

**T.C.
PAMUKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**

**ÜRETİM HATLARINDA SEZGİSEL YÖNTEMLERLE
TAMPON STOK DAĞILIMI OPTİMİZASYONU**

DOKTORA TEZİ

MEHMET ULAŞ KOYUNCUOĞLU

DENİZLİ, ARALIK - 2020

**T.C.
PAMUKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**



**ÜRETİM HATLARINDA SEZGİSEL YÖNTEMLERLE
TAMPON STOK DAĞILIMI OPTİMİZASYONU**

DOKTORA TEZİ

MEHMET ULAŞ KOYUNCUOĞLU

DENİZLİ, ARALIK - 2020

Bu tezin tasarımı, hazırlanması, yürütülmesi, arařtırmalarının yapılması ve bulgularının analizlerinde bilimsel etięe ve akademik kurallara özenle riayet edildiđini; bu alıřmanın dođrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etięe uygun olarak kaynak gösterildiđini ve alıntı yapılan alıřmalara atfedildiđine beyan ederim.



MEHMET ULAŐ KOYUNCUOĐLU

ÖZET

ÜRETİM HATLARINDA SEZGİSEL YÖNTEMLERLE TAMPON STOK DAĞILIMI OPTİMİZASYONU

DOKTORA TEZİ

MEHMET ULAŞ KOYUNCUOĞLU

PAMUKKALE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI

(TEZ DANIŞMANI: DR. ÖĞR. ÜYESİ LEYLA DEMİR)

DENİZLİ, ARALIK - 2020

İmalat sistemlerinin akıllı tasarımı, şirketlerin karlılığını doğrudan etkilediği ve çok maliyetli olduğu için endüstride ve akademide önemli bir araştırma alanıdır. Seri üretim hatları imalat endüstrisinde en çok karşılaşılan sistemlerden biridir ve bu hatların optimum tasarımı sistem performansını doğrudan etkiler. Bir üretim hattında önceden tanımlanmış kısıtlar altında, tampon stokların optimum kapasitesinin ve konumunun belirlenmesi olarak bilinen tampon stok dağılımı problemi (TSDP), NP-Zor bir kombinatoriyal optimizasyon problemidir. Tampon stokların kapasitesi ile hattın üretim oranı arasında cebirsel bir ilişkinin olmaması bu tasarım problemini çözmeyi zorlaştırmaktadır. Hattaki makinelerin rastgele bozulmaları sebebiyle problem aynı zamanda stokastik bir yapıya sahiptir ve problem boyutu arttıkça çözüm alanı da üstel bir şekilde büyür. Problemin bu zorluklarının üstesinden gelmek için probleme özgü sezgisel veya meta-sezgisel arama algoritmaları TSDP'yi çözmek için yaygın olarak kullanılmaktadır. Bu doktora tez çalışmasında, güvenilir olmayan seri üretim hatlarında TSDP'yi çözmek için değişken komşuluk arama (DKA) ve adaptif büyük komşuluk arama (ABKA) gibi komşuluk-tabanlı iki arama algoritması ve popülasyon-tabanlı olan savaştı genetik algoritma (SGA) ve büyük patlama-büyük çöküş (BP-BÇ) algoritması önerilmiştir. Literatürde TSDP'yi çözmek için bu algoritmalarından hiçbirinin daha önce kullanılmadığı görülmüştür. TSDP'nin çözümünde yeni algoritmalar önermenin yanı sıra, önerilen komşuluk tabanlı çözüm algoritmalarında arama eforunu azaltmak için üç yeni başlangıç sezgiseli önerilmiştir. Ayrıca, önerilen algoritmalarda arama etkinliğini artırmak için bir yerel arama stratejisi önerilmiştir. Önerilen çözüm algoritmalarının performansları literatürde mevcut karşılaştırma problemleri üzerinde test edilmiştir. Deneysel çalışmalar, önerilen tüm algoritmaların güvenilir olmayan seri üretim hatlarında TSDP için kaliteli çözümler elde etmede oldukça etkili olduğunu göstermiştir. Ayrıca, sadece önerilen başlangıç sezgisellerinin değil, aynı zamanda önerilen yerel arama stratejisinin de arama süreci üzerinde önemli bir iyileştirme sağladığı gözlemlenmiştir.

ANAHTAR KELİMELER: Tampon Stok Dağılımı Problemi, Seri Üretim Hatları, Değişken Komşuluk Arama, Adaptif Büyük Komşuluk Arama, Savaştı Genetik Algoritma, Büyük Patlama-Büyük Çöküş Algoritması, Başlangıç Sezgiselleri.

