kolonisi optimizasyonu (ACO), partiköl sürüsü optimizasyonu (PSO) ve benzetilmiş tavlama (SA) en bilinen meta sezgisel algoritmalarlardır.

Bilgisayar ve yapay zekâ alanındaki son gelişmelerin etkisiyle genetik algoritmalar (GA) ortaya çıktı. GA'lar, doğal seçim ve en uygun olanın hayatta kalma süreçlerini simöle ederek bir problemin en iyi çözümü için rastgele bir arama uygular. Birden fazla kaynaktan bağlam bilgisi genetik algoritmalarda bulunabilir (Goldberg ve Segrest, 1987; Zheng vd., (2005). GA'ların sağlamlığı ve çok modlu ortamda küresel maksimumu başarılı bir şekilde arama ve bulma yetenekleri araştırmacılar tarafından belgelenmiştir (Goldberg ve Segrest, 1987); (Li ve Love, 1997). Genetik algoritmalar, algılanan avantajları nedeniyle bir dizi altyapı ve inşaat yönetimi sorununu çözmek için başarıyla kullanıldı.

Uygulamalar, bir yüklenicinin işaretleme tekniğinin geliştirilmesini gerektirir (Hegazy ve Moselhi, 1994). GA tabanlı uygulamaların en önemli dezavantajı, taramanın önemli miktarda hesaplama süresi almasıdır (Feng vd., 1997) önceki doğrusal programlama tamsayı modellerinin bir iyileştirmesi olan GA modeli oluşturmuşlardır. Model, bir tahta üzerine inşa edildi ve doğrudan masrafla optimal takas eğrisi biçimini elde etmek için her iş için inşaat alternatifleri olarak değlerlendirildi. En iyi zaman maliyet ödünleşim yaklaşımını değlerlendirmek için dolaylı maliyet daha sonra uygulandı. Ancak model, bitişten-başlangıca kadar ortaklıklar içeren basit ağlarla sınırlıdır ve küçük kaynakları barındıramaz. Li ve Love, (1997) Model ise, gerçek sayılarda süreler oluşturmak için formöle edilmiştir. Bu şekilde her türlü önemli işlemi en aza indirgenmiştir. Araştırma ayrıca, hesaplama süresini basit genetik algoritmalara indirgeyen çeşitli iyileştirmeleri de içeriyor. Model, çarpışma aşamasında diğler önemli yolların



oluşturulmasını dikkate almayıp, gizli olanlar yerine çarpışma dönemleri için sabit değişkenlerle sınırlandırıldı.

Goldberg ve Segrest (1987), Microsoft Excel’de özetlenen bir arayüzle bilgisayar programı (TCGA) geliştirdikten sonra algoritmaları üzerinde anlaştılar. Algoritma kullanılarak elde edilen sonuçlar, farklı bir zaman maliyet ödünleşim problemi oluşturan 18 faaliyetli bir inşaat projesi için yüksek bir yüzdeyle en uygun çözümleri bulabildiğini fark etmişler.

Zheng vd. (2005), muhit optimizasyonuna takılma olasılığını azaltarak genetik sürüklenme gerçeğini dengelemeye çalışan GA ispatını önerdi. Bu nedenle (Zheng vd., 2005), önceki neslin doğasına ilişkin uygunluk değeri ihtiyacını ayarlamak için değiştirilmiş bir uyarlanabilir ağırlık yaklaşımını (MAWA) birleştirir. Nesiller arttıkça, değiştirilmiş uyarlanabilir ağırlık yaklaşımı (MAWA), erken durma koşullarına karşı koymak amacıyla mutasyon oranı için azalan bir kalıp yönlendirir. Bu model, özellikle daha küçük genel giderler getiren sorunlar için geçmiş algoritmadan daha iyi performans gösterir; aksi takdirde, gözlemlenen koşulların herhangi biri için tam Pareto cephesini uygulama yeterliliğine sahip değildir.

Feng vd. (1997), büyük ölçekli zaman maliyet ödünleşim problemlerine uyum sağlamada mevcut tekniklerin verimsizliğini sunarak, Holland’ın (1975) genetik algoritmasının (GA) kuralına bağlı olarak daha etkili bir model geliştirdi. Kendi modellerinde faaliyetlerin normal ve çarpışma seçenekleri ile ilgili bilgileri içeren iki kromozom kullandılar. Böylelikle, çözümlerin dışbükey gövdeye olan önemsiz uzaklıklarına göre objektif fonksiyon değerleri belirlenmiştir.

Colorni vd. (1994), Ng ve vd. (2008), karınca kolonisi optimizasyonunu (ACO) kullanarak çok amaçlı zaman maliyet ödünleşim problemini inceleyen ilk kişilerdi. Uygunluk işlevi çözümlerini değerlendirmek için değiştirilmiş uyarlanabilir ağırlık yaklaşımını (MAWA) desteklediler. Algoritmalarının büyüklüğü, Elbeltagi vd. (2005) tarafından incelenen diğer açıklayıcı stratejilere karşı önceden test edilmiştir. Sonuç, uygulanan Akut koroner sendrom (ACS) algoritmasının, önemli ölçüde az hesaplama varlıklarına ihtiyaç duyan zaman maliyet ödünleşim probleminin üstesinden gelmek için tatmin edici bir çözüm sağladığını ortaya koymaktadır.

Zhang vd., (2004) Tarafından sunulan karınca kolonisi algoritmasıyla (ACO) değiştirilmiş, uyarlanabilir ağırlık yaklaşımını (MAWA) entegre etmek için, Xiong ve Kuang, (2008) başka bir teknik denemiştir. Bu teknik, olası seçimlere karar vermek için,



yapılan iki seçeneğe dayanmaktadır. Gelişigüzel bir değişkenin kaydına göre, ana belirleme, bir maksimizasyon kriterine göre oluşturulur ve alternatif bir olasılık dağılım fonksiyonunu içerir.

Elbeltagi vd., (2007) karmaşık büyük ölçekli sorunlara ilgi gösteren güçlü bir algoritma sunmak için, karışık kurbağa leaping (SFL) algoritmasını inceledi. Sayısal problemleri, Visual Basic, Microsoft Project ve Microsoft Excel programları kullanılarak değiştirilmiş SFL (MSFL) uygulanarak değerlendirmiş. MSFL, temel SFL ve GA algoritmalarıyla karşılaştırılır ve büyük ölçekli sorunların çözümünde verimliliğini belirtmiştir.

Anagnostopoulos ve Kotsikas (2010), düğüm üzerinde faaliyet (AoN) şemalama yöntemini kullanan, benzetilmiş bir tavlama algoritmasının beş formunu analiz etmektedir. Anagnostopoulos ve Kotsikas (2010), çözülmüş malzemelerin tavlama prosedürüne neredeyse eşdeğer bir strateji uygulamayı uyguladılar. Bununla birlikte, birçok sorun faktörünün oluşturduğu çözümlerin kalitesini ve etkililiğini ölçmek için varians analizi (ANOVA) ve Duncan çoklu aralık testi (DMRT) kullanırlar. Test örnek setleri, Visual Basic programlama dilinde kodlanmış SA algoritmaları için RanGen2 programı kullanılarak rastgele üretilir. Sonuç olarak, SA varyasyonlarını Duncan testinin sonuçlarına göre, genel olarak avantajlı ve en dikkat çekici kötü algoritmalar için optimum çözümün son kesinlik aralığına göre sıraladılar.

Ayrık zaman maliyet ödünleşim problemi bir grafik olarak Afshar vd. (2009) tarafından sunulmuştur. Karınca kolonisine dayalı çok amaçlı optimizasyon modelini geliştirirler. Karınca kolonisindeki her çözüm, zaman maliyet ödünleşim problemlerini bir amacını keşfetmiştir. Karınca kolonisinden gelen bilgileri birleştirerek, zaman maliyet ödünleşim sorunlarını çözmek için, çok kolonili ve baskın olmayan bir arşivleme, Karınca kolonisi optimizasyon algoritması (ACO) kuruldu. Modelin etkinliği (Feng vd. 1997) ve Zheng vd.'dan (2005) elde edilen sonuçlar açısından incelenmiştir. Model, karşılaştırılan algoritmaları önemli ölçüde geliştirmiştir.

Yang, (2007b) zaman maliyet ödünleşim problemlerinin çökme seçeneklerini incelemek için, PSO algoritmasının uygulanmasını analiz etmiştir. Yang'ın (2007b) modeli, yöneticilerin daha fazla "eğer" (If) analizi yapmalarına yardımcı olmak için, pareto ön çözümünün çıktısını almaktır. Bu modelin kodlaması MATLAB optimizasyonu ile gerçekleştirilerek, sayısal bir simülasyona ve gerçek hayattaki bir otoyol restorasyon projesine ilerletilir. Araştırma, 8 faaliyetli bir ağın sayısal bir örneğini içerir ve vaka



çalışması 28 faaliyeti içerir. Önerilen algoritmanın performansını değerlendirmek için uygun parametrelerin benimsenmesiyle birlikte, on koşu başına ortalama yüzde sapma (APD) dikkate alınır. Sonuçta, PSO algoritmasının etkinliği ihmal edilebilir bir yüzde sapmayı karşılayarak onaylanmıştır.

Meyer ve Shaffer (1963), karma tamsayı programlama kullanarak, zaman ve maliyet arasındaki hem doğrusal hem de kesikli ilişkiyi dikkate alarak zaman maliyet ödünleşim problemini çözmüşlerdir. Aksi takdirde, tamsayı programlama, faaliyeti bitirmek için seçeneklerin sayısı yüksek olduğunda, çok fazla işlem süresi gerektirirdi.

Zaman ve maliyet arasında ödünleşim yapmak için bir proje planlarken, kalite gibi başka bir norm eklemek de mümkündür. Üçüncü bir hedef olarak kaliteyi zaman maliyet ödünleşim problemi ile birleştirmek, zaman-maliyet-kalite ödünleşimi olarak bilinen yeni bir problemten vazgeçer. [(Zhang ve Xing, 2010), (Babu ve Suresh, 1996), (Khang ve Myint, 1999), (Tareghian ve Taheri, 2006), (Kimet vd., 2012), (Mungle vd., 2013), (Tavana vd., 2014) ve (Monghasemi vd., 2015)], modellerini kullanarak yeni ödünleşim problemini değerlendirdiler.

Ayrıca, kaynakların mevcudiyetini hesaba katarak, [(Hegazy,1999), (Liu ve Wang, 2008), (Ghoddousiet vd., 2013), (Afruzi vd., 2014) ve (Rostami vd., 2014)] zaman maliyet ödünleşim sınırlı kaynakla çözdü.

Sönmez ve Bettemir (2012), ayrık zaman maliyet ödünleşim problemi (DTCTP) analizi için hibrit yöntemleri araştırdı. DTCTP problemi için genetik algoritma (GA), hibrit meta sezgisel (HMH), benzetilmiş tavlama (SA), kuantum simülasyonlu tavlama (QSA) ve hibrit algoritma (HA) gibi çeşitli yöntemler kullandılar. Hibrit algoritma (HA), 18 ila 630 aktivite içeren on kıyaslama optimizasyon problemi için kullanılmış. Çeşitli yöntemlerle elde edilen sonuçları eşleştirdiler ve SA ve QSA kullanmanın GA'nın yakınsamasını geliştirirken, Hibrit algoritmanın (HA) ayrık zaman maliyet ödünleşim problemi (DTCTP) performansını artırdığını bulmuşlar.

Bettemir ve Birgönül (2016), ayrık zaman maliyet ödünleşim problemine bir çözüm bulmak için araştırmalarını minimum maliyet eğimi yöntemine dayandırmışlar. Bunu ayrı zaman maliyet ödünleşim problemi için ilerlettiler; hızlandırma modları da ayrık olduğu için, maliyet fonksiyonundaki doğrusallığa zarar vermişler. Bu nedenle minimum maliyet eğimi yönteminin kullanılması, ayrık zaman maliyet ödünleşim problemi için uygun değildir.



Abdel-Raheem ve Khalafallah (2011), bir elektrik devresinin dallarındaki elektrik akımı akışının simülasyonuna dayanan “Electimize” adlı modern bir evrimsel algoritmanın geliştirilmesi üzerine çalıştı. Araştırmalarının önemli nedeni, inşaat endüstrisine mevcut evrimsel algoritmaların bazı eksikliklerinin üstesinden gelen sağlam bir optimizasyon aracı sağlamaktır.

Baykal (2006), ayrık zaman maliyet alternatifleri için son tarih problemiyle başlar. Dal ve sınır algoritması ve birkaç sezgisel işlem önermiştir. Tüm işlemler, problemin doğrusal programlama gevşemelerine bağlıdır. Doğrusal programlama gevşemesinin özellikleri belirlenir ve önerilen algoritmaların tasarımında uygulanır. Daha sonra ayrık zaman maliyet eğrisi problemi dikkate alınır. Bu yaklaşım, son tarih problemleri için başarılı bir çözümdür.

Aminbakhsh (2013), değiştirilmiş SAM yönteminin faydalarını PSO algoritması ile karıştıran bir hibrit-PSO modeli yarattı. İhtiyaç duyulan algoritmaların birleştirme işlemi Microsoft Visual Studio 2010 kullanılarak C ++ programlama dili ile yapılmıştır. PSO optimize edicinin potansiyelini onaylamak için, literatürden alınan kıyaslama optimizasyon problemleri önerilen algoritma kullanılarak çözülmüş ve önceki modellerle oluşturulan sonuçlar için karşılaştırma sunulmuştur. Ek olarak, AIMMS optimizasyon yazılımını kullanarak tamsayı programlamayı birleştirmek, performansı incelemek için örnek problemlerin tüm optimal çözümlerini keşfetmek için kullanılır. Sonuçlar, önerilen algoritmaların daha önce önerilen modelleri daha iyi olduğunu göstermektedir.

Deb (2001), ayrık zaman maliyet ödünleşim problemi için herhangi bir kesin çözüm algoritmasının, üstel zayıf bir senaryo sıkıntısını tamamen göstereceğini açıklamış; hesaplanan zamanın, problemin sayısı arttıkça üstel bir şekilde artacağı anlamına gelir. (Deb, 2001), (Feng vd., 1997), (Eshtehardian vd., 2008) ve (Afshar vd., 2009) dışbükey olmayan çözüm alanlarında kesin algoritmaların muhit optimizasyonunda sıkışıp kaldığı sonucuna varılmıştır. Dahası, sezgisel algoritmaları kullanan diğer araştırmacılar, bunların kesin prosedürlerle aynı olduklarını, ancak büyük ölçekli problemlerle etkili bir şekilde baş edemediklerini kabul ederler (Siemens,1971). Son olarak, mevcut meta-sezgisel algoritmaların ana kusurları, yerel optimizasyonda saplanma olasılığı olarak sonuçlandırılır (Zheng vd., 2005, Sönmez ve Bettemir, 2012).

Eirgash (2018) tarafından zaman maliyet ödünleşim problemini çözmek için kullanılan öğrenme tabanlı optimizasyon algoritması, eğitiminin tatmin edici tutumunu göstermiştir. Modelin NDS aşaması aracılığıyla kurulan çözümlerin, modelin MAWA



aşamasına göre mantıksal olarak iyi bir uyuma sahip olduğu, ayrıca NDS-TLBO algoritmasının kısmi rastgele ilk nüfus tabanlı sürümünün her ikisini de iyi olduğunu görülmüştür. Sonuç olarak, kullanılan algoritmanın önceki çalışmaların elde edilen sonuçları daha iyi olduğunu doğrulanmıştır.

Baskın olmayan sıralama II (NDS-II) birçok alanda çok amaçlı problemleri çözmek için kullanılmıştır. Örneğin su şebekeleri ve gecekondu alanları için planları yükseltmek için kullanıldı, ayrıca inşaatta enerji tüketimini ve sürdürülebilirliği optimize etmek için kullanıldı. Ek olarak, baskın olmayan sıralama II (NDS-II) şantiye işlemleri ve planlama sorunlarını halletmek için kullanılmıştır.

Fallah-Mehdipour vd. (2012), sırasıyla zaman-maliyet ve zaman-maliyet-kalite olmak üzere iki ödünleşim problemiyle başa çıkmak için NDS-II'yi uyguladı. Önerilen yöntemi doğrulamak için 18 ve 7 faaliyetli örnekler kullanıldı. Ayrıca çok amaçlı PSO uygulandı. Sonuçlar NDS-II'nin çok amaçlı PSO'ya tercih edildiğini onaylandı.

Su ağları ile ilgili olarak Creaco vd. (2014), bir su şebekesi iyileştirmesinin inşaat aşamalarını, olası iyileştirmelerin 100 yıllık planında su şebekesine yapılan iyileştirmelerin farklı aşamalarını göz önünde bulundurarak dört aşamada sınıflandırmıştır. Baskın olmayan sıralama II, iki hedef fonksiyon içinde en uygun çözümleri bulmak için altı ağ düğümü ve sekiz boru döşeme konumu modeliyle kullanıldı. Önerilen yaklaşım, yükseltme aşamaları boyunca su dağılımını ve basınç kalitesini korumak için en uygun çözümü sunarak tek aşamalı çalışmalardan daha iyi sonuçlar vermiştir.

Avrupa direktiflerinin yönergelerini dikkate alınarak baskın olmayan sıralama'nın değiştirilmiş bir versiyonunun kullanılmasıyla Hamdy vd. (2010), binaların enerji performansı için Avrupa direktiflerinin yönergelerini dikkate alarak, optimum maliyet ve sifıra yakın enerji inşa performansı için bir çözüm bulmuştur. Büyük kombinasyon sayısı nedeniyle çözüm alanı üç aşamaya bölünmüştür.

Yukarıdaki çalışmalarda belirtildiği gibi, baskın olmayan sıralama II (NDS-II) optimizasyon kaynak yönetimi, enerji ve çevre tasarımı ve proje çizelgeleme yeteneklerini doğruladı. Diğer genetik algoritmalara üstünlüğünün yanı sıra NDS-II, şemalama problemlerinde kullanılan MOPSO gibi bazı yöntemlerin de önüne geçmiştir.



2. YAPILAN ÇALIŞMALAR

Bu araştırmanın amacı, çok amaçlı baskın olmayan sıralama II (NDS-II) algoritmasını zaman maliyet ödünleşim problemlerini incelemek için uygulamaktır. Baskın olmayan sıralama II (NDS-II) algoritmasının teorik olarak belirlenmesi, zaman maliyet ödünleşimi optimizasyon problemlerine bir çözüm olarak basitleştirilerek, bu analizlerin zaman maliyetine özel bir önem kazandırılmıştır. Zaman maliyet ödünleşiminin ayarlanabilir bir örneğini oluşturmak için, proje zamanının ve maliyetinin amaç işlevini elde etmek için MATLAB’da kritik yol yöntemi kullanılmıştır. Proje toplam süresi, kritik yoldaki faaliyet süresinin toplamıdır. Bitişten-başlangıca (FS) mantıksal ilişkisini planlayan bir kritik yol metodu şemalaması elde etmek için, proje hedef süresi fonksiyonunu bulmak için bir yazılım kullanılmıştır.