ABSTRACT

OPTIMAL BUFFER ALLOCATION IN PRODUCTION LINES USING HEURISTIC METHODS

PH.D THESIS

MEHMET ULAŞ KOYUNCUOĞLU

PAMUKKALE UNIVERSITY INSTITUTE OF SCIENCE

INDUSTRIAL ENGINEERING

(SUPERVISOR:ASSIST. PROF. LEYLA DEMİR)

DENİZLİ, DECEMBER 2020

Intelligent design of manufacturing systems is an important research area in industry and academia because it directly affects the profitability of companies and is very costly. Serial production lines are one of the most encountered systems in manufacturing and the optimal design of production lines directly affects the system performance. The buffer allocation problem (BAP), known as determining the optimal capacity and location of the buffers under predefined constraints in a production line, is an NP-Hard combinatorial optimization problem. Moreover, solving this design problem is difficult because there is no algebraic relationship between the capacity of the buffers and production rate of the line. The problem has also a stochastic nature due to random failures of the machines in the line, and as the problem size increases the solution area also grows exponentially. To overcome these difficulties of the problem, problem-specific heuristics or meta-heuristic search algorithms are widely used to solve the BAP. In this Ph.D. study, two neighborhood-based search algorithms; i.e. variable neighborhood search (VNS) and adaptive large neighborhood search (ALNS) algorithms, and two population-based algorithms; i.e. combat genetic algorithm (CGA) and big bang-big crunch (BB-BC) algorithm are adapted to solve the BAP in unreliable serial production lines. It should be noted that none of these algorithms has been employed to solve the BAP before in the literature. Besides proposing new algorithms in solving the BAP, three new initialization heuristics are introduced in order to reduce the search effort for the proposed neighborhood-based solution algorithms. Moreover, a local search strategy is proposed to improve the search efficiency in the proposed algorithms. The performances of the proposed solution algorithms have been tested on existing benchmark problems taken from the literature. Computational studies have demonstrated that all the proposed algorithms are highly effective in finding good-quality solutions for the BAP in unreliable serial production lines. Moreover, it has been observed that not only the the proposed initialization heuristics but also the proposed local search strategy have significant improving effect on the search process.

KEYWORDS: Buffer Allocation Problem, Serial Production Lines, Variable Neighborhood Search, Adaptive Large Neighborhood Search, Combat Genetic Algorithm, Big Bang-Big Crunch Algorithm, Initialization Heuristics.

İÇİNDEKİLER

Sayfa

ÖZET	i
ABSTRACT	ii
İÇİNDEKİLER	iii
ŞEKİL LİSTESİ	vi
TABLO LİSTESİ	vii
KISALTMALAR LİSTESİ	viii
ÖNSÖZ	x
1. GİRİŞ	1
1.1 Tezin Amacı	1
1.2 Araştırma Metodolojisi.....	3
1.3 Tezin Tasarımı.....	4
2. TAMPON STOK DAĞILIMI PROBLEMİ	6
2.1 Giriş	6
2.2 Üretim ve Üretim Sistemi Tasarımı	6
2.3 Üretim Hatlarının Sınıflandırılması.....	7
2.4 Seri Üretim Hatlarında Tampon Stok Kullanımı	11
2.5 Tampon Stok Dağılımı Problemi	13
2.5.1 Tampon Stok Dağılımı Probleminin Özellikleri.....	14
2.5.2 Tampon Stok Dağılımı Probleminin Sınıflandırılması.....	15
2.5.3 Tampon Stok Dağılımı Probleminin Çözümünde Kullanılan Yöntemler	17
2.6 Özet	20
3. LİTERATÜR ARAŞTIRMASI	22
3.1 Giriş	22
3.2 TSDP’de Kullanılan Optimizasyon Algoritmaları	27
3.2.1 Sayma Yöntemi.....	27
3.2.2 Pratik Kural Önerileri	28
3.2.3 Dinamik Programlama.....	29
3.2.4 Matematiksel Programlama: Doğrusal Programlama, Karma Tamsayı Programlama, Doğrusal Olmayan Programlama	29
3.2.5 Sezgisel Algoritmalar	31
3.2.6 Meta-Sezgisel Yöntemler	33
3.3 TSDP’de Kullanılan Değerlendirme Algoritmaları.....	36
3.3.1 Analitik Yöntemler	36
3.3.2 Simülasyon.....	38
3.4 Özet	39
4. METODOLOJİ	41
4.1 Giriş	41
4.2 Ayırıştırma Yöntemi.....	41
4.3 Değişken Komşuluk Arama Algoritması	45
4.4 Büyük Komşuluk Arama Algoritması.....	50
4.5 Savaşçı Genetik Algoritma.....	56
4.6 Büyük Patlama – Büyük Çöküş Algoritması	59
4.7 Özet	62