Uzun yıllar optimizasyon yöntemi olarak tek amaçlı optimizasyon yöntemleri kullanılmaktadır. Uygunluk fonksiyonundaki çoklu hedeflerin birleştirilmesi, son araştırma çalışmalarında ünlü hale gelmiştir. Günümüzde herhangi bir projenin zamanını ve toplam maliyetini en aza indirmek çok amaçlı işlevler olarak kabul edilmektedir. Zaman maliyet ödünleşim optimizasyon problemini çözmek için çok amaçlı bir optimizasyon yaklaşımı gereklidir. Bu çalışmada, zaman maliyet ödünleşim problemini çözmek için Jaya algoritması önerilmektedir. Bu algoritmanın temel kuralı, elde edilen çözümlerin en iyi adaya yaklaştırılarak ve en kötü adaydan uzaklaştırılarak oluşturulmasıdır.

Zaman maliyet ödünleşim problemlerini çözmek için, bu tezde baskın olmayan sıralama II (NDS-II) kavramı Jaya yaklaşımı yardımıyla benimsenmiştir. Önceden, zaman maliyet ödünleşim problemlerini çözmek için baskın olmayan sıralama önemli ölçüde kullanılıyordu, ancak NDS-II olan güncellenmiş sürümde bazı dezavantajları çözüme ulaştırılmıştır. Günümüzde NDS-II, etkileyici sonuçları nedeniyle sıklıkla kullanılmaktadır. Baskın olmayan sıralama II (NDS-II) yaklaşımı, karar vericilerin deneyim ve bilgilerine bağlı olarak baskın olmayan çözümlerden tatmin edici bir çözüm arar. Pareto ön çözümü bulduktan sonra planlayıcılara ve karar vericilere verimli zaman maliyet kararları almalarında yardımcı olmaktadır. Pareto ön çözüm teorisi, optimizasyon ile ilgili birleşik bir standarda sahip olmayan çok amaçlı optimizasyondaki farklı çözümleri karşılaştırmak için genel olarak kabul edilen bir araçtır. Faaliyetlerin sayısı ve her faaliyet için seçim



seenekleriyle ilgili olarak, genellikle seimin birden fazla özümü vardır, ancak birbirinin lehine olmayan ve pareto özümleri olarak bilinen bir dizi özümü ierir.

2.1. Zaman Maliyet Ödünleşim Optimizasyonu

Ayrık zaman maliyet ödünleşim optimizasyon probleminin temel amacı, özel koşullar altında proje çizelgeleme için zaman ve maliyet arasında optimum bir denge sağlayan bir dizi zaman maliyet seimini tanımlamaktır. Zaman maliyet ödünleşim analizi, belirli bir son teslim tarihine sahip proje için veya programın gerisinde çalışan bir proje için proje son tarihini karşılamak için uygulanır. Daha önce bahsedildiği gibi, zaman maliyet ödünleşim problemleri genellikle bir projenin zamanının ve maliyetinin amacını bulmak için her faaliyet için uygun seeneklerin seilmesi ile uğraşır. Bir projenin hedef zamanı aşağıdaki denklemler kullanılarak bulunabilir.

$$ES_0 = 0 \text{ (0 alt simge sıfırı temsil eder)} \quad (1)$$

$$ES_j = \max \{EF_j\} \quad j=1, \dots, n+1 \quad (2)$$

$$EF_i = ES_i + t_i^{(k)} \quad i=0, \dots, n+1 \quad (3)$$

$$T = EF_{n+1} \quad (4)$$

T projenin toplam süresidir ve proje faaliyet ağının kritik yolunda olan kritik faaliyetlerin tüm süresi boyunca görünür. ES_j ve EF_j sırasıyla en erken başlangıç zamanı ve en erken bitiş zamanı; p_j hemen ardından gelen faaliyetin öncülü j ; $t_i^{(k)}$ faaliyet süresi i için k 'inci seenek, Faaliyet 0 ($n+1$) tek kukla faaliyettir.

Bir projenin toplam maliyeti, doğrudan maliyet ve dolaylı maliyetten oluşur. Bir proje ağındaki tüm faaliyetlerin doğrudan maliyetinin toplamı, doğrudan maliyeti verir. Bununla birlikte, dolaylı maliyet proje süresine bağlıdır.

Bu nedenle, bir projenin teslim tarihi uzarsa dolaylı maliyet artar. Aşağıdaki denklemler ile toplam proje maliyeti hesaplanabilir.

$$DC = \sum_{i=0}^{n+1} dc_i^{(k)} \quad (5)$$



$$IC = T \times ICR \quad (6)$$

$$C = DC + IC \quad (7)$$

DC'nin toplam doğrudan maliyeti olduğu ve IC'nin bir projenin dolaylı maliyetleri olduğu durumlarda, C proje toplam maliyetidir; $dc_i^{(k)}$ k'inci seçenek altındaki bir faaliyet I'nin doğrudan maliyetidir. ICR, bir projenin dolaylı maliyet oranıdır.

2.2. Baskın Olmayan Sıralama II (NDS-II)

Son yıllarda birçok çok amaçlı algoritma önerilmiştir. Bunun nedeni, tek seferde çoklu Pareto optimal çözümü bulma yetenekleridir. Pareto cephesinin yakınında veya yakınında yer alan çok sayıda alternatif çözüm sunan algoritma önemli bir değere sahiptir. Srinivas ve Deb (1995) tarafından önerilen baskın olmayan sıralama (NDS), ilk evrimsel algoritmalarından biridir.

Deb (2001) tarafından tanımlanan baskınlık kavramı; A tasarımı, en az bir kriterde daha iyiye ve diğer tüm hedeflerde daha kötü değilse, B tasarımına hâkim olur. Tasarımları baskınlık kavramına göre sıralama sürecine baskın olmayan sıralama (NDS) denir. Bir optimizasyon çalışmasının herhangi bir aşamasında, mevcut tasarımların bir popülasyonu arşivde tutulur. Her ilerlemede, popülasyondaki (veya arşivdeki) bazı diğer tasarımların hâkim olmadığı her uygulanabilir tasarıma 1 sıra verilir. Bunlar, popülasyonda sadece baskın olmayan tasarımlardır. Bu noktada, bu modeller doğru bir şekilde arşivden çıkarılır ve modellerin çoğu baskınlık için değerlendirilir. Geri kalan tasarımların hiçbirinin baskınlığında olmayanlara 2 sıralaması verilir. Yöntem, baskın olmayan tasarımlar elendikten sonra tasarımların geri kalanı yeniden konumlandırılarak daha düşük sıraları oluşturmak için tekrarlanır. Çalışma ilerledikçe, Pareto cephelerinde yeni tasarımlar baskın olacak ve diğer tasarımların yerini alacak. Sonuç, düzenli olarak başka tasarımlardan etkilenmeyen ve Pareto'ya doğru yakınsayan değişkenlerin bir kombinasyonu olacaktır. Deb (2001) tarafından ilk olarak literatüre sunulan baskın olmayan sıralama yöntemi (NDS II), temelinde baskınlık hesabı, sıralama ve kalabalık mesafesi hesaplaması gibi bir takım hesaplama süreçlerinden oluşmaktadır.

Amaç fonksiyonların değerlerinden oluşan çözümler kümesinde herhangi bir A ve B çözümleri karşılaştırıldığında eğer A çözümünün her bir değeri B çözümünün



değerlerinden iyi ise ve A'nın bütün değerleri B'dekileri eşit ya da daha iyi ise A çözümü B'ye göre daha baskındır denilir. Bu durum aşağıdaki matematiksel anlatımla ifade edilir.

$$P = \{A_1, \dots, A_n\} \quad (8)$$

$$Q = \{B_1, \dots, B_n\} \quad (9)$$

Eğer her bir $p \leq q$ ve herhangi bir $p < q$ ise p q 'ya göre baskındır değilse q p 'ya göre baskındır.

Buna “n” toplam amaç fonksiyonları sayısıdır. Başlangıç aşamasında bütün çözümlerin başkanlık değeri “0” olarak alınır. Bir popülasyonda tüm bireylerle karşılaştırıldığında baskın olan birey ya da bireylerin baskınlık değeri “1” olarak alınır. Geri kalan diğer bireyler (baskın olmayan bireyler) tekrar kendi aralarında karşılaştırılıp eğer diğerlerine göre baskın olan birey ya da bireyler varsa baskınlık değeri “2” olarak alınır.

Bu şekilde devam ettirilen baskınlık sıralamasının herhangi bir adımındaki bireyler aynı baskınlık değerine sahip olduğunda ya da başka bir anlatımla hepsi birbirine göre baskın olduğunda baskınlık sıralaması tamamlanmış olur. Bu duruma göre bir popülasyonda tüm bireylerin baskınlık sıralaması değeri “1” olabilir. Baskınlık sıralaması değerine matematiksel ifade olarak rank ismi de verilebilir. Rankı 1 olan bireyler en üstün bir bireydir.

Bir popülasyonda baskınlık sıralaması değeri aynı olan bireylerin başka bir kritere göre daha sıralanması gerekir. Kalabalık mesafesi olarak adlandırılan bu hesaplamada, her bir amaç fonksiyonu için küçükten büyüğe göre bireyler sıralanır. Birinci ve sonuncu değere sahip olan bireylerin kalabalık mesafesi değeri sonsuz olarak alınır.

Aradaki bireyler için ise,

$$km_{(i,j)} = \frac{f_{(j+1)} - f_{(j-1)}}{f_{(1)} + f_{(n)}} \quad i = 2, \dots, n - 1 \quad (10)$$

eşitliği ile hesaplanır.

Buna km kalabalık mesafesi değerini, f amaç fonksiyonu değerlerini, n ise baskınlık sıralamasındaki birey sayısı değerini ifade eder. Örnek verecek olursak, rankı “2” olan



birey sayısı, bireyin her bir amaç fonksiyonu için aldığı kalabalık mesafesi değerleri toplamı o bireyin tüm amaç fonksiyonları için toplam kalabalık mesafesi değerini verir.

$$Tkm_{(i)} = \sum_{j=1}^m km_{(i,j)} \quad (11)$$

Burada m amaç fonksiyonu sayısı ve tkm birey için toplam kalabalık mesafesi değeridir.

Rank değerleri ve kalabalık mesafesi hesaplanan bir popülasyondaki tüm bireyler, önce rank değerleri ve sonrasında kalabalık mesafesi hesap değerleri dikkate alınarak nihai bir sıralamaya tabi tutulur. Böylece rank değeri “1” olanlar ilk sıralarda yer alır.

2.3. Jaya ile Optimum Çözüm

Jaya kısıtlı ve kısıtlı olmayan optimizasyon problemlerini çözmek için, başarıyla uygulanan basit ve güçlü bir küresel optimizasyon algoritmasıdır. Yine de bu algoritma yalnızca ortak kontrol parametrelerini gerektirir ve algoritmaya özgü herhangi bir kontrol parametresi gerektirmez. Belirli bir problem için en iyi çözüme doğru ilerleyerek ve en kötü çözümden kaçınarak çözümün elde edilebilmesi esasına dayanır. Bu algoritmanın güzelliği, maksimum nesil sayısı ve popülasyon boyutu gibi yalnızca birkaç kontrol parametresine ihtiyaç duymasındır.

Jaya, yanlış bir şekilde ayarlandığında kaçınılmaz ve istenmeyen yakınsamalara yol açabilecek olan hesaplama parametreleri gibi parametrelere ihtiyaç duymadan çalışabilen kullanışlı bir algoritmadır. Her yinelemede parametreleri kontrol etmek genellikle zordur. Ek olarak, algoritmaya özgü parametrelerin kontrolü o kadar da kolay değildir. Jaya algoritmasını diğerlerinden daha güçlü kılan en önemli neden muzaffer doğasıdır (Jaya, zafer anlamına gelen Sanskritçe (Sanskrit) bir kelimedir) ve herhangi bir hiper parametre içermemesidir (Rao, 2016).

Minimize edilecek veya maksimize edilecek amaç fonksiyonu $f(x)$ 'dir. Herhangi bir i iterasyonda, m tasarım değişken sayısı olduğunu varsayalım $j=1,2, \dots, m$, n aday çözüm sayısı $k=1,2, \dots, n$. Tüm aday çözümlerde, $f(x)$ 'den en iyi aday değeri $f(x)_{\text{en iyi}}$ olarak; yine tüm aday çözümlerde, $f(x)$ 'den en kötü aday değeri $f(x)_{\text{en kötü}}$ olarak elde edilir. Eğer $X_{j,k,i}$ j'inci değişkenin değeri ise k'inci aday için i'inci yineleme sırasında, Denklem (12)'de gösterildiği şekilde güncellenir.

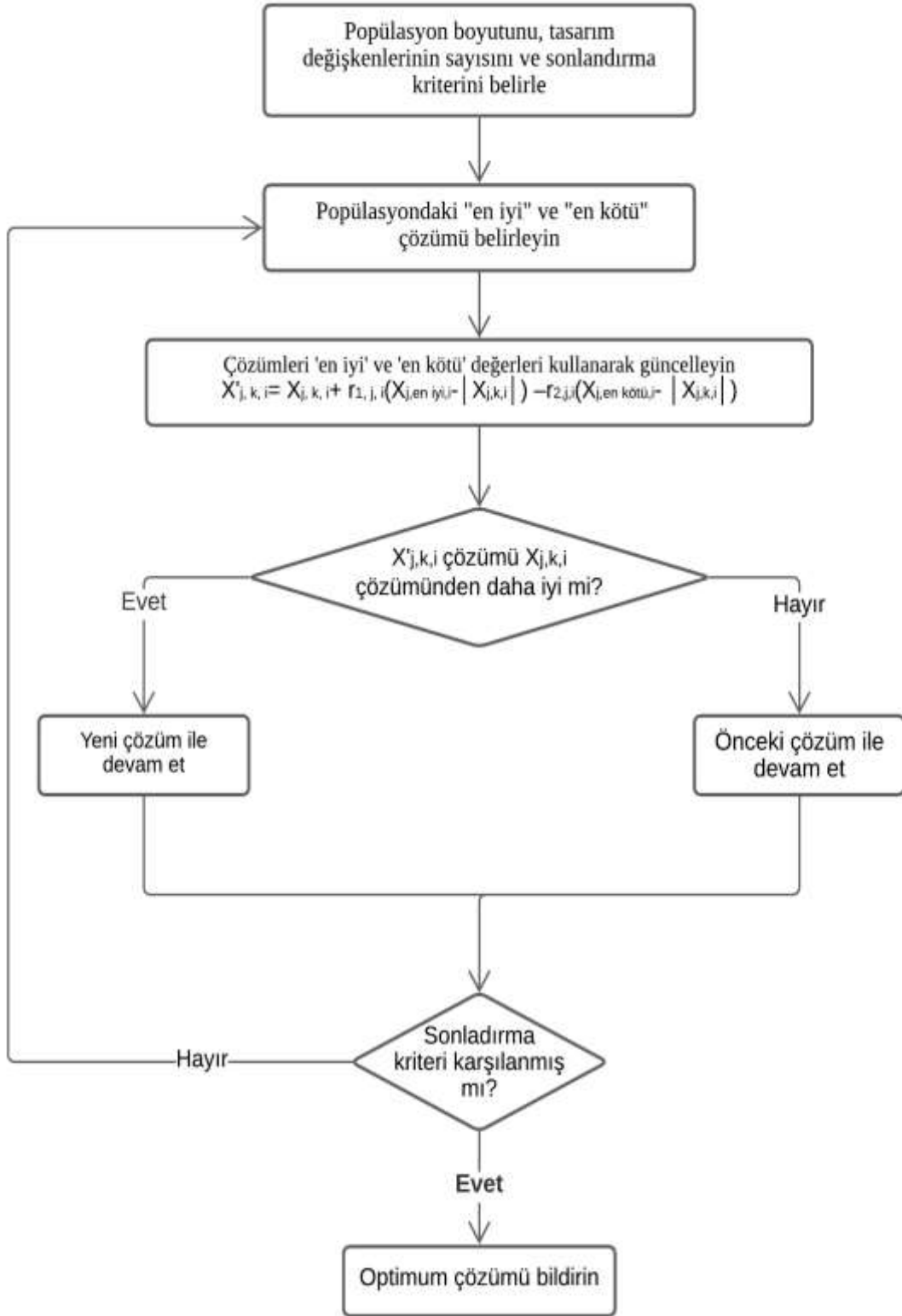


$$X'_{j,k,i} = X_{j,k,i} + r_{1,j,i} (X_{j,en\ iyi,i} - |X_{j,k,i}|) - r_{2,j,i} (X_{j,en\ kötü,i} - |X_{j,k,i}|) \quad (12)$$

$X_{j,en\ iyi,i}$ en iyi aday için j değişkeninin değeridir, $X_{j,en\ kötü,i}$ en kötü aday için j değişkeninin değeridir. $X'_{j,k,i}$ $X_{j,k,i}$ 'nin güncellenmiş değeridir; Ayrıca, $r_{1,j,i}$ ve $r_{2,j,i}$ j 'inci değer için iki rastgele sayılardır. i 'inci yineleme ve $[0, 1]$ aralığında " $r_{1,j,i} (X_{j,en\ iyi,i} - |X_{j,k,i}|)$ " terimi çözümün en iyi çözüme yaklaşma eğilimini gösterir ve " $r_{2,j,i} (X_{j,en\ kötü,i} - |X_{j,k,i}|)$ " çözümün en kötü çözümden kaçınma eğilimini gösterir. $X'_{j,k,i}$ daha iyi fonksiyon değeri veriyorsa kabul edilir. Yinelemenin sonunda kabul edilen tüm işlev değerleri korunur ve bu değerler bir sonraki yinelemenin girdisi olur.

Şekil 2.1, Jaya algoritmasının akış şemasını göstermektedir. Jaya algoritması her zaman en iyi çözüme ulaşmaya ve en kötü çözümden uzaklaşmaya çalışır. Algoritma en iyi çözüme ulaşarak galip gelmeye çalışır ve dolayısıyla Jaya olarak adlandırılır.





Şekil 2.1. Jaya algoritmasının akış şeması

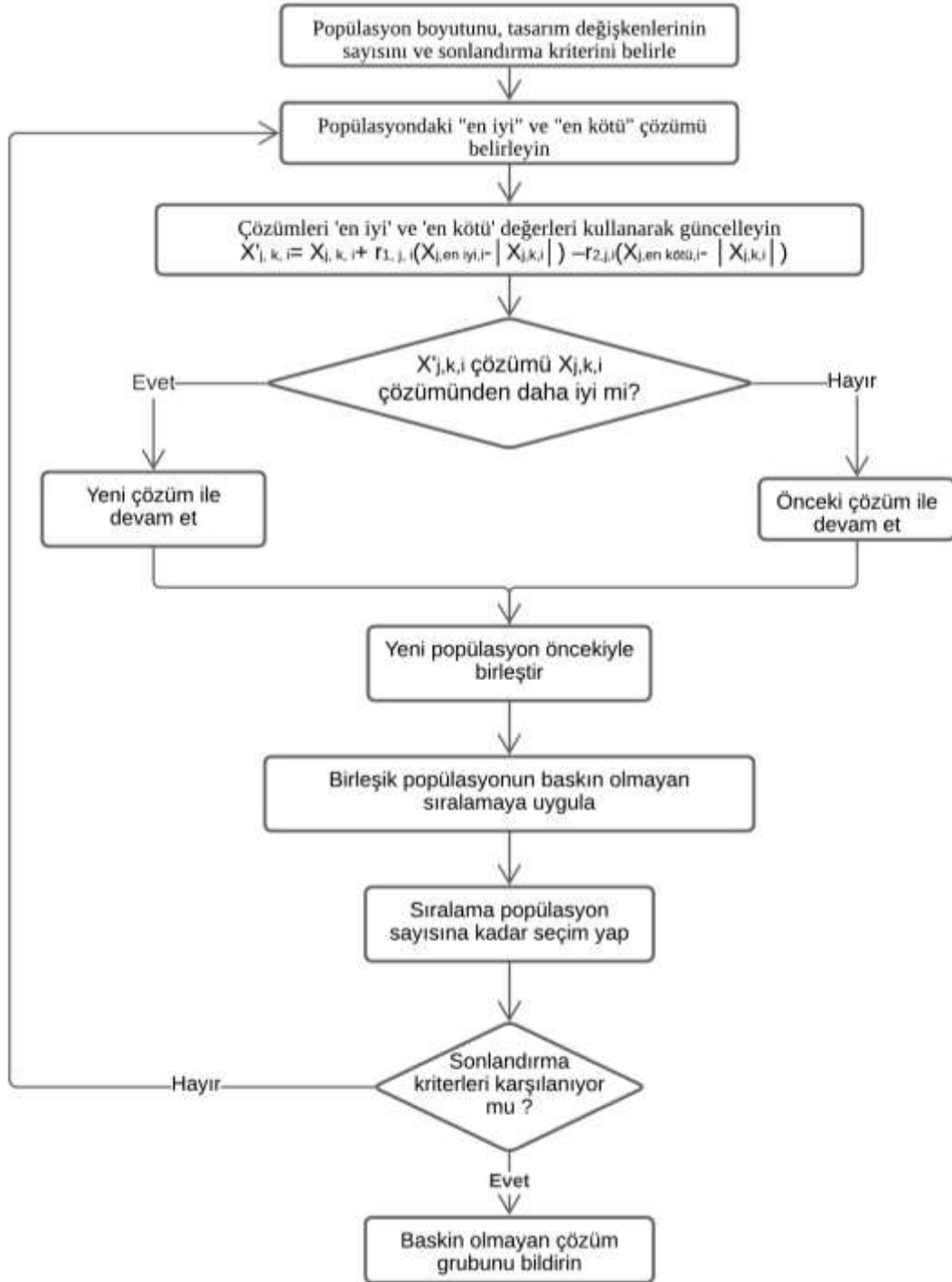


2.4. Zaman Maliyet Ödünleşim Problemleri İçin Baskın Olmayan Sıralama II – Jaya Algoritması

Baskın olmayan sıralama II (NDS II)- Jaya algoritması, çok amaçlı optimizasyon problemlerini çözmek için uygundur. NDS II- Jaya algoritması, en kötü çözümden kaçınarak en iyi çözüme doğru ilerlemeye devam etmeyi amaçlar. Hedefleri etkili ve verimli bir şekilde ele almak için Jaya algoritması, (Deb vd., 2001) tarafından önerilen baskın olmayan sıralama II yaklaşımı ile ilişkilendirilir. Baskın olmayan sıralama II (NDS II)- Jaya algoritması, çok amaçlı optimizasyon problemlerini çözmek için baskın olmayan sıralama II yaklaşımının ve Jaya algoritmasının dikkate değer özelliklerini içerir.

Baskın olmayan sıralama II (NDS II)- Jaya algoritması sürecinin akış şeması Şekil 2.2'de sunulmuştur.





Şekil 2.2. NDS II- Jaya algoritmasının akış şeması.



3. BULGULAR

Baskın olmayan sıralama II-Jaya (NDSII-Jaya) algoritmasının performansını göstermek için algoritma birkaç örnek üzerinde incelenecektir. Önerilen modeli doğrulamak için incelenen örnekler daha önce birçok araştırmacı tarafından çözülmüştür. Kullanılan algoritmanın performansı değerlendirmesini göstermek için küçük ve orta ölçekli örneklerin yanı sıra büyük ölçekli örnekler de değerlendirilecektir.

Literatür araştırmalarına dayanarak, kullanılan algoritmanın performansını doğrulamak için en çok uygulanan 7 ve 18 faaliyetli problemler de çözülmüştür. Bu problemler, baskın olmayan sıralama II (NDS II) yaklaşımları ile birleştirilen çeşitli meta-sezgisel algoritmalar kullanan yaygın çalışmalarda kullanılmıştır. Sonuçlar başlangıçta Zheng vd., (2005), Ng vd., (2008) tarafından geliştirilen modellerle karşılaştırılmıştır. 63 faaliyet örneğinin baskın olmayan sıralama II -Jaya kullanılarak değerlendirilmesinde elde edilen sonuçlar, NDS-GA, NDS-ACO ve NDS-PSO Bettemir (2009) modelleri aracılığıyla elde edilen çözümlerle karşılaştırılacaktır. Ek olarak, 630 aktivite inşaat projesi örneği NDS-TLBO kullanılarak Eirgash, (2018) tarafından bulunan sonuçlara karşı test edilir. Baskın olmayan sıralama II-Jaya, birçok araştırmada kullanılan tüm algoritmalar nedeniyle birkaç sonuç verir, kullanılan model, karar vericilerin nihai en iyi çözümü belirlemesini gerektirmektedir.

Doğrulama süresi boyunca, örnek problemlerden herhangi birinin analizi için on koşu uygulanmıştır. Uygulanan tüm zaman maliyet ödünleşim problemlerinin detayları, seçilen parametre değerleri ve baskın olmayan sıralama II-Jaya algoritmaları için sayısal simülasyonların sonuçları devam eden bölümde sunulmaktadır.

3.1. İnşaat Projelerinde Zaman Maliyet Ödünleşim Problemleri İçin Baskın Olmayan Sıralama II-Jaya Uygulaması

Bir dizi Pareto ön çözüm bulmak için, Jaya ile birleştirilmiş baskın olmayan sıralama II (NDS-II) tabanlı çok amaçlı bir optimizasyon modeli önerilir. NDSII-Jaya'nın performansını sunmak için 7, 18, 63, 630 faaliyetli değişen önceki araştırmalardan alınan inşaat projelerinden örnekler incelenir. Sonuçlar, optimum veya optimuma yakın çözümler dikkate alınarak önceden önerilen modeller kullanılarak elde edilenlerle karşılaştırılır.



NDSII-Jaya algoritmasının inşaat yönetimi alanındaki zaman maliyet ödünleşim problemleri için etkili bir şekilde çalıştığı sonucuna varılmıştır. Literatürden en çok alınan problemler 18 ve 63 faaliyet içeren projelerdir. Önceki zaman maliyet ödünleşim problemlerin araştırmasının Hegazy (1999), Zheng vd. (2004), Elbeltagi, vd. (2007), Afshar vd. (2009), Zhang ve Xing (2010), Meyer ve Shaffer (1963), Yang (2007b), büyük kısmı, önerilen meta-sezgiselliğin verimliliğini değerlendirmek için 18 faaliyete kadar olan küçük örnek problemleri kullanıldı. Ayrıca 63 faaliyetli projeler Sönmez ve Bettemir (2012), Aminbakhsh (2013) tarafından kullanılmıştır. Baskın olmayan sıralama II-Jaya algoritması, zaman maliyet ödünleşimini en aza indirmek için daha karmaşık bir sorunun çözümü üzerinde de test edilmiştir. Böylelikle, Bettemir'den (2009) alınan ve Eirgash (2018) tarafından kullanılan 63 faaliyetli projenin örnek problemi de bu çalışmada önerilen model ile çözülmüştür. Elde edilen sonuçlar, önceki meta-sezgisel algoritmalar tarafından bildirilen çözümleri karşılaştıran önerilen algoritmanın etkinliğini göstermektedir.

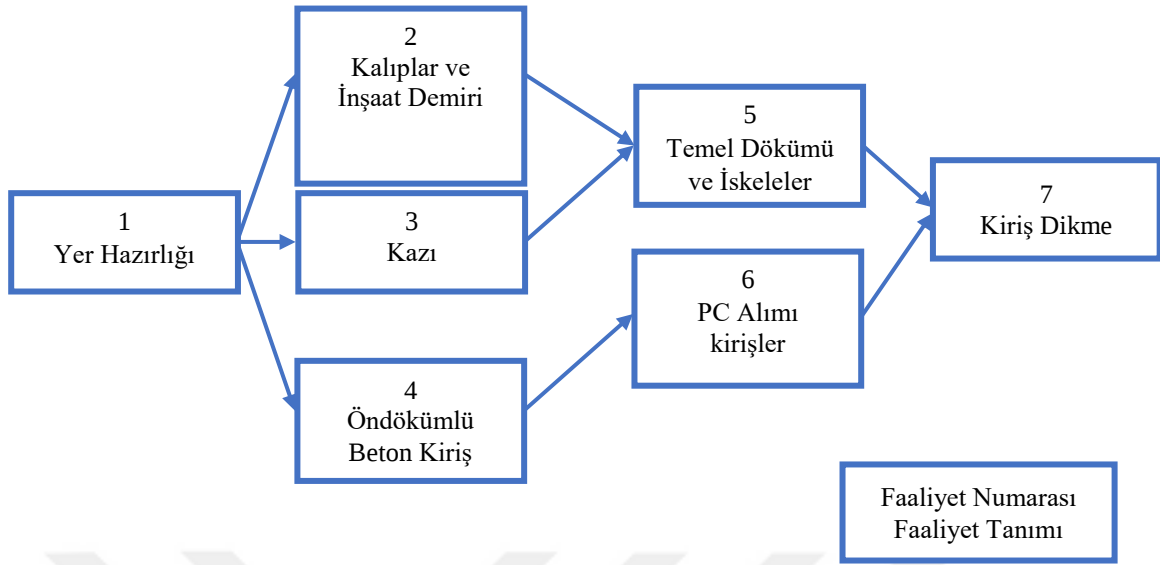
3.2. Çeşitli Faaliyetler için Sayısal Örnekler

Zaman maliyet ödünleşim problemleri, pareto ön çözümlerini elde etmede kullanılan baskın olmayan sıralama II-Jaya modelinin performansını ortaya çıkarmak için, teknik literatürden alınan küçük, orta ve büyük ölçekli problemler incelenmiştir. Kullanılan algoritma MATLAB üzerinde kodlanmıştır, koşullar Intel (7Gen) i7 ve 8GB RAM'e sahip bir kişisel bilgisayardan gerçekleştirilmiştir. Ardışık deneysel seri numarası tüm örnekler için 10 koşu olarak alınmıştır.

3.2.1.7 Faaliyetli Proje Örneği

Feng vd. (1997) tarafından tanıtilen ve Şekil 3.1'de gösterilen ağ, 3 ila 5 olası seçenek (mod) ile Bitişten-Başlangıca (FS) mantıksal ilişkisine sahip 7 faaliyet içerir. Dolaylı maliyet günlük 1500\$ olarak alınmıştır. Olası faaliyet seçenekleri, karşılık gelen süreler ve maliyetlerle birlikte Tablo 3.1'de sunulmuştur.





Şekil 3.1. 7 Faaliyetin ayarlanması

Tablo 3.1. 7 faaliyetli proje için (modlar) seçenekler

Faaliyet	Faaliyet Sayı	Öncül Faaliyet	Doğrudan maliyet (\$)	Süre (gün)	Mod
Yer hazırlığı	1	-	23000	14	1
			18000	20	2
			12000	24	3
Kalıplar ve inşaat demiri	2	1	3000	15	1
			2400	18	2
			1800	20	3
			1500	23	4
			1000	25	5
Kazı	3	1	4500	15	1
			4000	22	2
			3200	33	3
Öndökümlü beton kiriş	4	1	45000	12	1
			35000	16	2
			30000	20	3
Temel dökümü ve iskeleler	5	2,3	20000	22	1
			17500	24	2
			15000	28	3
			10000	30	4
PC alımı kirişler	6	4	40000	14	1
			32000	18	2
			18000	24	3
Kiriş dikme	7	5,6	30000	9	1
			24000	15	2
			22000	18	3



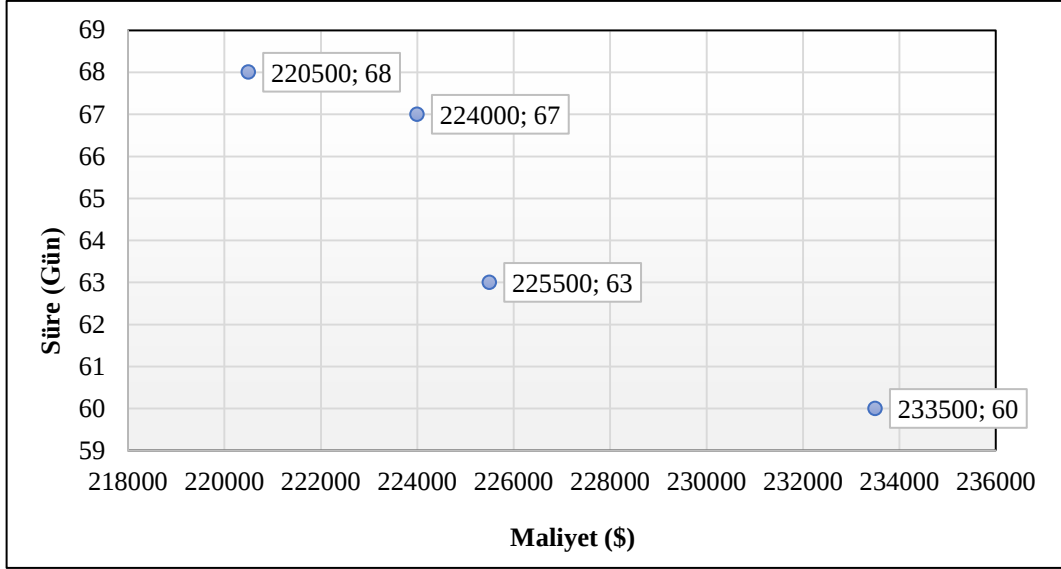
Baskın olmayan sıralama II-Jaya'nın sonuçları Tablo 3.2'de sunulmuştur ve 7 faaliyetli problem için önceki çözümlerle karşılaştırılmıştır. Gen ve Cheng (2000), Zheng vd. (2005) ve Magalhaes-Mendes (2015) tarafından elde edilen çözümler ile bu çalışmada NDS-II-Jaya kullanılarak elde edilenlerden daha üstün değildir. Bu yöntemde elde edilen sonuç, MAWA-GA ile elde edilenden daha az maliyet sağlar. Xiong vd. (2008) MAWA-ACO ve Eirgash (2018) tarafından MAWA-TLBO için ulaşılan Pareto ön çözümleri baskın olmayan sıralama II-Jaya yöntemiyle ulaşılan sonuçlarla aynıdır. NDSII-Jaya'nın önceki yöntemlerle karşılaştırılması, NDSII-Jaya'nın küçük ölçekli zaman maliyet ödünleşim problemleri için önceki önerilen algoritmalara kadar iyi çalıştığını belirtilebilirdi. Bu örnekte, popülasyon sayısı 50 ve iterasyon sayısı 100 olarak alınmış.

Tablo 3.2. 7 faaliyetli problem için karşılaştırma

Yazar	Yöntem	Kriterler		Hesaplama süresi
		Süre (Gün)	Maliyet (\$)	
(Gen ve Cheng, 2000)	MAWA-GA	79	256400	-
(Zheng vd., 2004)		66	236500	-
(Magalhaes-Mendes, 2015)		63	225500	5 saniye 50 nesil için
(Xiong vd., 2008)	MAWA-ACO	60	233500	-
		62	233000	
		63	225500	
		67	224000	
		68	220500	
(Eirgash, 2018)	MAWA-TLBO	60	233000	1 saniye 10 nesil için
		62	225500	
		63	227500	
		66	224000	
		67	220500	
		68	233500	
Bu çalışma	NDS-II-Jaya	60	233500	1 saniye 10 nesil için
		63	227400	
		67	224000	
		68	220500	

NDS-II-Jaya kullanılarak elde edilen sonuçların (Pareto ön çözümleri) grafiksel gösterimi Şekil 3.2'te gösterilmektedir.





Şekil 3.2. NDS-II-Jaya tarafından 7 faaliyetli problem için ulaşılan Pareto optimal çözümü

Tablo 3.3 önerilen modlardan seçilen en iyi modu göstermektedir, örnek olara ilk pareto için 6'inci faaliyet hariç diğer faaliyetler için birinci mod seçilmektedir. İkinci pareto için 4'üncü ve 6'inci faaliyet için 2'nci mod seçilmiştir, diğer faaliyetler için birinci mod seçilmektedir.