5. TAMPON STOK DAĞILIMI PROBLEMİ İÇİN DEĞİŞKEN KOMŞULUK ARAMA TABANLI BİR ÇÖZÜM YAKLAŞIMI	64
5.1 Giriş	64
5.2 Önerilen Başlangıç Çözüm Sezgiselleri	64
5.2.1 Rastgele Başlangıç Çözümü	65
5.2.2 Modifiye Edilmiş Kâse Fenomeni Sezgiseli.....	65
5.2.3 PaVi Sezgiseli	68
5.3 Önerilen DKA Algoritması	70
5.3.1 Komşuluk Yapıları.....	70
5.3.2 Yerel Arama.....	72
5.3.3 Kabul ve Durdurma Kriteri.....	73
5.4 Deneysel Çalışmalar	74
5.4.1 Üretim Oranı Maksimizasyonu.....	75
5.4.2 Toplam Tampon Stok Miktarı Minimizasyonu	82
5.5 Bulgular ve Tartışma	85
5.6 Sonuç	85
6. BOZULMAYA MARUZ KALAN ÜRETİM HATLARINDA ÜRETİM ORANI MAKSİMİZASYONU İÇİN ADAPTİF BÜYÜK KOMŞULUK ARAMA ALGORİTMASI	87
6.1 Giriş	87
6.2 Önerilen Başlangıç Çözüm Sezgiselleri	88
6.3 Çıkarma ve Ekleme Sezgiselleri.....	90
6.3.1 Konum Tabanlı Çıkarma-Ekleme Operatörü.....	90
6.3.2 Durum Tabanlı Çıkarma-Ekleme Operatörü	92
6.4 Adaptif Mekanizma	94
6.5 IncDec Stratejisi	95
6.6 Kabul Kriteri.....	95
6.7 Durdurma Kriteri	96
6.8 Deneysel Çalışmalar	97
6.8.1 Küçük Boyutlu Problemler	98
6.8.2 Orta Büyüklükte Problemler	99
6.8.3 Büyük Boyutlu Problemler	102
6.9 Bulgular ve Tartışma	108
6.10 Sonuç	110
7. TAMPON STOK DAĞILIMI PROBLEMİNİN ÇÖZÜMÜNDE POPÜLASYON TABANLI ARAMA ALGORİTMALARI: SAVAŞÇI GENETİK ALGORİTMA VE BÜYÜK PATLAMA-BÜYÜK ÇÖKÜŞ ALGORİTMASI	111
7.1 Giriş	111
7.2 Savaşçı Genetik Algoritma.....	111
7.3 Büyük Patlama – Büyük Çöküş Algoritması	117
7.4 Deneysel Çalışmalar	121
7.4.1 Parametre Optimizasyonu	122
7.4.2 Karşılaştırma Problemleri İçin Deneyler	123
7.5 Bulgular ve Tartışma	133
7.6 Sonuç	133
8. SONUÇ VE ÖNERİLER	134
8.1 Tezin Özeti	134
8.2 Bulgular	136
8.3 Katkıları	137