Tablo 3.3. NDS-II-Jaya kullanılarak 7 faaliyetli problem için seçilen seçenekler

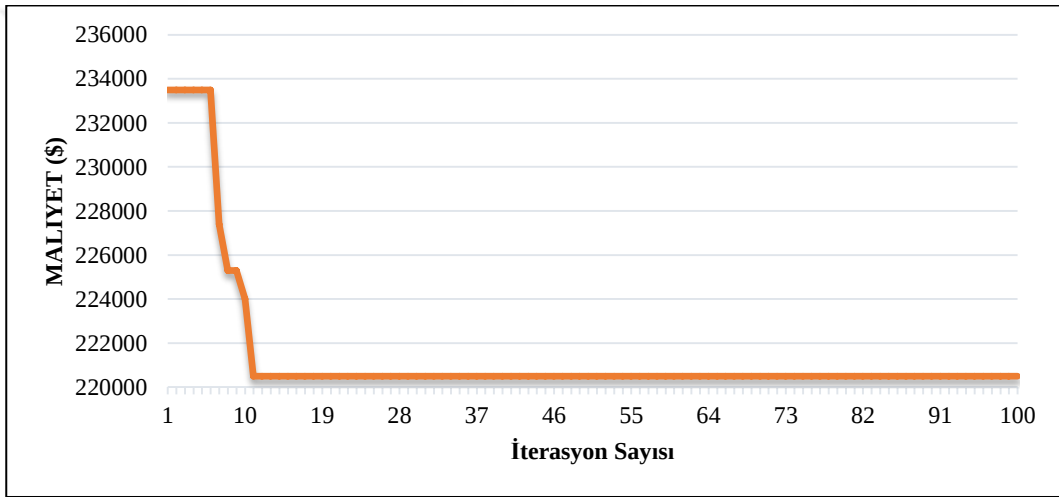
Pareto ön çözümleri	Süre (gün)	Maliyeti (\$)	En iyi modlar seçildi						
			1	2	3	4	5	6	7
1	60	233500	1	1	1	1	1	2	1
2	63	225500	1	1	1	2	1	2	1
3	67	224000	1	2	1	3	2	3	1
4	68	220500	1	4	1	3	1	2	1

Zaman maliyet optimizasyonu inşaat projesinin zamanını ve maliyetini düşürmekte ve gecikmeleri etkilemektedir. Normal uzunluğu bulmak için ileriye dönük planlama kullanılarak, ilk olarak projenin kritik yolu ölçüldü ve bunun için seçilen seçenekler normal süre ve normal maliyetti, proje süresi 105 gün ve 253700 \$ idi. Zaman maliyet optimizasyonu modelini kullandıktan sonra maksimum zaman değeri 68 gündü ve maliyet 220.500 dolardı. Optimize edilmiş fiyatlar kullanılarak toplam giderden %15 ve proje

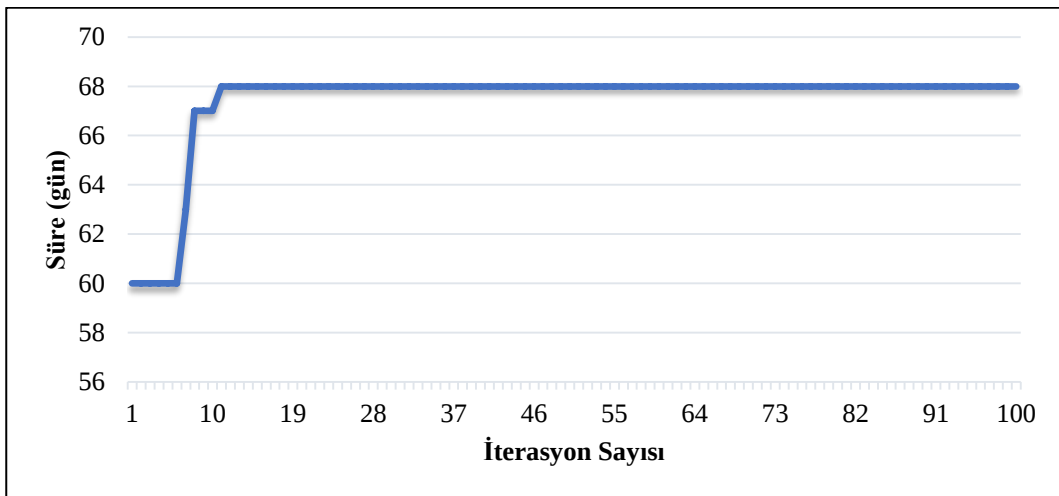


süresinden %54,4 tasarruf edildi. Baskın olmayan sıralama II-Jaya, 5050 olası farklı programı araştırdı. Bu nedenle fonksiyon değerlendirme sayısı 5050 ($NFE=100 \times 50 + 50$) olarak alınabilir.

Aşağıdaki şekiller zaman, maliyet ve iterasyon sayısı arasındaki yakınsamayı temsil etmektedir. Baskın olmayan sıralama II- Jaya algoritmasını 10 iterasyon sayısında en uygun veya en yakın çözümü verdiğini şekillerde göstermektedir. Kullanılan iterasyon sayısı biraz daha büyük olduğu ve 10 iterasyon sonra gereksiz yere çalıştığı anlamına gelir. Bu nedenle, zaman maliyet ödünleşim problemi için iterasyon sayısı en uygun çözümden emin olmak için 20 olarak alınabilir.

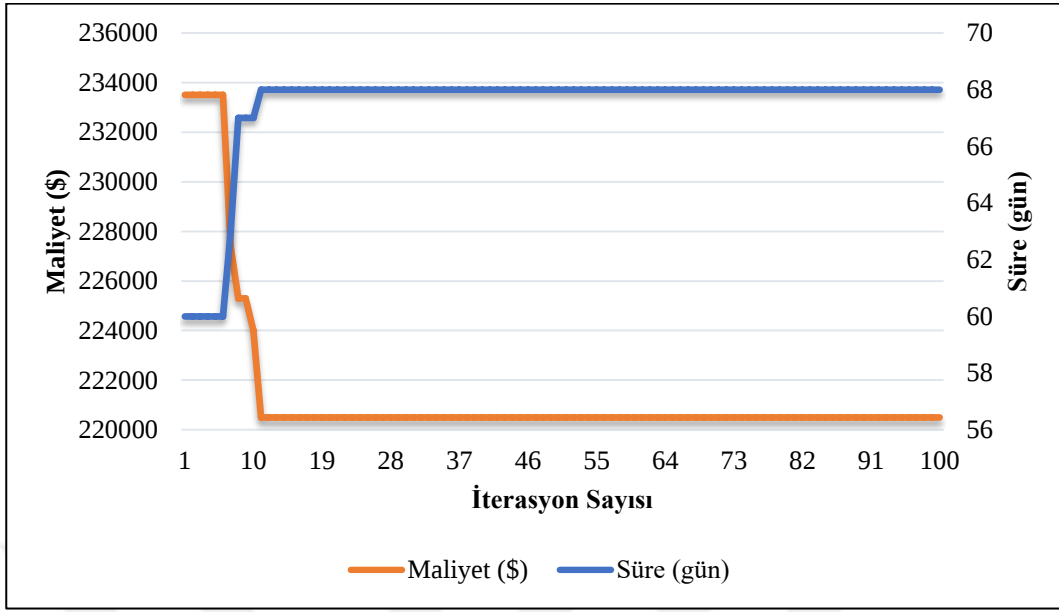


Şekil 3.3. NDSII-Jaya kullanarak 7 faaliyet için maliyet ve iterasyon sayısı yakınsaması



Şekil 3.4. NDSII - Jaya kullanarak 7 faaliyet için süre ve iterasyon sayısı yakınsaması



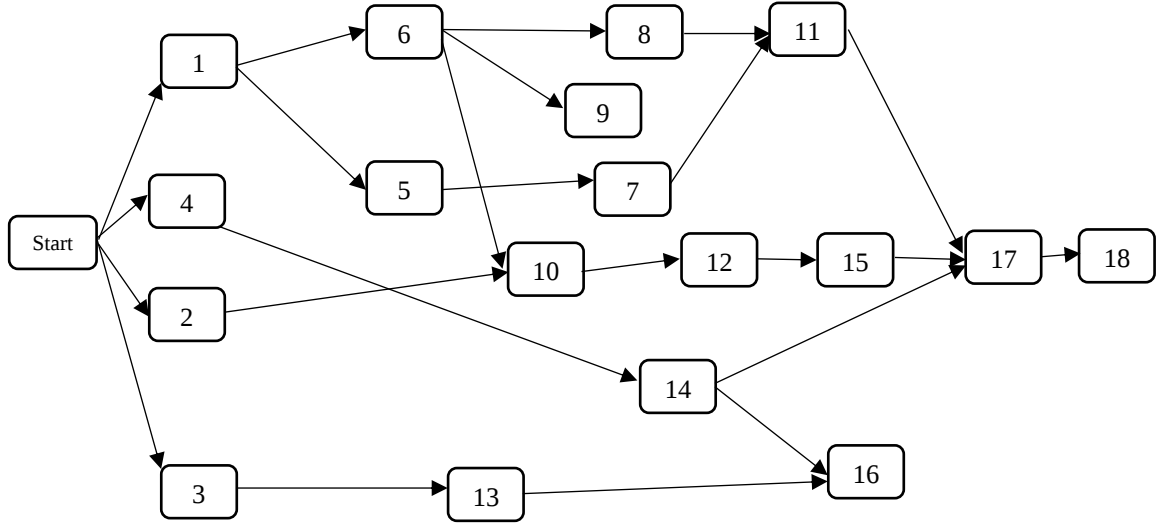


Şekil 3.5. NDSII - Jaya kullanarak 7 faaliyet için süre, maliyet ve iterasyon sayısı yakınsaması

3.2.2. Beş Modlu 18 Faaliyetli Proje Örneği

Bir çalışma örneği, orijinal olarak Feng vd. (1997) tarafından tanıtılan on sekiz faaliyetten oluşan bir projedir. Mantıksal Bistişten-Başlangıça (FS) ilişkisine sahip ağ Şekil 3.3'te gösterilmektedir. Tablo 3.4, model proje için faaliyet ilişkilerini, her faaliyet için beş modu ve bunlarla ilgili zaman ve maliyeti göstermektedir. Bu problemde uygulanan dolaylı maliyet oranı günlük 1500 \$ 'dır.





Şekil 3.6. 18-faaliyetine örnek projesi için faaliyet ilişkileri

Tablo 3.4. Beş modlu 18 faaliyetli proje için seçenekler

Faaliyet Numarası	Öncül Faaliyet	Mod 1		Mod 2		Mod 3		Mod 4		Mod 5	
		Doğrudan Maliyet(\$)	Süre (gün)	Doğrudan Maliyet(\$)	Süre (gün)	Doğrudan Maliyet(\$)	Süre (gün)	Doğrudan Maliyet(\$)	Süre (gün)	Doğrudan Maliyet(\$)	Süre (gün)
1	-	2400	14	2150	15	2400	16	1500	21	1200	24
2	-	3000	15	2400	18	1900	20	1500	23	1000	25
3	-	4500	15	4000	22	1800	33				
4	-	45000	12	35000	16	3200	20				
5	1	20000	22	17500	24	30000	28	10000	30		
6	1	40000	14	32000	18	15000	24				
7	5	30000	9	24000	15	18000	18				
8	6	220	14	21	15	22000	16		21		24
9	6	300	15	240	18	200	20	208	23	120	25
10	2,6	450	15	400	22	180	33	150		100	
11	7,8	450	12	350	16	320	20				
12	5,9,10	2000	22	1750	24	1500	28		30		
13	3	4000	14	3200	18	1800	24				
14	4,10	3000	9	2400	15	2200	18				
15	12	4500	12	3500	16						
16	13,14	3000	20	2000	22	1750	24	1500	28	1000	30
17	11,14,15	4000	14	3200	18	1800	24			1200	
18	16,17	3000	9	2400	15	2200	18			1000	

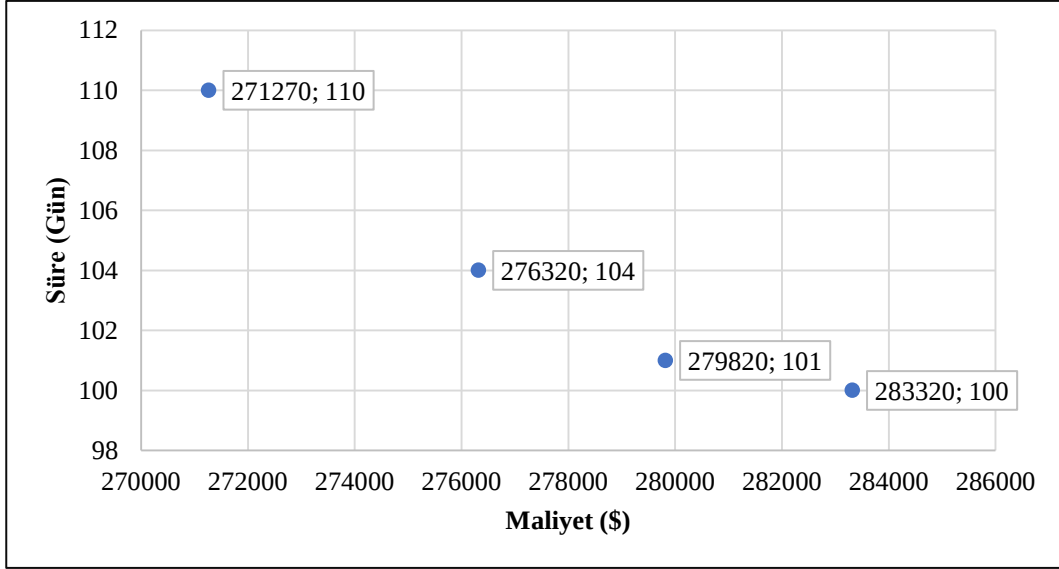


Baskın olmayan sıralama II-Jaya algoritması, MAWA-GA tabanlı zaman maliyet optimizasyon modeli (Zheng vd., 2005), MAWA-AS (Afshar vd., 2009) ve aynı projeyi kullanan MAWA-SGPU algoritması (Ng ve vd., 2008) arasında yapılan karşılaştırma Tablo 3.5'te gösterilmiştir. Tablo 3.5'te görülebileceği gibi, baskın olmayan sıralama II - Jaya tabanlı model, MAWA-GA ve MAWA-AS modellerine göre daha az popülasyon boyutu ve yineleme sayısı ile yürütülmüştür. Ayrıca NDS-II-Jaya tabanlı modelin aynı proje tamamlanma süresi ile daha optimum bir maliyet değeri sunduğu gözlemlenebilir. Örneğin 100 gün için baskın olmayan sıralama II-Jaya ile elde edilen çözüm maliyeti 283320\$ iken MAWA-GA model maliyeti 287720\$'dır. Bu 4400\$'lık bir tasarrufla sonuçlanır, bu da toplam maliyetin%1,50'sine karşılık gelir. MAWA-GA ve NDS-II-Jaya modelleri arasındaki ACS-SGPU ve AS-MAWA modelinde toplam maliyet 285400\$, 286670\$'dır. Ayrıca ACS-SGPU ve NDSII-Jaya yinelemeleri, MAWA-GA ve MAWA-AS'ninkilerden daha küçüktür. Çözümlerin kalitesine rağmen ACS-SGPU tarafından üretilen NDS-II-Jaya kadar iyi değildir, MAWA-GA ve MAWA-AS modellerinden üstündür ve daha iyi Pareto ön çözümleri üretebilir.

Tablo 3.5. NDS-II-Jaya kullanarak 5 modlu 18 faaliyetli problem için karşılaştırma

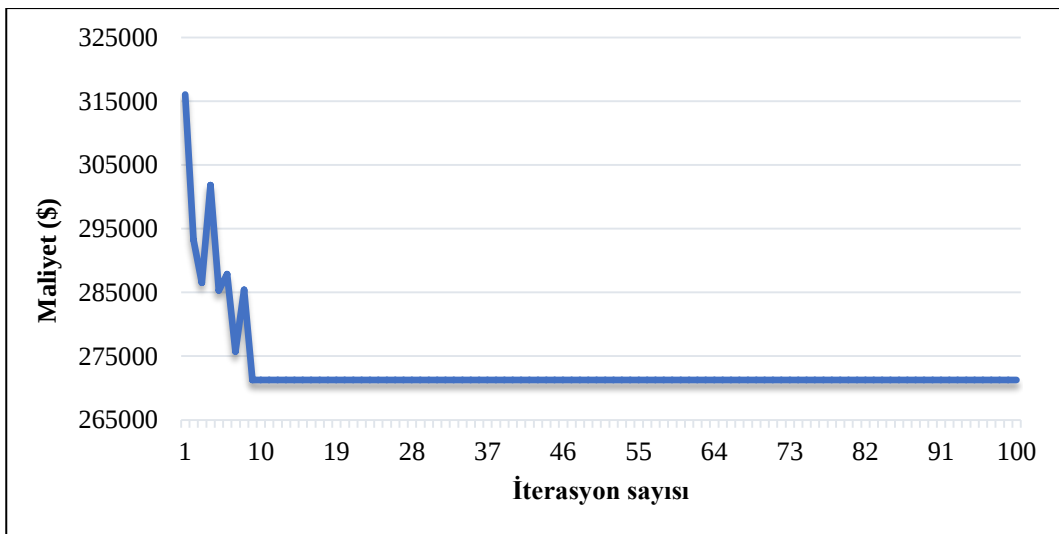
Yazar	Yöntem	Kriterler		Popülasyon hacımı	İterasyon sayısı
		Süre (Gün)	Maliyet (\$)		
(Ng ve vd., 2008)	MAWA-GA	100	287720	50	500
		101	284020		
		104	280020		
		110	273720		
(Zheng vd., 2005)	MAWA-ACS-SGPU	100	285400	10	200
		101	282508		
		104	277200		
		110	273165		
(Afshar vd. 2009)	MAWA-AS	100	286670	50	400
		101	281300		
		104	277265		
		110	272265		
Bu araştırma	NDS-II-Jaya	100	283320	50	100
		101	279820		
		104	276320		
		110	271270		





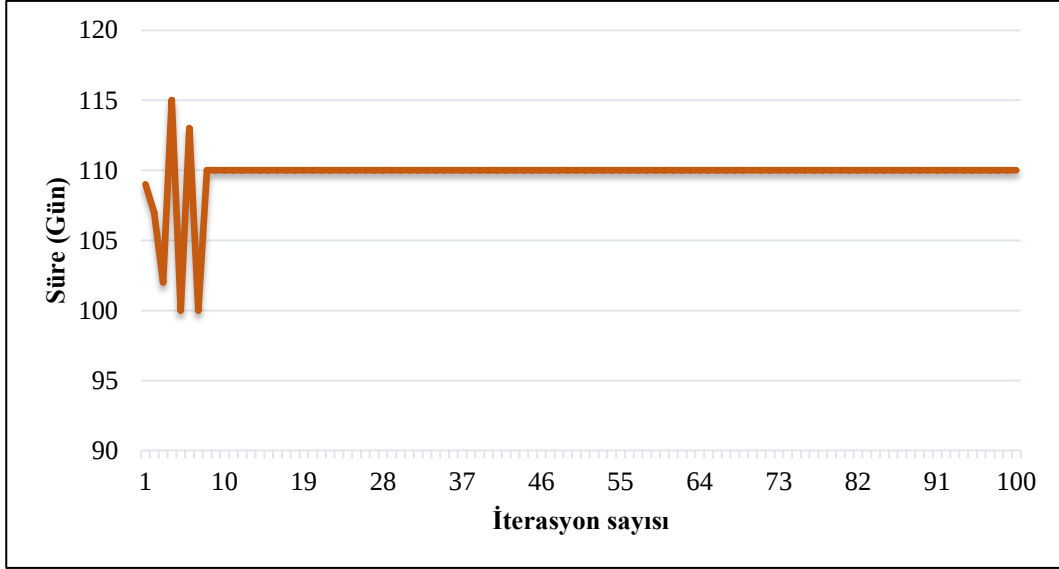
Şekil 3.7. NDS-II-Jaya tarafından 7 faaliyetli problem için ulaşılan Pareto optimal çözümü

Aşağıdaki şekiller zaman, maliyet ve iterasyon sayısına arasındaki yakınsamayı temsil etmektedir. Baskın olmayan sıralama II- Jaya algoritmasını 8'inci iterasyon sayısında en uygun veya en yakın çözümü verdiğini şekillerde göstermektedir. Kullanılan iterasyon sayısı biraz daha büyük olduğu ve 8'inci iterasyon sonra gereksiz yere çalıştığı anlamına gelir. Bu nedenle, kullanıldığı zaman-maliyet ödünleşim problemi için iterasyon sayısı en uygun çözümünden emin olmak için 15 olarak alınabilir.