8.4	Gelecekte Yapılabilecek Çalışma Önerileri	138
9.	KAYNAKLAR.....	139
10.	ÖZGEÇMİŞ.....	160

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1: Üretim hatlarının özelliklerine göre sınıflandırılması (Li ve diğ. 2009)	8
Şekil 2.2: K makine ve $K-1$ tampon stok alanından oluşan bir seri üretim hattı ..	12
Şekil 2.3: 3 makineli hat için toplam tampon stok miktarı ve maksimum üretim oranı ilişkisi.....	15
Şekil 2.4: TSDP'nin genel çözüm süreci.....	18
Şekil 4.1: Ayrıştırma yöntemi	42
Şekil 4.2: DKA'da çözüm uzayının taranması	47
Şekil 4.3: Kapasite kısıtlı ARP için çıkarma ve ekleme operatörü uygulama örneği	52
Şekil 5.1: Sağ rotasyon komşuluk yapısı.....	70
Şekil 5.2: Sol rotasyon komşuluk yapısı.....	71
Şekil 5.3: Ters çevirme komşuluk yapısı.....	71
Şekil 5.4: Yer değiştirme komşuluk yapısı.....	71
Şekil 5.5: 40 makineli hat için önerilen DKA-MEKF'nin performans değerlendirmesi.....	80
Şekil 6.1: Önerilen konum tabanlı çıkarma-ekleme operatörü koşulları.....	91
Şekil 6.2: Konum tabanlı çıkarma-ekleme operatörü: 5 makineli hat örneği.....	92
Şekil 6.3: Durum tabanlı çıkarma-ekleme operatörü: 5 makineli hat örneği	94
Şekil 6.4: Önerilen ABKA algoritması ve FA karşılaştırması	106
Şekil 6.5: ABKA'da kullanılan başlangıç sezgisellerinin karşılaştırması	107
Şekil 7.1: 10 makineli hat için iki noktalı çaprazlama operatörünün uygulanması	114
Şekil 7.2: 10 makineli hat için önerilen mutasyon operatörlerinin uygulanması	115
Şekil 7.3: BP-BÇ algoritması: 3 makineli hat örneği	121
Şekil 7.4: 10000 değerlendirme için SGA ve BP-BÇ algoritmasının karşılaştırması	131
Şekil 7.5: 80 makineli hat için önerilen BP-BÇ algoritmasıyla elde edilen en iyi tampon stok konfigürasyonu	132

TABLO LİSTESİ

Sayfa

Tablo 3.1: TSDP literatürüne genel bakış	24
Tablo 5.1: 5 makineli hattın güvenilirlik parametreleri (Ho ve diğ. 1979).....	75
Tablo 5.2: 9 makineli hattın karşılaştırmalı sonuçları.....	76
Tablo 5.3: 20 makineli hattın güvenilirlik parametreleri (Li ve diğ. 2015)	77
Tablo 5.4: 20 ve 40 makineli hatların karşılaştırmalı sonuçları	79
Tablo 5.5: Üretim oranı maksimizasyonu için deneysel çalışmaların özeti.....	81
Tablo 5.6: Wilcoxon işaretli-sıra testi	81
Tablo 5.7: 12 makineli hat (Park 1993).....	83
Tablo 5.8: 12 makineli hattın karşılaştırmalı sonuçları.....	83
Tablo 5.9: 10 makineli blok hat (Gershwin ve Goldis 1995).....	84
Tablo 5.10: Büyük boyutlu üretim hattının karşılaştırmalı sonuçları	84
Tablo 6.1: 5 makineli hattın güvenilirlik parametreleri (Ho ve diğ. 1979).....	98
Tablo 6.2: 5 makineli hattın karşılaştırmalı sonuçları.....	99
Tablo 6.3: 9 makineli hattın güvenilirlik parametreleri ve karşılaştırmalı sonuçları.....	101
Tablo 6.4: 10 makineli hattın güvenilirlik parametreleri (Nahas ve diğ. 2006). 102	
Tablo 6.5: 10 makineli hattın karşılaştırmalı sonuçları.....	102
Tablo 6.6: 20 makineli hattın güvenilirlik parametreleri (Li ve diğ. 2015)	103
Tablo 6.7: 20 ve 40 makineli hatların karşılaştırmalı sonuçları	105
Tablo 6.8: Deneysel sonuçların özeti	109
Tablo 6.9: Wilcoxon işaretli-sıra testi	110
Tablo 7.1: SGA ve BP-BÇ algoritmasının optimum parametre değerleri	122
Tablo 7.2: 3 makineli hattın güvenilirlik parametreleri (Gershwin ve Schor 2000)	123
Tablo 7.3: 3 makineli hattın karşılaştırmalı sonuçları.....	123
Tablo 7.4: 5 makineli hattın güvenilirlik parametreleri (Ho ve diğ. 1979).....	124
Tablo 7.5: 5 makineli hattın karşılaştırmalı sonuçları.....	124
Tablo 7.6: 10 makineli hattın güvenilirlik parametreleri (Nahas ve diğ. 2006). 124	
Tablo 7.7: 10 makineli hattın karşılaştırmalı sonuçları.....	125
Tablo 7.8: 20 makineli hatların karşılaştırmalı sonuçları.....	125
Tablo 7.9: 40 makineli hatların karşılaştırmalı sonuçları.....	126
Tablo 7.10: 20 makineli hattın güvenilirlik parametreleri (Li ve diğ. 2015)	127
Tablo 7.11: Büyük boyutlu hatların karşılaştırmalı sonuçları.....	129
Tablo 7.12: IncDec stratejisinin önerilen algoritmalar üzerindeki etkisi	129
Tablo 7.13: Friedman testi	132
Tablo 8.1: DKA-MEKF, ABKA-DKF, SGA ve BP-BÇ algoritmalarının karşılaştırması.....	136

KISALTMALAR LİSTESİ

ABKA	:	Adaptif Büyük Komşuluk Arama
ALNS	:	Adaptive Large Neighborhood Search
ARP	:	Araç Rotalama Problemi
ATSS	:	Aday Tampon Stok Konfigürasyonu Sayısı
AY	:	Ayrıştırma Yöntemi
BAP	:	Buffer Allocation Problem
BAS	:	Blocking-After-Service
BB-BC	:	Big Bang-Big Crunch
BBS	:	Blocking-Before-Service
BOSGA	:	Baskın Olmayan Sıralı Genetik Algoritma
BOSGA-II	:	Baskın Olmayan Sıralı Genetik Algoritma-II
BP-BÇ	:	Büyük Patlama-Büyük Çöküş
CE	:	Centre Elements
CGA	:	Combat Genetic Algorithm
CONWIP	:	CONstant Work In Process
CPU	:	Central Process Unit
DC	:	Degraded Ceiling
DDX	:	Dallery-David-Xie
DKA	:	Değişken Komşuluk Arama
DKF	:	Düzleştirilmiş Kâse Fenomeni
DP	:	Doğrusal Programlama
DTA	:	Dağıtım Tahmini Algoritması
DTL-TS	:	Demir-Tunalı-Løkktangen-Tabu Search
FA	:	Fast Algorithm
GA	:	Genetik Algoritma (Genetic Algorithm)
GG	:	Gershwin-Goldis
GSP	:	Gezgin Satıcı Problemi
IncDec	:	Increment-Decrement
İBB	:	İşleme Bağlı Bozulma
İTA	:	İndirgenmiş Tavan Algoritması
KKO	:	Karınca Kolonisi Optimizasyonu
KY	:	Komşuluk Yapısı
LEE	:	Left Edge Elements
LNS	:	Large Neighborhood Search
l-opt	:	local-optimum
MA	:	Miyopik Algoritma
MEKF	:	Modifiye Edilmiş Kâse Fenomeni
MGA	:	Melez Genetik Algoritma
MTBF	:	Mean Time Between Failures
MTTR	:	Mean Time To Repair
NP/TS	:	Nested Partitions/Tabu Search
NRGA	:	Non-dominated Ranked Genetic Algorithm
NSGA-II	:	Non-dominated Sorting Genetic Algorithm-II
PaVi	:	Papadopoulos ve Vidalis (2001)
PSO	:	Parçacık Sürü Optimizasyonu
REE	:	Right Edge Elements
SA	:	Simulated Annealing

SGA	:	Savaşçı Genetik Algoritma
SKP	:	Sıralı Karasel Programlama
SÖB	:	Servis Öncesi Bloklanma
SSB	:	Servis Sonrası Bloklanma
TA	:	Tabu Arama
TB	:	Tavlama Benzetimi
TS	:	Tabu Search
TSDP	:	Tampon Stok Dağılımı Problemi
VNS	:	Variable Neighborhood Search
VTO	:	Vekil Tabanlı Optimizasyon
WIP	:	Work-In-Process
YSA	:	Yapay Sinir Ağı
ZBB	:	Zamana Bağlı Bozulma

ÖNSÖZ

Her şeyden önce, doktora ders sürecinde ve doktora tez çalışması sürecinde beni gitmek istediğim yönde destekleyen, her zaman olumlu yaklaşan, mesleki ve hayat tecrübesinden faydalandığım çok değerli danışman hocam Dr. Öğr. Üyesi Leyla DEMİR'e rehberliği ve teşviki için teşekkür ederim. Çalışma ve üretme konusunda şevkimin kırılmasına müsaade etmeyen ve yolumu her kaybettiğimde bana ışık olan yönlendirmeleriyle beni hep cesaretlendirmiştir.

Lisans eğitimimin ilk yılından bugüne kadar ve doktora tezi boyunca değerli görüş ve önerilerini eksik etmeyen Prof. Dr. Aşkın GÜNGÖR'e, özellikle uyguladığım yöntemler konusunda detaylı anlatımlarıyla yol gösterici olan Doç. Dr. Şener AKPINAR'a teşekkürlerimi sunarım.