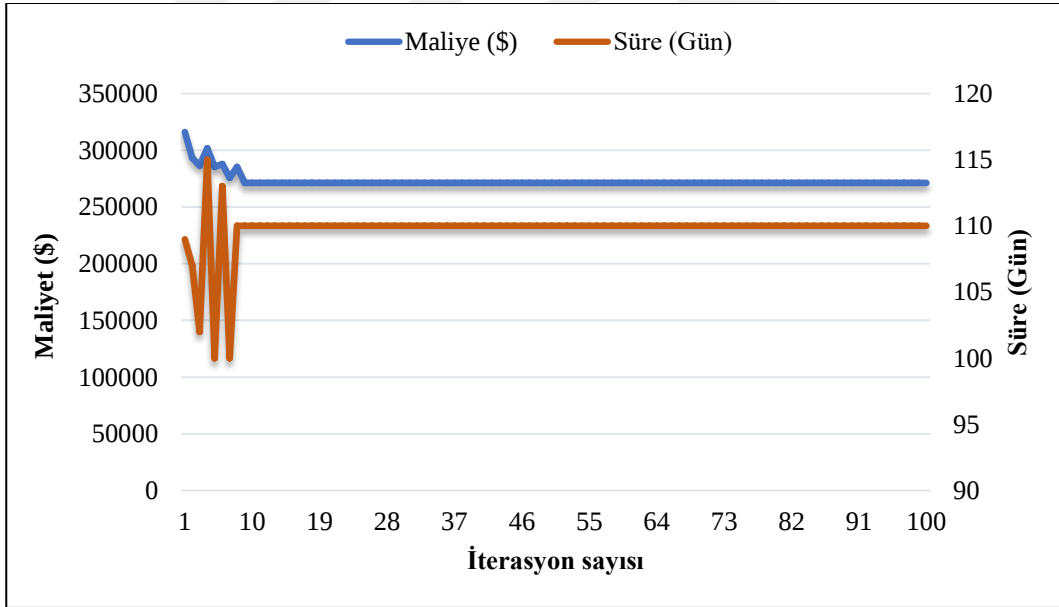


Şekil 3.8. NDSII - Jaya kullanarak 5 modlu 18 faaliyet için maliyet ve iterasyon sayısı yakınsaması





Şekil 3.9. NDSII - Jaya kullanarak 5 modlu 18 faaliyet için süre ve iterasyon sayısı yakınsaması



Şekil 3.10. NDSII - Jaya kullanarak 7 faaliyet için süre, maliyet ve iterasyon sayısı yakınsaması

3.2.3. Üç Modlu 18 Faaliyetli Proje Örneği

Feng vd. (1997) başlangıçta bu örnek problemi sunmuştur. Tablo 3.6, model projesinin faaliyet ilişkileri, her faaliyet için inşaat modları ve bunlarla ilişkili zaman ve



maliyet gibi detaylarını göstermektedir. Ayrıca, dolaylı maliyet için maliyet oranı 1000 \$/gün'dür.

Tablo 3.6. Üç modlu 18 faaliyetli projeler için seçenekler

Faaliyet Numarası	Öncül Faaliyet	Mode 1		Mode 2		Mode 3	
		Süre(gün)	Doğrudan Maliyet (\$)	Süre(gün)	Doğrudan Maliyet (\$)	Süre(gün)	Doğrudan Maliyet (\$)
1	-	14	2400	24	1200	21	1500
2	-	15	3000	25	1000	23	1500
3	-	15	4500	33	3200	33	3200
4	-	12	45000	20	30000	20	30000
5	1	22	20000	30	10000	30	10000
6	1	14	40000	24	18000	24	18000
7	5	9	30000	18	22000	18	22000
8	6	14	220	24	120	21	208
9	6	15	300	25	100	23	150
10	2,6	15	450	33	320	33	320
11	7,8	12	450	20	300	20	300
12	5,9,10	22	2000	30	1000	30	1000
13	3	14	4000	24	1800	24	1800
14	4,10	9	3000	18	2200	18	2200
15	12	12	4500	16	3500	16	3500
16	13,14	20	3000	30	1000	28	1500
17	11,14,15	14	4000	24	1800	24	1800
18	16,17	9	2400	18	1200	18	2200

Tablo 3.7, birkaç matematiksel ve evrimsel temelli yöntemin meta-sezgisel ile karşılaştırılmasına yönelik sonuçları göstermektedir. Önerilen baskın olmayan sıralama II-Jaya algoritması, optimizasyon uygulamaları kullanan diğer GA tabanlı RKV-TCO ve Kısıt Programlama (CP) kadar iyi ve özdeş optimal çözümü doğrular.

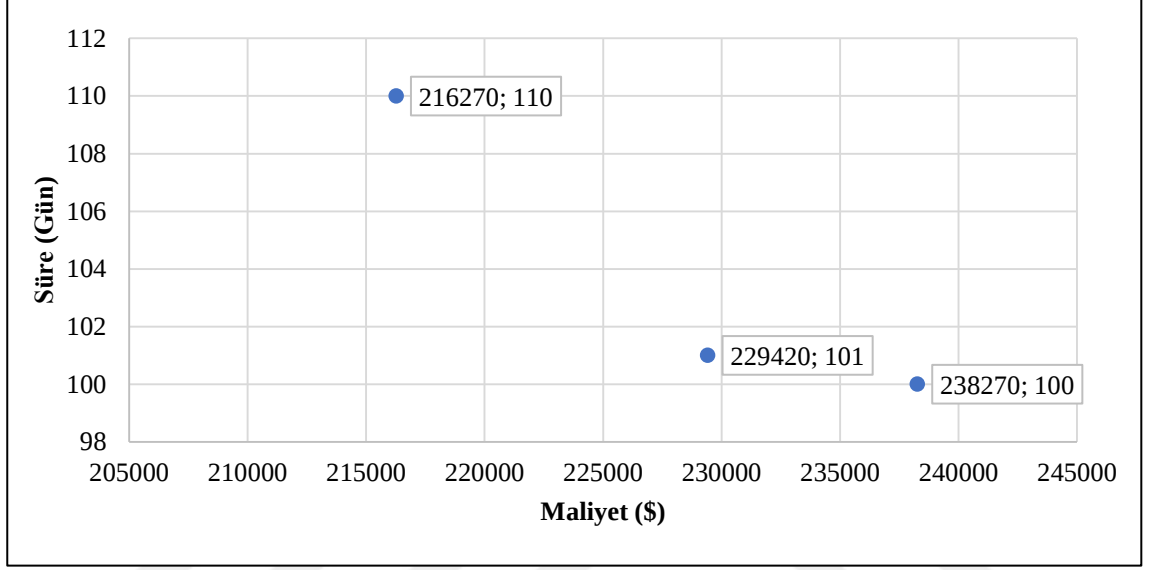


Tablo 3.7. Üç modlu 18 faaliyetli projenin farklı algoritmaları arasında karşılaştırma

YAKLAŞIMLAR	Yazar	Kriterler		Hesaplama zamanı
		Zaman(gün)	Maliyet (\$)	
Excel Çözücü	(Golzarpoor, 2012)	110	254620	2 dakika
Risk Çözücü Platform Standart SLGRG Doğrusal Olmayan		110	216270	1.5 dakika
Risk Çözücü Platform Standart Büyük Ölçekli GRG Çözücü		110	216270	1.5 dakika
TCT Optimizasyonu Evolver (bir evrimsel motor içerir)		110	238070	30 dakika
Risk Çözücü Platform Standart Evrimsel Çözücü		110	275320	18 dakika
Optimizasyon Sonuçları Kullanmakta CPLEX CP Optimizer		110	216270	9 dakika
IBM ILOG Optimizasyon Stüdyosu		110	216270	9 dakika
Rastgele Anahtar Varyantı Zaman-Maliyet Optimizasyonu için (RKVTCO)	(Magalhaes-Mendes, 2015)	110	216270	50 nesil için 5 saniye
Feng vd. (1997) modeli	(Feng vd.,1997)	110	216270	-
		Populasyon hacmı: 400 İterasyon sayısı: 50 NFE: 20000		
MAWA-TLBO	(Eirgash,2018)	110	216270	50 nesil için 1 dakika
		Populasyon hacmı: 40 İterasyon sayısı: 70 NFE: 5640		
NDS-II-Jaya	Bu araştırma	110	216270	60 nesil için 1 dakika
		101	229420	
		100	238270	
		Populasyon hacmı: 50 İterasyon sayısı: 100 NFE: 5050		



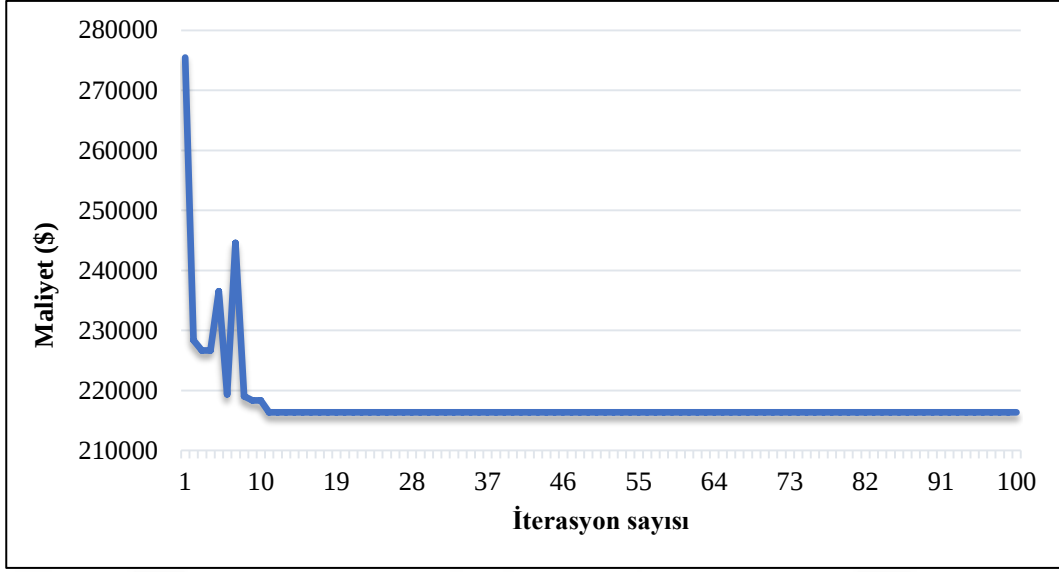
Baskın olmayan sıralama II-Jaya algoritması optimum çözüme hızlı bir şekilde ulaşır. Kullanılan bu algoritma, toplam arama alanının sadece küçük bir kısmını arayarak verimliliğini ve doğruluğunu ifade eder. Baskın olmayan sıralama II-Jaya, çözüm alanının sadece küçük bir kısmının popülasyon sayısı 50 ve yineleme sayısı 100 olan 5050 ($NFE = 50 \times 100 + 50$) olası farklı çizelgeyi araştırdı.



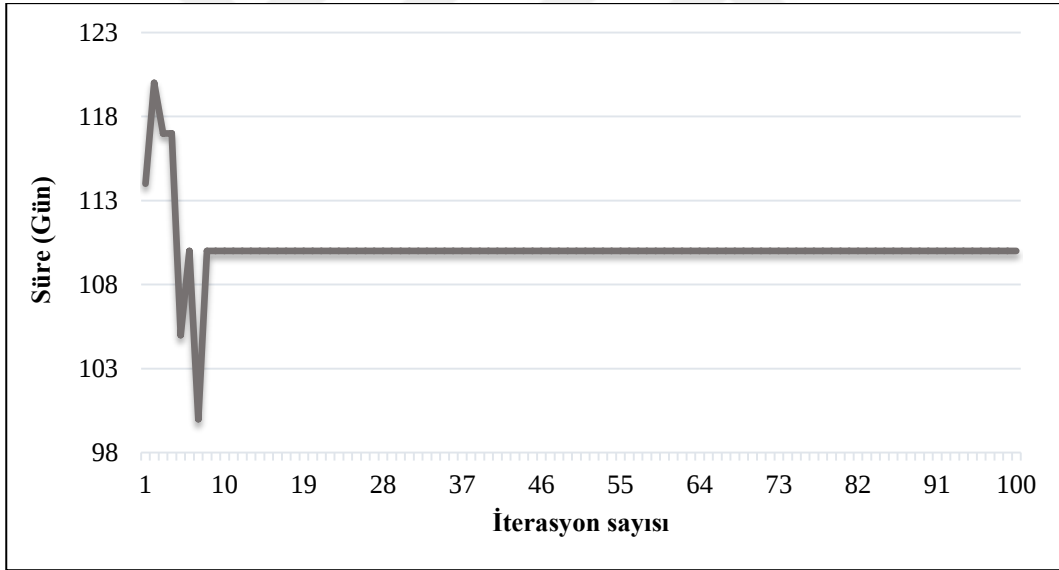
Şekil 3.11. NDS-II-Jaya tarafından 18 faaliyetli problem için ulaşılan Pareto optimal çözümü

Aşağıdaki şekiller zaman, maliyet ve iterasyon sayısına arasındaki yakınsamayı temsil etmektedir. Baskın olmayan sıralama II- Jaya algoritmasını kullanılan iterasyon sayısı biraz daha büyük olduğu ve 8’inci iterasyon sonra gereksiz yere çalıştığı anlamına gelir. Bu nedenle, kullanıldığı zaman-maliyet ödünleşim problemi için iterasyon sayısı en uygun çözümünden emin olmak için 15 olarak alınabilir.



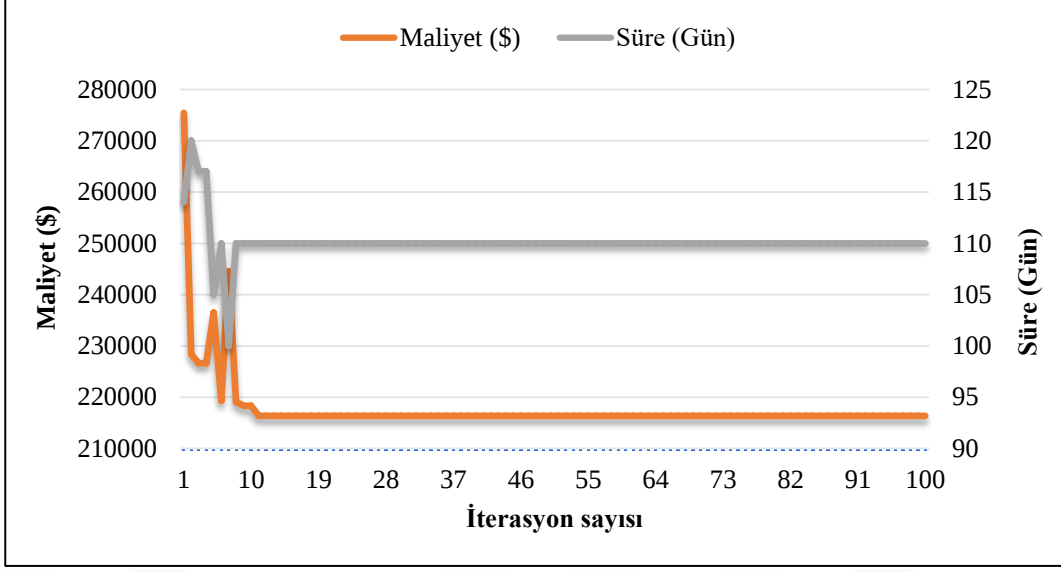


Şekil 3.12. NDSII - Jaya kullanarak 5 modlu 18 faaliyet için maliyet ve iterasyon sayısı yakınsaması



Şekil 3.13. NDSII - Jaya kullanarak 5 modlu 18 faaliyet için süre ve iterasyon sayısı yakınsaması.



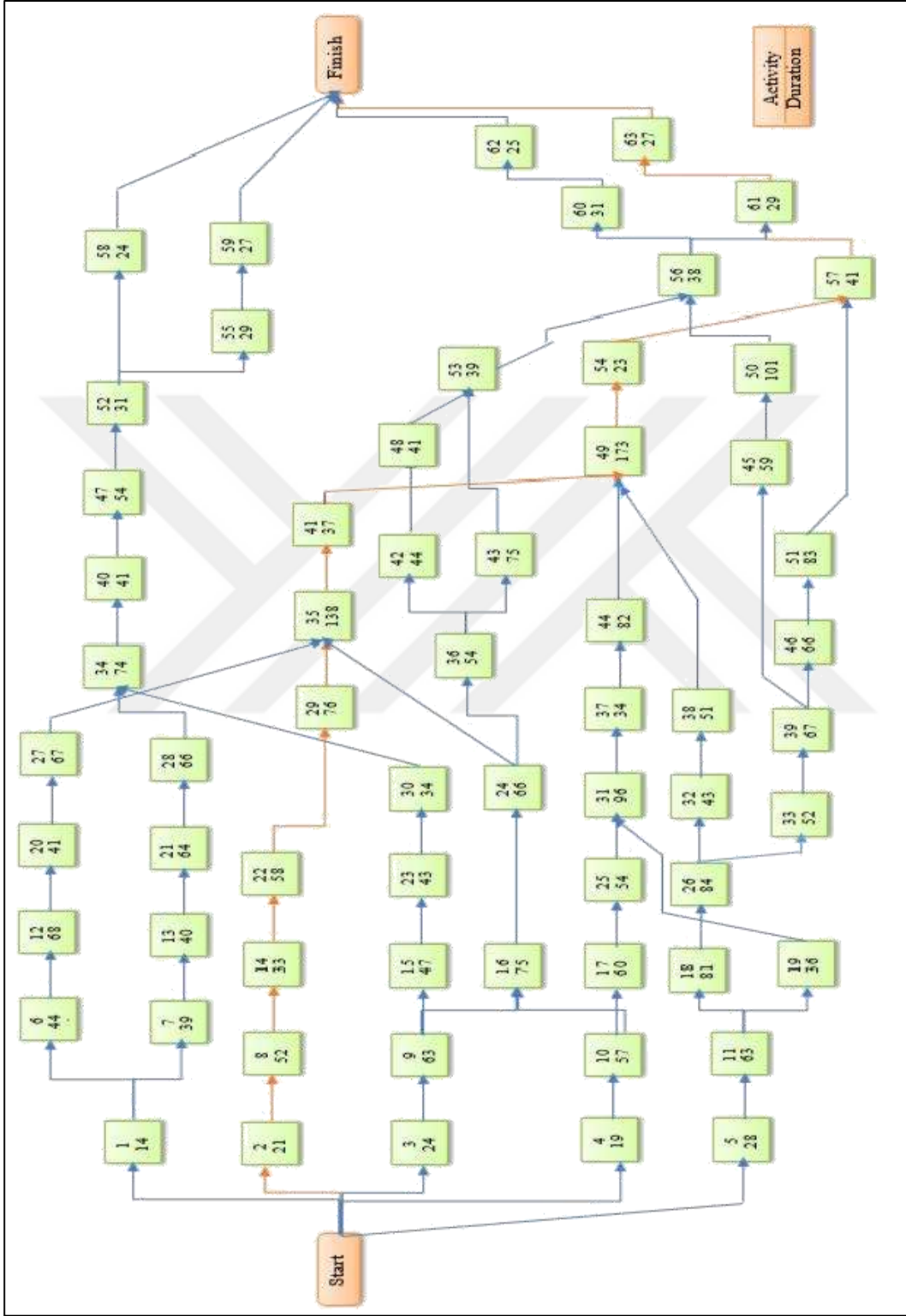


Şekil 3.14. NDSII - Jaya kullanarak 5 modlu 18 faaliyet için süre, maliyet ve iterasyon sayısı yakınsaması

3.2.4. 63 Faaliyetli Proje Örneği.