Ayrıca, tez savunma jürime katılmayı kabul eden Prof. Dr. Deniz TÜRSEL ELİİYİ ve Doç. Dr. Hacer GÜNER GÖREN'e şükranlarımı sunarım.

Tez çalışmam boyunca anlayışını esirgemeyen, çok yoğun tez yazma sürecimde evdeki bütün sorumluluğu alan, sağlığımı önceleyen ve beni her daim dinleyen hayat arkadaşım Pelin KOYUNCUOĞLU'na en derin duygularıyla teşekkür ederim.

Desteğini hep hissettiğim aileme ve çalışma arkadaşlarıma teşekkür ederim.

Mehmet Ulaş Koyuncuoğlu
Denizli, Aralık, 2020

1. GİRİŞ

1.1 Tezin Amacı

Bir üretim sistemi birbirine seri halde bağlanmış makineler ve bu makinelerin arasındaki tampon stok alanlarından meydana gelir. Üretim hattında parçalar ilk makineden son makineye doğru sıra ile işlenerek bitmiş ürün halinde hattı terk ederler. Birim zamanda hattan çıkan ürün miktarı üretim hattının ana performans göstergesi olan üretim oranını belirler. Bu performans göstergesini olabildiğince yüksek tutmak için üretim hattının etkin şekilde tasarlanması gerekir. Etkin üretim hattı tasarımında ise birbiriyle çelişen farklı optimizasyon problemlerinin çözülmesi işletmelerin karşılaştığı başlıca sorunlardan biridir.

Seri üretim hatlarının performansı makinelerin güvenilirlik parametreleri ve değişken işlem sürelerinden etkilenir. Üretim hattı üzerindeki bu etkileri azaltmak ve makinelerin mümkün olduğunca birbirinden bağımsız çalışmasını sağlamak amacıyla makinelerin arasında tampon stok kullanılır. Üretim hattında tampon stok kullanmak hem fiziksel alan gereksiniminden dolayı kısıtlıdır hem de maliyetlidir. Üretim hattındaki makineler arasında kullanılacak tampon stokların yerinin ve miktarının belirlenmesi olarak bilinen tampon stok dağılımı problemi (TSDP) bu doktora tezinin ana konusudur.

Bu tez çalışmasında makinelerin bozulmaya maruz kaldığı seri üretim hatlarında tampon stok dağılımı probleminin en iyi çözümü araştırılmıştır. Seri üretim hatlarında makine bozulmaları ve tamir faaliyetlerinden kaynaklanan duruşlar veya stokastik işlem süreleri nedeniyle üretimde meydana gelen dalgalanmalar hattın etkinliğinin azalmasına neden olur. Hattın etkinliğini azaltan bu olumsuz olayların etkisini engellemek veya minimuma indirmek için üretim hatlarındaki makinelerin arasında belli miktarda tampon stok bulundurulmasına izin verilir. Tampon stokların kullanımıyla makinelerin aç kalmasından ya da bloke olmasından kaynaklanan duruşlar engellenmiş ve böylece âtil süre azaltılmış olur. Diğer yandan, tampon stok

için fiziksel alan ayırmanın bir maliyeti vardır ve bu maliyet ile tampon stok bulundurmanın sağlayacağı fayda arasında bir denge kurmak işletme yöneticileri açısından önemlidir. Bu nedenle tampon stok dağılımı problemi bir optimizasyon problemi olarak karşımıza çıkmaktadır. Tezin amacı, rastgele bozulmalara maruz kalan güvenilir olmayan seri üretim hatlarının tasarımı için etkin algoritmalar geliştirerek en iyi tampon stok dağılımının bulunması ve bu sayede üretim hatlarının daha etkin ve verimli bir şekilde çalıştırılmasını sağlayarak işletmelerin rekabet gücünün artırılmasına katkı sağlamaktır.

Seri üretim hatlarının etkin bir şekilde tasarlanması problemi hem akademi hem de imalat endüstrisi için hala önemli ve karmaşık bir araştırma alanı olduğundan, bu tez çalışması genel olarak güvenilir olmayan seri üretim hatlarında üretim oranı maksimizasyonuna odaklanmaktadır. Bu bağlamda, bu tez çalışmasının ilgili bilimsel yazına katkılarını aşağıdaki gibi ifade etmek mümkündür:

- Literatürde TSDP'nin çözümünde daha önce uygulanmamış 4 algoritma problemin yapısına uygun bir şekilde uyarlanmıştır. Bu algoritmalar, Değişken Komşuluk Arama (DKA; *Variable Neighborhood Search: VNS*) algoritması, Adaptif Büyük Komşuluk Arama (ABKA; *Adaptive Large Neighborhood Search: ALNS*) algoritması, Savaşçı Genetik Algoritma (SGA; *Combat Genetic Algorithm: CGA*) ve Büyük Patlama-Büyük Çöküş (BP-BÇ; *Big Bang-Big Crunch: BB-BC*) algoritmasıdır.