Bu çalışmada baskın olmayan sıralama II-Jaya yaklaşımı, NDS'ya dayalı birkaç yaklaşımla karşılaştırılacaktır. Literatürde daha fazla etkinlik içeren NDSII-Jaya ile incelenen herhangi bir projeye rastlanmamıştır. Baskın olmayan sıralama II-Jaya'nın 18'den fazla faaliyet içeren bir inşaat projesine entegre performansını sergilemek için bu çalışmada, Bettemir'den (2009) alınan 63 faaliyetli bir proje Şekil 3.15.'de gösterilen ağ NDSII-Jaya tarafından incelenmiştir. Tablo 3.8, model projesinin faaliyet ilişkileri, her faaliyet için inşaat modları ve bunlarla ilişkili zaman ve maliyet gibi detaylarını göstermektedir.





Şekil 3.15. 63 faaliyetli ağın ağ temsili



Tablo 3.8. 63 faaliyetli zaman maliyet ödünleşim problemi için veriler

Faaliyet Sayı	Öncül Faaliyet	Mod 1		Mod 2		Mod 3		Mod 4		Mod 5	
		Süre (gün)	Maliyet (\$)	Süre (gün)	Maliyet (\$)	Süre (gün)	Maliyet (\$)	Süre (gün)	Maliyet (\$)	Süre (gün)	Maliyet (\$)
1	-	14	3700	12	4250	10	5400	9	6250		
2	-	21	11250	18	14800	17	16200	15	19650		
3	-	24	22450	22	24900	19	27950	17	31650		
4	-	19	17800	17	19400	15	21600				
5	-	28	31180	26	34200	23	38250	21	41400		
6	1	44	54260	42	58450	38	63225	35	68150		
7	1	39	47600	36	50750	33	54800	30	59750		
8	2	52	62140	47	69700	44	72600	39	81750		
9	3	63	72750	59	79450	55	86250	51	91500	49	99500
10	4	57	66500	53	70250	50	75800	46	80750	41	86450
11	5	63	83100	59	89450	55	97800	50	104250	45	112400
12	6	68	75500	62	82000	58	87500	53	91800	49	96550
13	7	40	34250	37	38500	33	43950	31	48750		
14	8	33	52750	30	58450	27	63400	25	66250		
15	9	47	38140	40	41500	35	47650	32	54100		
16	9,10	75	94600	70	101250	66	112750	61	124500	57	132850
17	10	60	78450	55	84500	49	91250	47	94640		
18	10,11	81	127150	73	143250	66	154600	47	161900		
19	11	36	82500	34	94800	30	101700				
20	12	41	48350	37	53250	34	59450	32	66800		
21	13	64	85250	60	92600	57	99800	53	107500	49	113750
22	14	58	74250	53	79100	50	86700	47	91500	42	97400
23	15	43	66450	41	69800	37	75800	33	81400	30	88450
24	16	66	72500	62	78500	58	83700	53	89350	49	96400
25	17	54	66650	50	70100	47	74800	43	79500	40	86800
26	18	84	93500	79	102500	73	111250	68	119750	62	128500
27	20	67	78500	60	86450	57	89100	56	91500	53	94750
28	21	66	85000	63	89750	60	92500	58	96800	54	100500
29	22	76	92700	71	98500	67	104600	64	109900	60	115600
30	23	34	27500	32	29800	29	31750	27	33800	26	36200
31	19,25	96	145000	89	154800	83	168650	77	179500	72	189100
32	26	43	43150	40	48300	37	51450	35	54600	33	61450
33	26	52	61250	49	64350	44	68750	41	74500	38	79500
34	28,30	74	89250	71	93800	66	99750	62	105100	57	114250
35	24,27,29	138	183000	126	201500	115	238000	103	283750	98	297500
36	24	54	47500	49	50750	42	56800	38	62750	33	68250
37	31	34	22500	32	24100	29	26750	27	29800	24	31600



Tablo 3.8. Tablo devamı

38	32	51	61250	47	65800	44	71250	41	76500	38	80400
39	33	67	81150	61	87600	57	92100	52	97450	49	102800
40	34	41	45250	39	48400	36	51200	33	54700	31	58200
41	35	37	17500	31	21200	27	26850	23	32300		
42	36	44	36400	41	39750	38	42800	32	48300	30	50250
43	36	75	66800	69	71200	63	76400	59	81300	54	86200
44	37	82	102750	76	109500	70	127000	66	136800	63	146000
45	39	59	847500	55	91400	51	101300	47	126500	43	142750
46	39	66	94250	63	99500	59	108250	55	118500	50	136000
47	40	54	73500	51	78500	47	83600	44	88700	41	93400
48	42	41	36750	39	39800	37	43800	34	48500	31	53950
49	38,41,44	173	267500	159	289700	147	312000	138	352500	121	397750
50	45	101	47800	74	61300	63	76800	49	91500		
51	46	83	84600	77	93650	72	98500	65	104600	61	113200
52	47	31	23150	28	27600	26	29800	24	32750	21	35200
53	43,48	39	31500	36	34250	33	37800	29	41250	26	44600
54	49	23	16500	22	17800	21	19750	20	21200	18	24300
55	52,53	29	23400	27	25250	26	26900	24	29400	22	32500
56	50,53	38	41250	35	44650	33	47800	31	51400	29	55450
57	51,54	41	37800	38	41250	35	45600	32	49750	30	53400
58	52	24	12500	22	13600	20	15250	18	16800	16	19450
59	55	27	34600	24	37500	22	41250	19	46750	17	50750
60	56	31	28500	29	30500	27	33250	25	38000	21	43800
61	56,57	29	22500	27	24750	25	27250	22	29800	20	33500
62	60	25	38750	23	41200	21	44750	19	49800	17	51100
63	61	27	9500	26	9700	25	10100	24	10800	22	12700

Proje üç modlu 2 faaliyet, dört modlu 15 faaliyet ve beş modlu 46 faaliyet içerir. Proje iki durumda incelenmiştir: Dolaylı maliyet birinci durumda (63a) günde 2300\$ alınır, ikinci durumda (63b) ise günde 3500\$ olarak uygulanmaktadır. 63a için 630 gün ve 5.421.120\$; 63b için 621 gün ve 6.176.170\$'lık optimal çözümler Bettemir (2009) tarafından tamsayı programlama kullanılarak sağlanmıştır. Bettemir (2009), bahsedilen zaman maliyet ödünleşim problemini çözmek için üç temel algoritma kullanmıştır.

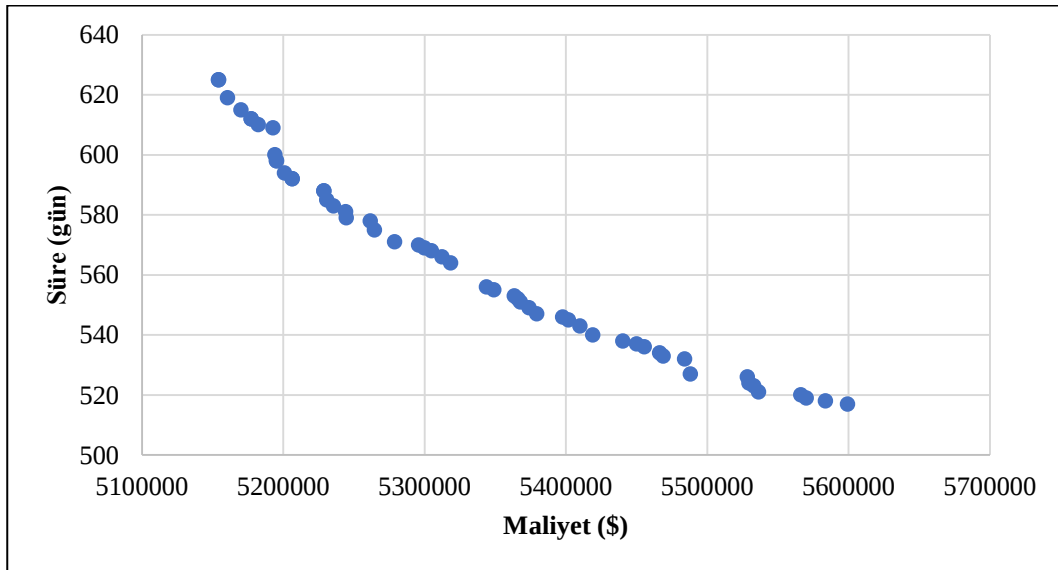
Bu çalışmada baskın olmayan sıralama II-Jaya kullanılarak elde edilen sonuçlar, Bettemir'in (2009) NDS-GA, NDS-ACO, NDS-PSO ve Eirgash'in (2018) NDS-TLBO modellerinden elde edilen çözümlerle karşılaştırılmıştır. 63a ve 63b faaliyet problemlerinin karşılaştırılan sonuçları sırasıyla Tablo 3.9 ve 3.10'da sunulmuştur. Mevcut çözülmüş



problemlerin Pareto ön çözümlerinin grafiksel gösterimleri Şekil 3.16 ve 3.17’de verilmiştir.

Tablo 3.9. NDS-II-Jaya kullanılan Durum 1 için 63 faaliyetli projenin analiz sonuçları (IC= \$2300)

No	Bettemir (2009)						Eirgash (2018)		Bu araştırma	
	NDS-GA		NDS-ACO		NDS-PSO		NDS-TLBO		NDS-II-Jaya	
	Süre	Maliyet	Süre	Maliyet	Süre	Maliyet	Süre	Maliyet	Süre	Maliyet
1	641	5704200	635	5490120	637	5421620	630	5428870	540	5419090
2	661	5712485	653	5494410	644	5428920	630	5428120	540	5407540
3	650	5722260	638	5491180	651	5439620	630	5427770	543	5384130
4	653	5713450	657	5491620	634	5422920	630	5428120	538	5419865
5	645	5699650	644	5494920	651	5440570	630	5428920	538	5420940
6	639	5684295	626	5486630	633	5421320	637	5428220	539	5416900
7	640	5695655	664	5495080	633	5421320	633	5428870	542	5420620
8	621	5707600	661	5490350	633	5421620	628	5428170	545	5414790
9	641	5693015	643	5490680	633	5421320	633	5428470	543	5409330
10	623	5690790	635	5492210	633	5421320	633	5428720	545	5424865
Popülasyon sayısı			-					180	180	
İterasyon sayısı			-					450	400	

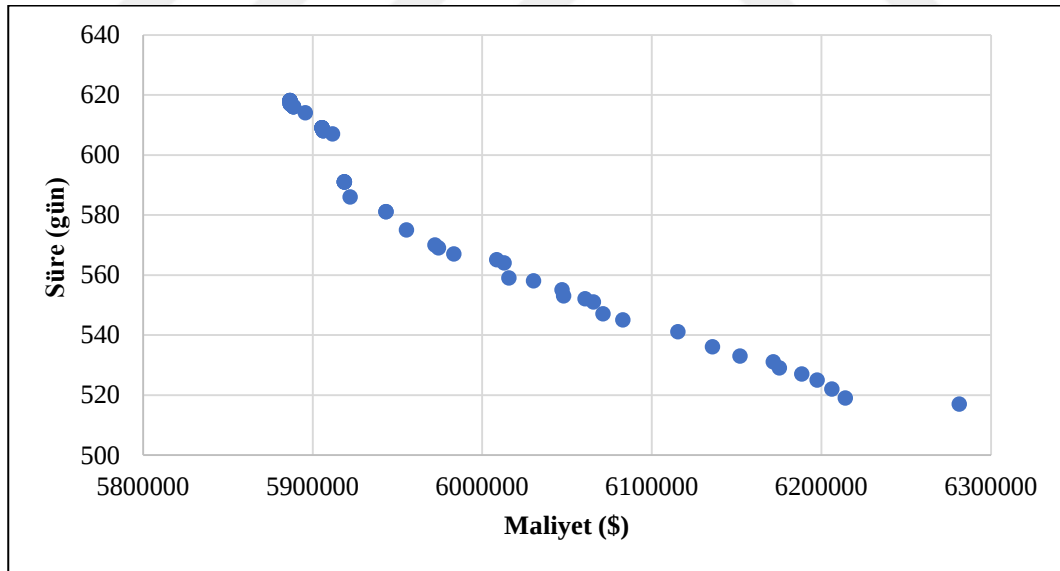


Şekil 3.16. Durum 1 için NDS-II-Jaya algoritması ile elde edilen 63 faaliyetli problemin Pareto ön çözümleri



Tablo 3.10. NDS-II-Jaya kullanılan Durum 2 için 63 faaliyetli projenin analiz sonuçları (IC= \$3500)

No	Bettemir (2009)						Eirgash (2018)		Bu araştırma	
	NDS-GA		NDS-ACO		NDS-PSO		NDS-TLBO		NDS-II-Jaya	
	Süre	Maliyet	Süre	Maliyet	Süre	Maliyet	Süre	Maliyet	Süre	Maliyet
1	617	6462580	631	6219220	644	6201720	612	6192140	533	6152200
2	651	6411540	632	6205850	629	6217470	617	6184820	536	6147070
3	647	6442440	626	6234520	644	6210170	590	6188690	522	6160395
4	639	6420500	640	6223830	648	6218170	588	6195910	521	6191850
5	648	6447900	617	6231440	649	6216020	591	6191490	519	6178910
6	627	6433810	627	6197070	647	6207870	586	6196840	517	6191520
7	618	6439240	604	6247850	651	6216220	592	6189140	523	6176760
8	623	6449790	635	6231860	649	6215420	589	6199870	520	6185320
9	630	6443805	623	6198650	645	6208920	617	6187390	520	6166960
10	629	6450065	651	6262830	642	6198520	616	6190570	523	6137790
Popülasyon boyutu			-				180		180	
İterasyon sayısı			-				450		400	



Şekil 3.17. Durum 2 için NDS-II-Jaya algoritması ile elde edilen 63 faaliyetli problemin Pareto ön çözümleri

Baskın olmayan sıralama II-Jaya, $NFE = 72180 (= 180 \times 400 + 180)$ olası farklı çizelgeleri araştırdı; çözüm alanının yalnızca ihmal edilebilir bir bölümü Pareto ön çözümler üretebildi. Popülasyon ve yineleme sayısı sırasıyla 180 ve 400'dür.

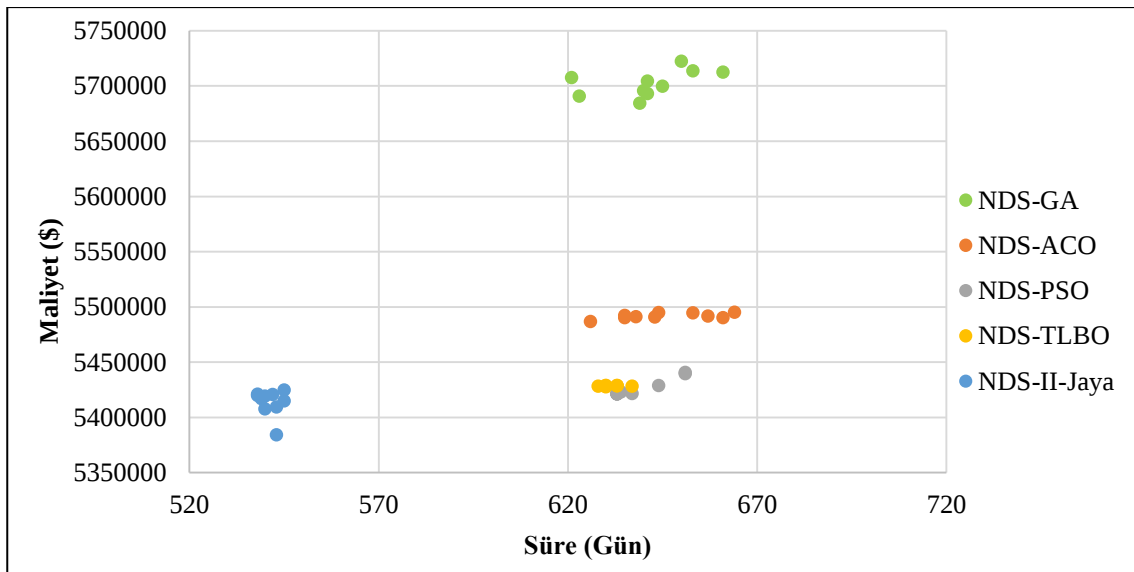


Tablo 3.11. NDS-II-Jaya kullanarak 63a ve 63b problemleri için yüzde sapma ortalamasını karşılaştırılması

Algoritmalar	Durum 1		Durum 2	
	Koşu	YSO (%)	Koşu	YSO (%)
GA, Bettemir (2009)	10	5,86	10	5,16
ACO, Bettemir (2009)	10	1,2	10	0,7
PSO, Bettemir (2009)	10	0,152	10	0,2
NDS-TLBO, (Eirgash, 2018)	10	0,128	10	0,14
NDS-II-Jaya (Bu araştırma)	10	0,135	10	0,12

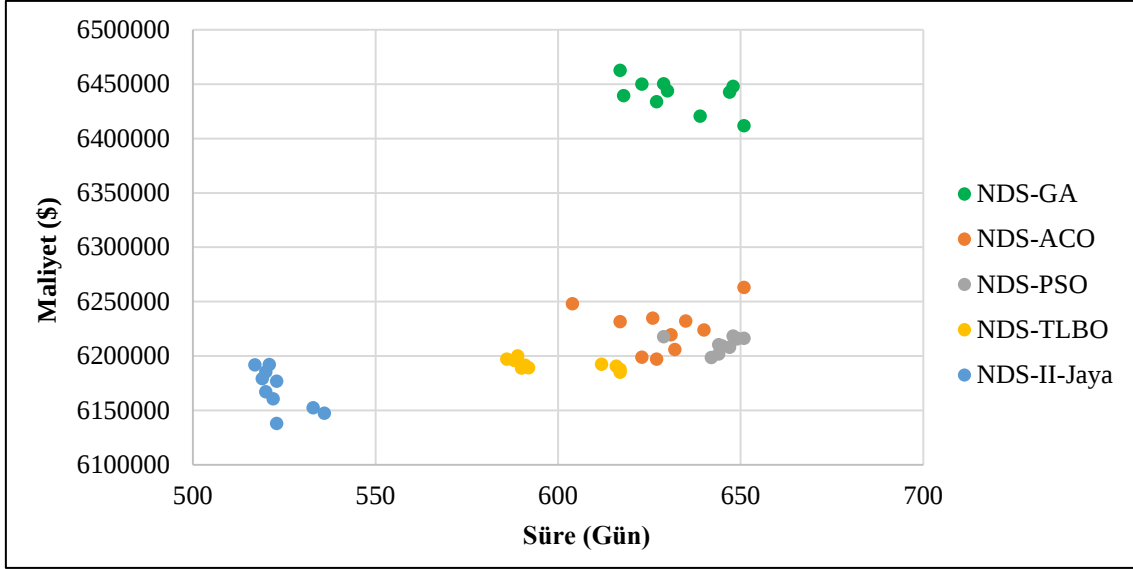
Tablo 3.11 Yüzde sapma ortalaması değerleri sırasıyla %0,135 ve %0,12'dir. Baskın olmayan sıralama II-Jaya tabanlı modelin hem fonksiyon değerlendirme sayısının hem de yüzde sapma ortalamasını Bettemir (2009) ve Eirgash'ın (2018) modellerinden çok düşük olduğu anlamına gelir. Sonuç olarak, önerilen algoritma, önceki modellere nazaran orta ölçekli zaman maliyet ödünleşim problemlerini çözmede daha fazla keşif kabiliyetine sahip ve daha pratiktir.