- Önerilen DKA algoritmasında, klasik DKA algoritmasından farklı olarak, komşuluk yapıları arama etkinliğini arttırmak için adaptif bir şekilde kullanılmıştır.

- ABKA algoritmasını TSDP'nin çözümünde kullanmak üzere iki yeni çıkarma-ekleme operatör çifti önerilmiş ve adaptif bir şekilde kullanılmıştır.

- Önerilen bu iki komşuluk tabanlı arama algoritmasının harcadığı arama eforunu azaltmak üzere üç farklı başlangıç çözümü oluşturma sezgiseli önerilmiştir.

- SGA ve BP-BÇ algoritması sürekli karşılaştırma fonksiyonları yerine ilk defa bir üretim hattı tasarım problemine uyarlanmıştır.

- Popülasyon tabanlı olan SGA ve BP-BÇ algoritması, TSDP literatüründe çoğunlukla simülasyonla beraber kullanılan popülasyon tabanlı algoritmaların aksine bu çalışmada analitik bir değerlendirme algoritmasıyla birlikte kullanılmıştır.

- Ayrıca çözüm uzayında aramanın yoğunlaştırılması için bir yerel arama sezgiseli önerilmiş ve bu tez kapsamında önerilen tüm algoritmaların ihtiyaç duyulan aşamalarında kullanılmıştır.

1.2 Araştırma Metodolojisi

Makinelerin rastgele bozulmalara maruz kaldığı güvenilir olmayan seri üretim hatlarının performansını artırmak için tampon stokların hat boyunca nasıl dağıtılacağına karar verilmesi problemi bu tez çalışmasının konusunu oluşturmaktadır. Tampon stok dağılımları arasında en uygun olan konfigürasyona karar vermek için bir performans değerlendirme yöntemi ve bir optimizasyon yöntemi kapalı formda yinelemeli olarak uygulanmalıdır. Optimizasyon yöntemi kullanılarak elde edilen tampon stok konfigürasyonuna değerlendirme yöntemi uygulanarak hattın performansı ölçülmektedir. Bu işlem önerilen optimizasyon algoritmasının durdurma koşulu sağlanana kadar yinelenerek optimum tampon stok konfigürasyonu elde edilmektedir. Bu tez çalışması kapsamında hattın performans ölçütleri yaklaşık bir değerlendirme yöntemi olan ayrıştırma yöntemi (AY; *decomposition method: DM*) kullanılarak belirlenmiştir. Optimizasyon yöntemi olarak 4 farklı meta-sezgisel algoritma problemin yapısına uygun bir şekilde uyarlanmıştır.

TSDP gibi kombinatoriyal optimizasyon problemlerini çözmek için birçok meta-sezgisel algoritma kullanılsa da Niyomubeyi ve diğ. (2020) tarafından belirtildiği gibi, belirli bir probleme en uygun olan algoritmayı bulma sorusu tatmin edici bir şekilde cevaplanmamıştır. Ayrıca, çözümlerde çeşitliliğin sağlanması ve erken yakınsama meta-sezgisel algoritmalar için hala kritik konulardır (Jourdan ve diğ. 2009, Gogna ve Tayal 2013, Niyomubeyi ve diğ. 2020). Genel olarak, bir meta-sezgisel algoritmanın performansı büyük ölçüde ele alınan probleme ve algoritmada kullanılan parametre değerlerine bağlıdır (Mobin ve diğ. 2018, Joshi ve Bansal 2020). Bu nedenle, çeşitlilik ve yakınsama açısından güçlü yeni algoritmaların, optimize edilmiş parametrelerle genetik algoritma ve tabu arama gibi klasik meta-sezgisel algoritmalara veya literatürdeki en son bilinen algoritmalara karşı performanslarının araştırılması gerekmektedir. TSDP literatüründe, farklı özellikteki sezgisel ve meta-sezgisel algoritmalar TSDP'yi çözmek için kullanılmış olsa da komşuluk tabanlı

arama algoritmaları olan DKA ve ABKA algoritmaları ile popülasyon tabanlı olan SGA ve BP-BÇ algoritmalarının problemin çözümünde kullanılmadığı görülmüştür.