Bu çalışmada baskın olmayan sıralama II-Jaya kullanılarak elde edilen sonuçlar NDS-TLBO, NDS-PSO, NDS-ACO ve NDS-GA modelleri ile elde edilen çözümlerle karşılaştırılmıştır. 63 faaliyet (durum1 ve durum2) zaman maliyet ödünleşim probleminin karşılaştırılan sonuçları Şekil 3.18 ve Şekil 3.19'de sunulmuştur.



Şekil 3.18 NDS-II-Jaya kullanılan Durum 1 için 63 faaliyetli problemi analiz karşılaştırması (IC= \$2300)





Şekil 3.19. NDS-II-Jaya kullanılan Durum2 için 63 faaliyetli problemi analiz karşılaştırması (IC= \$3500)

3.2.5. 630 Faaliyetli Proje Örneği

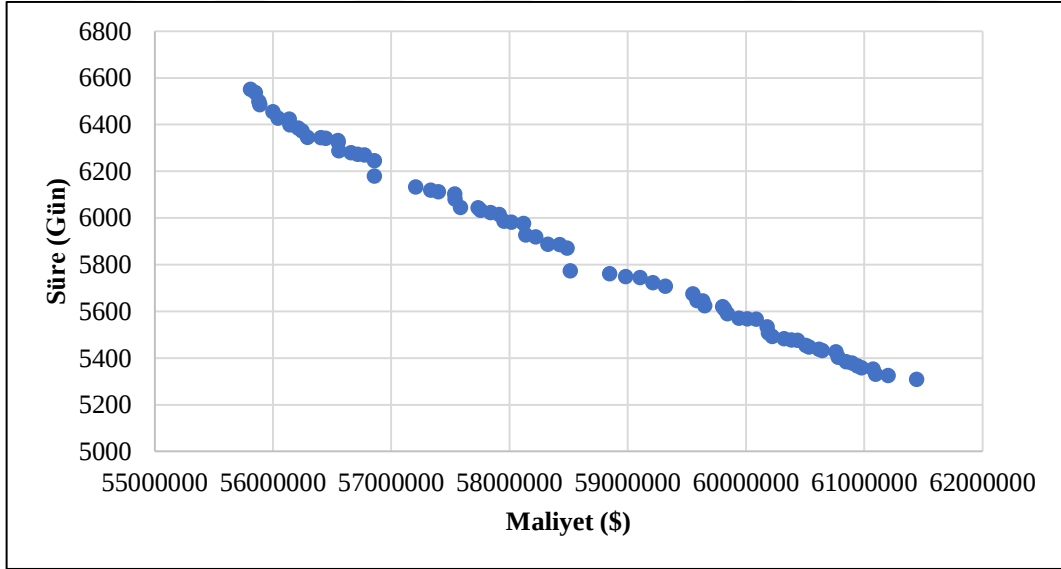
Baskın olmayan sıralama II-Jaya'nın 630 faaliyetli büyük ölçekli bir projedeki verimliliğini araştırmak da önerilen algoritma ile çözülmüştür. Model proje 63 faaliyetli projenin 9 kez çoğaltılmasıyla oluşturulmuştur. (Bettemir, 2009). Proje modeli iki örnek durum içermektedir: Günlük dolaylı maliyetleri 2300\$'lık Durum 1 ve 3500\$'lık Durum 2 çözülmüştür. Durum 1 ve Durum 2 için karışık tamsayı programlama kullanılarak Bettemir (2009) tarafından elde edilen global optimumlar sırasıyla 54.211.200 \$ ve 61.761.700 \$'dır.

Bu proje için de ardışık on deneysel çalışma yürütülmüştür. Durum 1 ve Durum 2 için on çalışmanın en iyi sonuçları Tablo 3.12 ve Tablo 3.13'te optimizasyondan karşılık gelen yüzde sapma ortalamasını (% YSO) ile sunulmuş ve Eirgash (2018) ile karşılaştırılmıştır. Önerilen baskın olmayan sıralama II-Jaya ile elde edilen Pareto ön çözümünün grafik gösterimleri Şekil 3.20 ve 3.21'te sunulmuştur.



Tablo 3.12. 630 faaliyetli proje için en iyi sonuçlar (Durum1: Günlük dolaylı maliyet 2300\$)

NDS-TLBO (Eirgash,2018)			NDS-II-Jaya		
Süre (gün)	Maliyet (\$)	Yüzdelik Sapma	Süre (gün)	Maliyet (\$)	Yüzdelik Sapma
6373	54611340	0,74	6551	55811410	2,95
6387	54775880	1,04	6655	56082145	3,45
6383	54805960	1,09	6591	55969370	3,24
6364	54829460	1,14	6619	56230260	3,72
6360	54856620	1,19	6654	56466440	4,16
6302	54943070	1,35	6570	56130710	3,54
6377	54692200	0,88	6664	56083385	3,45
6388	54705310	0,91	6599	55969170	3,24
6346	54849940	1,17	6596	55977065	3,26
6300	54992260	1,44	6602	56219930	3,71
Popülasyon hacmi: 250		%YSO 1,10	Popülasyon hacmi: 250		%YSO 3,47
İterasyon sayısı: 450			İterasyon sayısı: 450		
NFE : 225250			NFE : 112750		

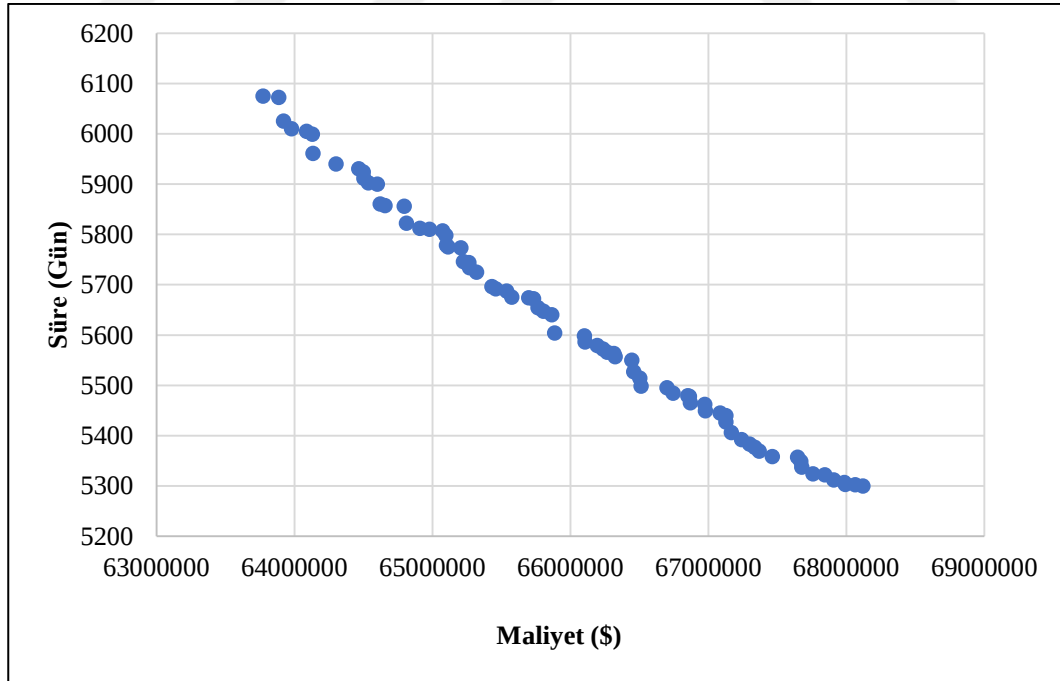


Şekil 3.20. Durum 1 için NDS-II-Jaya algoritması ile elde edilen 630 faaliyetli problemin Pareto ön çözümleri



Tablo 3.13. 630 faaliyetli proje için en iyi sonuçlar (Durum 2: Günlük dolaylı maliyet \$3500)

NDS-TLBO (Eirgash, 2018)			NDS-II-Jaya		
Süre (gün)	Maliyet (\$)	Yüzdelik Sapma	Süre (gün)	Maliyet (\$)	Yüzdelik Sapma
6212	62793865	1.67	6150	63959520	3,56
6220	62750580	1.60	6173	63952720	3,55
6204	62591490	1.34	6123	64038260	3,69
6232	62692340	1.50	6149	63778570	3,27
6236	62741130	1.58	6077	63956125	3,55
6225	62586260	1.33	6080	64073810	3,74
6201	62744310	1.59	5996	63852690	3,39
6127	62650570	1.43	6075	63774240	3,26
6190	62699400	1.51	6152	63946930	3,54
6279	62734550	1.57	6149	64108140	3,80
Popülasyon Boyutu: 250		%YSO 1,51	Popülasyon Boyutu: 250		%YSO 3,53
İterasyon sayısı: 450			İterasyon sayısı: 450		
NFE : 225250			NFE : 112750		



Şekil 3.21. Durum 2 için NDS-II-Jaya algoritması ile elde edilen 630 faaliyetli problemin Pareto ön çözümleri



Durum 1 ve Durum 2 için on koşu ortalama değerleri sırasıyla Tablo 3.14 ve 3.15'te sunulmuştur ve diğer araştırmalarla karşılaştırılmıştır. Ek olarak, Tablo 3.15 Durum 1 ve Durum 2'nin önceki modellerle karşılaştırılan yüzde sapma ortalamasını (%YSO) değerlerini göstermektedir.

Tablo 3.14. Durum 1 için 10 koşunun ortalama değerlerinin karşılaştırması (IC = 2300 \$)

Açıklama	Eirgash (2018)	Bettemir (2009)			Bu araştırma
	NDS-TLBO	NDS-ACO	NDS-PSO	NDS-GA	NDS-II-Jaya
Ortalama değeri	54806204	56703583	54815790	58983147	56652057
Populasyon sayısı	250	-	-	-	250
İterasyon sayısı	450	-	-	-	450
NFE	225250	250000	250000	250000	112750

Tablo 3.15. Durum 2 için 10 koşunun ortalama değerlerinin karşılaştırması (IC = 3500 \$)

Açıklama	Eirgash (2018)	Bettemir (2009)			Bu araştırma
	NDS-TLBO	NDS-ACO	NDS-PSO	NDS-GA	NDS-II-Jaya
Ortalama değer	62698449	64574989	63121500	66395840	64456586
Populasyon sayısı	250	-	-	-	250
İterasyon sayısı	450	-	-	-	450
NFE	225250	250000	250000	250000	112750

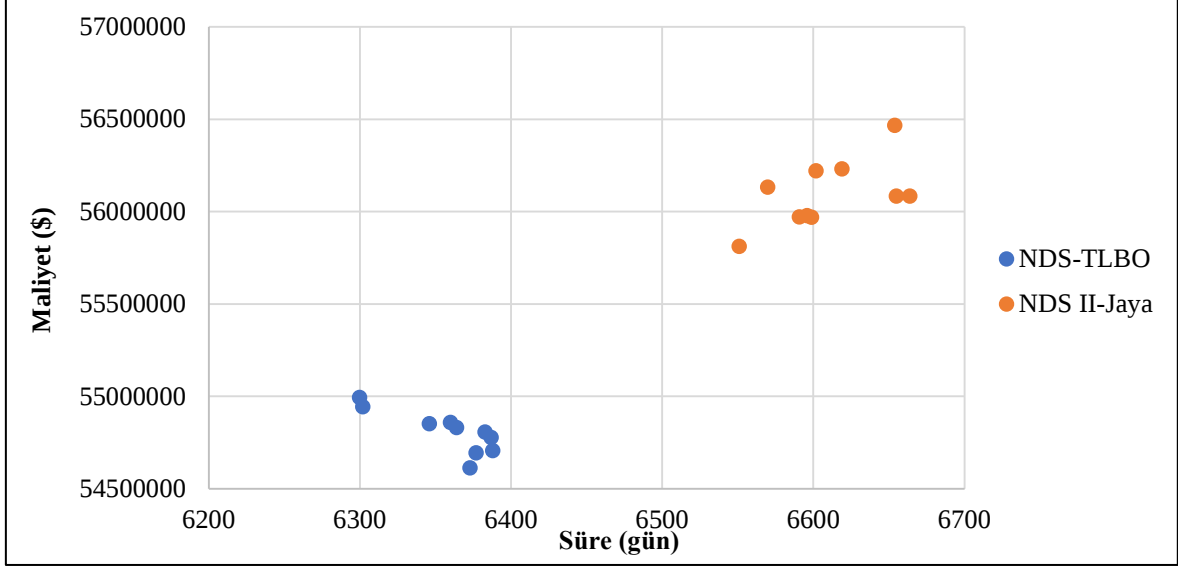
Tablo 3.16. NDS-II-Jaya ile elde edilen modeller için optimal çözümlerden 630 faaliyetli problemin ortalama sapmaları

Algoritmalar	Durum 1		Durum 2	
	Koşu	% YSO	Koşu	% YSO
GA, Sönmez ve Bettemir (2012)	10	8.83	10	7.5
ACO, Sönmez ve Bettemir (2012)	10	4.59	10	4.55
PSO, Sönmez ve Bettemir (2012)	10	1.11	10	2.21
NDS-TLBO, (Eirgash,2018)	10	1.10	10	1.51
NDS-II-Jaya (Bu araştırma)	10	3.53	10	3.47

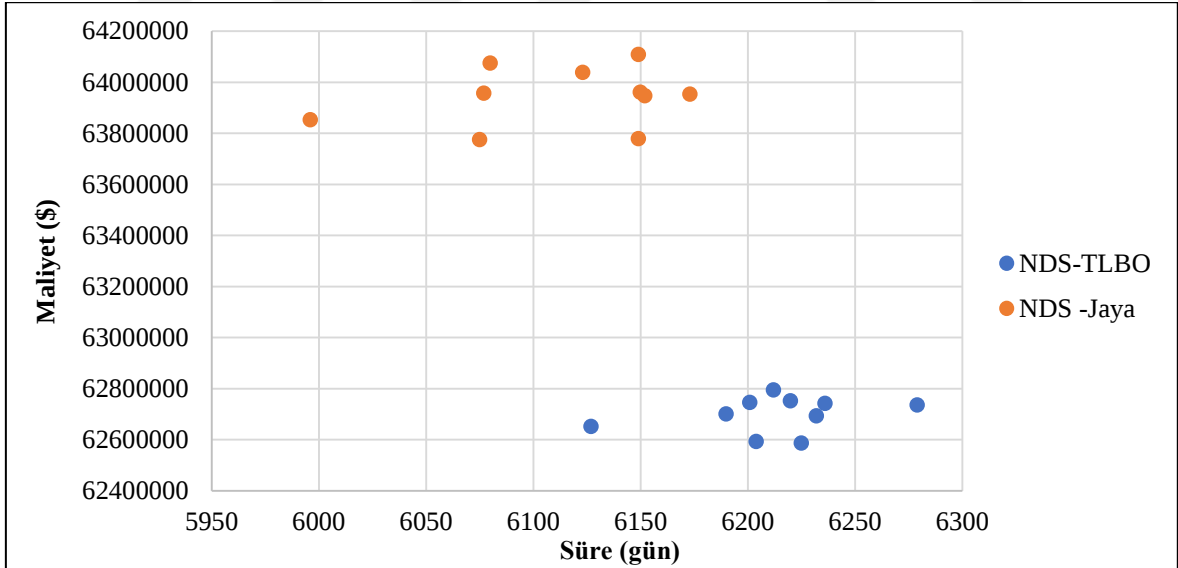
Bu çalışmada baskın olmayan sıralama II-Jaya kullanılarak elde edilen sonuçlar NDS-TLBO modeli ile elde edilen çözümlerle karşılaştırılmıştır. 630 faaliyet (durum1 ve



durum2) zaman maliyet ödünleşim probleminin karşılaştırılan sonuçları Şekil 3.22 ve Şekil 3.23'te sunulmuştur.



Şekil 3.22. NDS-II-Jaya kullanılan Durum 1 için 630 faaliyetli problemi analiz karşılaştırması (IC= \$2300)



Şekil 3.23. NDS-II-Jaya kullanılan Durum 2 için 630 faaliyetli problemi analiz karşılaştırması (IC= \$3500)

İki durum için baskın olmayan sıralama II -Jaya'nın YSO değerleri 3.53 ve 3.47'dır. Bu değerler NDS-GA'nın YSO değerlerinden küçüktür, ancak Sönmez ve Bettemir (2012)



tarafından önerilen NDS-ACO, NDS-PSO modellerinin değerlerinden büyüktür. Ayrıca, Tablo 3.16 dikkate alındığında (Eirgash, 2018) tarafından önerilen NDS-TLBO'dan daha büyüktür. Baskın olmayan sıralama II-Jaya'nın büyük ağlar için sonuçları, önceki araştırmada kullanılan diğer algoritmadan farklı olduğunu göstermektedir; proje maliyeti yüksekti ancak proje süresi sonuçları önceki araştırmadan çok farklıydı, en iyi çözüm hangisinin projesi için uygun olduğuna karar verebilecek proje yöneticisi ile ilgilidir.

Bu nedenle, baskın olmayan sıralama II-Jaya modeli, GA ile diğer modellerin performansı arasındadır. Ayrıca zaman maliyet ödünleşim problemleri için normal bir alternatif sağlar. İnşaat yönetimi alanında, zaman maliyet ödünleşimi optimizasyon zorlukları doğrusal programlama veya diğer hesaplama yaklaşımları ile çözülmeyecektir. Bununla birlikte, bu tür sorunları optimize etmek için çeşitli meta-sezgisel optimizasyon algoritmaları uygulanmıştır. Bu analiz, çok amaçlı optimizasyonla sorunları çözmek için önerilen modelin uygunluğunu doğrulamak için yeni geliştirilen bir baskın olmayan sıralama II-Jaya algoritmasını ve kalabalık mesafesi hesaplama mekanizmasını açıklar. Baskın olmayan sıralama II-Jaya algoritmasının etkinliği, 7 faaliyetli ve 18 faaliyetli küçük ölçekli projelerde, 63 faaliyetli orta ölçekli projede ve 630 faaliyetli daha karmaşık ve büyük ölçekli projede test edilmiştir. Sayısal sonuçlara dayanarak, baskın olmayan sıralama II-Jaya tabanlı modelinin, hem toplam işlev değerlendirme sayısından daha az hem de daha önce önerilen modellerden yüksek ortalama yüzde sapma ile alternatif Pareto ön çözüm ürettiği belirtilebilir. Sonuçlar ayrıca, baskın olmayan sıralama II-Jaya'nın küçük ve orta ölçekli zaman maliyet ödünleşim problemlerinin eşzamanlı optimizasyonunu çözmek için, büyük bir potansiyele sahip olduğunu göstermektedir. Dahası, basitlik mevcut yöntemin güçlü yanı olarak dikkate alınabilir.



4. SONUÇLAR

Bu tezde, önerilen algoritmaların doğrulama ve performans değerlendirmelerinin eşiğinde, modellere iyi bilinen bir 7 ve 18 faaliyetli kıyaslama probleminin yanı sıra daha karmaşık bir 63 ve 630 faaliyetli problem uygulanarak deneysel analizler yapılmıştır. Literatürde bu örnekler için verilen çözümlere göre, algoritmaları çalıştırmak için gereken yeterli değer, bir dizi keşif ve hata sonrasında kabul edilmiştir. Bu çalışmalardan elde edilen sonuçlarla, uygulanan algoritmaların etkinliği ve verimliliği araştırılmıştır. Bu tezde tanıtılan zaman maliyet ödünleşim problemlerinde kullanılan baskın olmayan sıralama II-Jaya, bu nedenle birçok araştırmada test edilen diğer algoritmalarla karşılaştırılmıştır.

Bununla birlikte, hesaplama sonucu yaklaşık 63 faaliyete kadar, zaman maliyet ödünleşim problemini çözmek için kullanılan baskın olmayan sıralama II-Jaya algoritmasının tatmin edici davranışını ortaya çıkarmıştır. Faaliyet sayısının ve dolaylı maliyet değerinin sıklığının algoritmanın etkinliğini etkilediği görülmektedir. Bu meta-sezgisel prosedür, optimum çözümlerden ortalama yüzde ondan fazla sapmayan çözümler üretir.

Bu çalışmada, günlük dolaylı maliyetlerin daha düşük olduğu test problemleri için elde edilen çözümlerin kalitesinin bir şekilde bozulduğu görülmüştür. Ayrıca bu modelin küçük ve orta ölçekli problemler için baskın olmayan cepheyi tespit etme yeterliliğine göre sağlamlığı doğrulanmıştır. Bu nedenle, uygulanan algoritmanın literatürde bildirilen önceki çalışmaların sonuçlarını daha iyi çalıştığını kanıtlanmıştır.

Bu araştırma, ayırık zaman maliyet ödünleşim problemlerini geniş ölçekte ele almanın eksikliklerinin güçlendirilmesine büyük ölçüde katkıda bulunmaktadır. Literatürdeki zaman maliyet ödünleşim problemlerini çözme yöntemleri 63 aktiviteyle sınırlı olmasına rağmen, önerilen algoritmayı kullanarak, çözümün verimliliğinden ödün vermeden operasyon sayısı, global optimumdan makul bir aralıkta 630 aktiviteye çıkar. Aşağıdakiler araştırmanın katkılarının bir özetidir:

- İnşaat mühendisliği inşaat yönetimi alanında, zaman maliyet ödünleşim problemlerinin çözümünde, ilk defa çok amaçlı baskın olmayan sıralama II-Jaya uygulanmasında kullanılmak üzere, MATLAB ortamında esnek bir zaman maliyet ödünleşim zaman maliyet ödünleşim modeli geliştirilmiştir.



- Geliştirilen kısmi rastgele ilk çok büyük ölçekli zaman maliyet ödünleşim problemlerini çözme yeteneğini kanıtladı. Çözümler, kabul edilebilir bir işlem süresi ile optimuma yakın tatmin edicidir.
- Uygulanan çok amaçlı baskın olmayan sıralama II-Jaya yaklaşımlarının önceki optimizasyon yöntemlerine göre üstünlüğü konusunda sonuca varılmıştır.



5. ÖNERİLER

Bu araştırmada sunulan inşaat projelerinin zaman ve maliyet optimizasyonundaki önemli iyileştirmelere rağmen, mevcut araştırmanın gelecekteki uzantıları için aşağıdakiler de dahil olmak üzere çeşitli başka geliştirmeler sunulmaktadır:

- Kalite, kaynak ve güvenlik gibi problemler ekleyerek amaç fonsiyonu ikiden fazla genişletmek daha fazla bağıllığı hak eden bir araştırma alanı olacaktır. Sonuç olarak, model daha karmaşık bir birleşimsel optimizasyon sorununa dönüşecektir ve bu sorunun çözülmesi daha zor ve daha fazla zaman alır.
- Zaman ve maliyeti optimize ederken kaynak kullanımını gerçekleştirmek için optimizasyon modelini kaynak tahsisi ve kaynak seviyelendirme sınırlandırmalarını içerecek şekilde genişletmek.



6. KAYNAKLAR

- Abdel-Raheem, M. ve Khalafallah, A., 2011. Using eclecticize to solve the time cost tradeoff problem in construction engineering., PhD thesis, Department of Civil, Environmental and Construction Engineering, University of Central Florida Orlando, Florida.
- Afruzi, E. N., Najafi, A. A., Roghanian, E. ve Mazinani, M., 2014. A multi-objective imperialist competitive algorithm for solving discrete time, cost and quality trade-off problems with mode-identity and resource-constrained situations. Computers and Operations Research, 50,80-96.
- Afshar, A., Ziaraty, A., Kaveh. A ve Sharifi. F., 2009. Nondominated Archiving Multicolumn Ant Algorithm in Time–Cost Trade-Off Optimization. Journal of Construction Engineering Management. 135, 7, 668-674.
- Ahmet-Baykal, H., 2006. Discrete time/cost trade-off problem in project scheduling. M.Sc. Thesis, the graduate school of natural and applied sciences, Middle East Technical University, Turkey.
- Aminbakhsh, S., 2013. Hybrid particle swarm optimization algorithm for obtaining Pareto front of discrete time–cost trade-off problem M.Sc. Thesis.Türkiye: Orta Doğu Teknik Üniversitesi, Ankara.
- Aminbakhch, S. ve Sönmez, R., 2016 Applied discrete particle swarm optimization method for the large-scale discrete time-cost trade-off problem. Expert System with Applications 51 177-185
- Anagnostopoulos, K. P. ve Kotsikas, L., 2010. Experimental evaluation of simulated annealing algorithms for the time-cost trade-off problem. Applied Mathematics and Computation, 217, 1 260-270.
- Azeez, M.N. ve Angham, A., 2014. Construction Time-Cost Optimization Modelling Using Ant Colony Optimization. Journal of Engineering, 20, 1,1-18.
- Babu, A. J. G. ve Suresh, N. 1996. Project management with time-cost, and quality considerations. European Journal of Operational Research, 88, 2, 320-327.
- Bettemir, Ö. H., 2009. Optimization of time-cost-resource trade-off problems in project scheduling using meta-heuristic algorithms. Doktora tezi, Orta Doğu Teknik Üniversitesi, Türkiye, Ankara.
- Bettemir, Ö. H. ve Birgönül, T., 2016. Network analysis algorithm for the solution of discrete time-cost trade-off problem, KSCE Journal of Civil Engineering, 21, 4,1047-1058



- Colorni, M., Dorigo, V., Maniezzo ve M. Trubian., 1994. Ant System for Job-Shop Scheduling. Belgian Journal of Operations Research, Statistics and Computer Science (JORBEL), 34, 39–53.
- Creaco E., Franchini M. ve Walski T., 2014. Network design through the phasing of construction approach. Procedia Engineering. 89: 823-830. DOI:10.1016/j.proeng.2014.11.513
- De, P., Dunne, E. J., Ghosh, J. B. ve Wells, C. E., 1997. Complexity of the discrete time-cost trade-off problem for project networks. Operations Research, 45, 2 302-306.
- Deb, K. ve Goldberg, D., E., 1989. “An investigation of niche and species formation in genetic function optimization,” in Proceedings of the Third International Conference on Genetic Algorithms, J. D. Schaffer, Ed. San Mateo, CA: Morgan Kauffman, pp. 42–50.
- Deb, K., Agrawal, S., Pratap, A. ve Meyarivan T., 2000. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II, KanGAL Report No. 200001
- Deb, K., 2001. Multiobjective Optimization using Evolutionary Algorithms. New York: John Wiley and Sons Ltd., İngiltere.
- Demeulemeester, E. L., Herroelen, W. S. ve Elmaghraby, S.E., 1996. Optimal procedures for the discrete time cost trade-off problem in project networks. European Journal of Operational Research, 88, 1, 50-68.
- Demeulemeester, E., De Reyck, B., Foubert, B., Herroelen, W. ve Vanhoucke, M., 1998. New computational results on the discrete time/cost trade-off problem in project networks. Journal of the Operational Research Society, 49, 11, 1153-1163.
- Eirgash, M. A., 2018. “Pareto-Front Performance Of Multiobjective Teaching Learning Based Optimization Algorithm On Time-Cost Trade- Off Optimization Problems.” Yüksek lisans tezi, Karadeniz Teknik Üniversitesi, Türkiye, 110.
- Elbeltagi, E., Hegazy, T. ve Grierson, D., 2005. Comparison among Five Evolutionary-Based Optimization Algorithms, Elsevier Ltd. Advanced Engineering Informatics, 19, 43–53.
- Elbeltagi, E., 2006 Ph.D., P.Eng. Lecture Notes on Construction Management, Chapter 8 164-181
- Elbeltagi, E., Hegazy, T. ve Grierson, D., 2007. A modified shuffled frog-leaping optimization algorithm: Applications to project management. Structure and Infrastructure Engineering- STRUCT INFRASTRUCT ENG. 3. 53-60. 10.1080/15732470500254535.



- Eshtehardian, E., Afshar, A. ve Abbasnia, R., 2008. Time–cost optimization: using GA and fuzzy sets theory for uncertainties in cost. Construction Management and Economics, 26, 7, 679–691.
- Fallah-Mehdipour E, Haddad O.B., Rezapour T. M. ve Mariño M., 2012. Extraction of decision alternatives in construction management projects: Application and adaptation of NSGA-II and MOPSO. Expert Systems With Applications, 39(3): 2794-2803. DOI: 10.1016/j.eswa.2011. 08.139.
- Feng,C. W., Liu, L. L. ve Burns, S. A. 1997. “Using genetic algorithms to solve construction time cost trade off problems” Journal of Computing in Civil Engineering, 11, 3, 184-189.
- Fonseca, C. M. ve Fleming, P. J., 1998 Multiobjective optimization and multiple constraint handling with evolutionary algorithms–Part II: Application example. IEEE Transactions on Systems, Man, and Cybernetics: Part A: Systems and Humans.38-47.
- Fred, G. ve Manuel. L., 1999. Tabu search I. 10.1287/ijoc.1.3.190.
- Gen, M. ve Cheng, R., 2000. Genetic algorithms and engineering optimization. Wiley-International science, New York.
- Ghoddousiet, P., Eshtehardian, E., Jooybanpour, S. ve Javanmardi, A., 2013. Multi-mode resource-constrained discrete time–cost-resource optimization in project scheduling using non-dominated sorting genetic algorithm. Automation in Construction, 30, 216-227.
- Goldberg, D. E. ve Segrest, P., 1987. Finite Markov chain analysis of genetic algorithms. Paper presented at the Proceedings of the Second International Conference on Genetic Algorithms on Genetic algorithms and their application, Cambridge, Massachusetts, United States.
- Golzarpour, B., 2012. Time-Cost Optimization of Large-Scale Construction Projects Using Constraint Programming. Yüksek Lisans Tezi, Waterloo Üniversitesi, Ontario, Kanada.
- Hamdy M., Hasan A. ve Siren K., 2012. A multi-stage optimization method for cost-optimal and nearly-zero-energy building solutions in line with the EPBD-recast 2010. Energy and Buildings. 2013; 56, 1, 189-203. DOI: 10.1016/j.enbuild.2012.08.023.
- Hegazy, T. ve Menesi, W., 2010. Critical Path Segments Scheduling Technique. Journal of Construction Engineering and Management, 136, 10 1078-1094.
- Hegazy, T., 1999. Optimization of resource allocation and levelling using genetic algorithms. Journal of Construction Engineering and Management, 125, 3 167- 175.



- Hegazy, T. ve Moselhi, O., 1994. Analogy-based solution to markup estimation problem. ASCE Journal of Computing in Civil Engineering, 8, 1, 72–87.
- Holland, J. H., 1975. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence: University of Michigan Press.
- Jaskowski, P. ve Sobotka, A., 2006. “Scheduling construction projects using evolutionary algorithms” Journal of Construction Engineering and Management, 132, 8, 861-870.
- Keeney, R. L. ve Raiffa, H., 1976. Decisions with multiple objectives: preferences and value trade-offs. Sydney.
- Khang, D. B. ve Myint, Y. M., 1999. Time, cost and quality trade-off in project management: A case study. International Journal of Project Management, 17, 4, 249-256.
- Kimet, J., Kang, C. ve Hwang, I., 2012. A practical approach to project scheduling: considering the potential quality loss cost in the time–cost trade-off problem. International Journal of Project Management, 30, 2, 264-272.
- Li, H. ve Love, P., 1997. Using improved genetic algorithms to facilitate time–cost optimization. ASCE Journal of Construction Engineering and Management, 123, 3, 233-237.
- Liu, S. S. ve Wang, C. J., 2008. Resource-constrained construction project scheduling model for profit maximization considering cash flow. Automation in Construction, 17, 8, 966-974.
- Magalhaes-Mendes, J., 2015. Multiobjective optimization: Time-Cost Application in Construction. Congresso de Métodos Numéricos em Engenharia, CMN.
- Meyer, W. L. ve Shaffer, L. R., 1963. Extensions of the critical path method through the application of integer programming. University of Illinois.
- Meyer, W. L. ve Shaffer, L. R., 1965. Extending CPM for Multiform Project Time-Cost Curves. Journal of Construction Division, ASCE, 91, 145-68.
- Mohammadi, G., 2011. “Using genetic algorithms to solve industrial time cost tradeoff problems,” Indian Journal of Science and Technology, vol. 4, No. 10, pp. 1273-1278.
- Monghasemi, S., Nikoo, M. R., Fasaee, M. A. K. ve Adamowski, J., 2015. A novel multi criteria decision making model for optimizing time–cost–quality trade-off problems in construction projects. Expert Systems with Applications, 42, 6, 3089-3104.



- Moselhi, O., 1993. Schedule compression using the direct stiffness method. Canadian Journal of Civil Engineering, 20, 65-72.
- Moussourakis, J. ve Haksever, C., 2004. Flexible model for time /cost trade-off problem. Journal of Construction Engineering and Management-ASCE, 130, 3,307-314.
- Mungle, S., Benyoucef, L., Son, Y. J. ve Tiwari, M. K., 2013. A fuzzy clustering-based genetic algorithm approach for time–cost–quality trade-off problems: A case study of highway construction project. Engineering Applications of Artificial Intelligence, 26, 8,1953-1966.
- Ng, S. T. ve Zhang, Y. S., 2008. Optimizing construction time and cost using ant colony optimization approach. Journal of Construction Engineering and Management ASCE, 134, 9,721-728.
- Pathak, B. K. ve Srivastava, S., 2007 “MOGA-based time cost tradeoffs: responsiveness for project uncertainties,” IEEE Congress on evolutionary computation, s.3085-3092.
- Rao, R.V., 2016 Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int J Indust Eng Computat, 7, 1, 19-34.
- Rostami, M., Moradinezhad, D. ve Soufipour, A., 2014. Improved and competitive algorithms for large scale multiple resource constrained project-scheduling problems. KSCE Journal of Civil Engineering, 18, 5, 1261-1269.
- Rudolph, G., 1999 Evolutionary search under partially ordered sets. Teknik Rapor No. CI-67/99, Dortmund: Department of Computer Science/LS11, University of Dortmund, Germany.
- Siemens N., 1971 A Simple CPM Time-Cost Trade off Algorithm. Management Science, 17,354 –363.
- Sönmez, R. ve Bettemir, O.H., 2012. A hybrid genetic algorithm for the discrete time-cost trade-off problem. Expert Systems with Applications, 39, 13,11428-11434.
- Srinivas, N. ve Deb, K. (1995) Multi-Objective function optimization using non dominated sorting genetic algorithms, Evolutionary Computation, 2, 3, 221-248.
- Surajit K.S. ve Sultana P., 2012. GA Based Multi-Objective Time-Cost Optimization in a Project with Resources Consideration. International Journal of Modern Engineering Research, IJMER, 6, 4352-4359.
- Tareghian, H. R. ve Taheri, S. H., 2006. On the discrete time, cost and quality trade-off problem. Applied Mathematics and Computation, 181, 2, 1305-1312.



- Tavana, M., Abtahi, A. R. ve Khalili-Damghani, K., 2014. A new multi-objective multimode model for solving pre-emptive time-cost-quality trade-off project scheduling problems. Expert Systems with Applications, 41, 4 1830-1846.
- Vanhoucke, M., 2005. New computational results for the discrete time/cost trade-off problem with time-switch constraints. European Journal of Operational Research, 165, 2 359-374.
- Xiong, Y. ve Kuang, Y., 2008. Applying an Ant Colony Optimization Algorithm-Based Multiobjective Approach for Time-Cost Trade-Off. Journal of Construction Engineering and Management, 134, 2,153-156.
- Yang, H. H. ve Chen, Y. L., 2000. Finding the critical path in an activity network with time-switch constraints. European Journal of Operational Research, 120, 3 603-613.
- Yang, I. T., 2007b. Performing complex project crashing analysis with aid of particle swarm optimization algorithm. International Journal of Project Management, 25, 6, 637-646.
- Zhang, H. ve Xing, F., 2010. Fuzzy-multi-objective particle swarm optimization for time-cost-quality trade-off in construction. Automation in Construction, 19, 8,1067-1075.
- Zheng, D, Ng, S. ve Kumaraswamy, M., 2005. Applying Pareto Ranking and Niche Formation to Genetic Algorithm-Based Multiobjective Time-Cost Optimization. Journal of Construction Engineering and Management, 131, 1 81-91.
- Zheng, D.X.M., Ng, S.T. ve Kumaraswamy, M.M., 2004. Applying a genetic algorithm based multiobjective approach for time-cost optimization. Journal of Construction Engineering and Management, 130, 2, 168 - 176.
- Zitzler E., Deb K. ve Thiele L., 2000. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. Evol Comput, 8, 2, 173-195.



ÖZGEÇMİŞ

İlköğretimini 2011 yılında Oum alqura Lisesinde tamamladı. İlkokul ve lise eğitimini FAS'ta tamamladıktan sonra, Gaziantep'te Hasan Kalyoncu Üniversitesi Mühendislik Fakültesi İnşaat Mühendisliği Bölümünü kazandı. 2016 yılında lisans eğitimini bu bölümde tamamlayarak aynı yıl Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü İnşaat Mühendisliği Anabilim Dalı'nda başladığı yüksek lisans öğrenimine halen devam etmektedir.

Bekar olup Türkçe, İngilizce, Fransızca, Arapça bilmektedir.

Yayınlar:

Dede, T. ve Kamil, S., 2019. Application of MOO Techniques for the Time-Cost Trade off Problems in Civil Engineering. Current Trends in Civil & Structural Engineering. 1. 10.33552/CTCSE.2019.01.000511.