Hem problemin hem de DKA ve ABKA algoritmalarının komşuluk yapıları/operatörleri ile SGA ve BP-BÇ algoritmasının çeşitlilik ve yakınsama hızı gibi özellikleri dikkate alındığında, bu yöntemlerin TSDP için iyi kalitede çözümler üretmesi beklenmektedir. DKA, ABKA, SGA ve BP-BÇ algoritmaları, genetik algoritma (GA), tabu arama (TA) ve tavlama benzetimi (TB) gibi diğer meta-sezgisel algoritmalara kıyasla daha az parametre ile çalışırlar. Ayrıca TSDP literatüründen de bilindiği üzere, TA (Demir ve diğ. 2011, 2012) ve TB (Spinellis ve Papadopoulos 2000, Altıparmak ve diğ. 2002) gibi komşuluk tabanlı sezgisel yöntemler TSDP'nin çözümünde oldukça başarılıdır. Diğer taraftan, SGA ve BP-BÇ algoritmalarının performansları çeşitlilik (*diversity*) ve yakınsama hızı (*convergence speed*) açısından karşılaştırılmak istenmektedir. BP-BÇ algoritması, TA algoritmasındaki çeşitlendirme (*diversification*) ve yoğunlaştırma (*intensification*) stratejilerine benzeyen büyük patlama (*big bang*) ve büyük çöküş (*big crunch*) aşamalarına sahip olduğundan, BP-BÇ algoritmasının performansının TA algoritmasına benzer veya ondan daha iyi olması beklenmektedir. Bu nedenle tez çalışması kapsamında, bu argümanları araştırmak için BP-BÇ algoritması ve SGA, TA algoritması ile de karşılaştırılmıştır.

Tez kapsamında TSDP'nin çözümü için önerilen tüm algoritmaların performansı literatürdeki karşılaştırma problemleri üzerinde test edilmiş, çözüm kalitesi açısından literatürde mevcut algoritmalar ile karşılaştırılmış ve elde edilen sonuçlar istatistiksel olarak analiz edilmiştir.

1.3 Tezin Tasarımı

Bu tez çalışması aşağıdaki şekilde organize edilmiştir:

İkinci bölümde üretim ve üretim sistemi tasarımı kavramları açıklanarak üretim hatlarının karakteristiklerine değinilmiştir. Daha sonra tezin konusu olan TSDP'nin özellikleri, sınıflandırılması ve problemin çözümünde kullanılan yöntemler detaylı olarak anlatılmıştır.

Üçüncü bölümde, TSDP için yapılan literatür araştırmasının ayrıntılarına yer verilmiştir. Literatürde TSDP'nin çözümünde kullanılan optimizasyon ve değerlendirme algoritmaları ayrı ayrı ele alınmış ve bu alanda yapılan çalışmalar detaylandırılmıştır. Okuyucuların TSDP alanında yapılmış bu çalışmalarını literatürden daha rahat takip edebilmeleri amacıyla, tez boyunca sık kullanılan terimlerin İngilizce karşılıkları parantez içinde verilmiştir.

Dördüncü bölümde, tez çalışması kapsamında TSDP'nin çözümü için önerilen DKA, ABKA, SGA ve BP-BÇ algoritmalarının temel özelliklerine ve literatürdeki uygulamalarına yer verilmiştir.

Beşinci bölümde, TSDP'nin çözümü için önerilen DKA algoritmasının ayrıntıları verilmiştir. Bu tez çalışmasında önerilen DKA ve ABKA algoritmalarına rastgele çözüm dışında üç farklı başlangıç çözümü ile başlanması önerilmiştir. Başlangıç çözümü üretme sezgisellerinden ikisi bu bölümde diğeri ise altıncı bölümde tanıtılmıştır. Bu bölümde DKA'nın komşuluk yapıları ve yerel arama algoritmasının detayları sunulurken önerilen DKA algoritmasının literatürdeki karşılaştırma problemleri üzerindeki performansı detaylı olarak analiz edilmiştir.

Altıncı bölümde, önerilen ABKA algoritmasının detaylarına yer verilmiş ve yeni bir başlangıç sezgiseli önerilmiştir. ABKA algoritmasını TSDP'ye uyarlayabilmek için iki yeni çıkarma-ekleme operatör çifti önerilmiş ve adaptif bir şekilde kullanılmıştır. Önerilen ABKA algoritmasının performansı literatürdeki karşılaştırma problemleri üzerinde test edilmiş ve elde edilen bulgular ayrıntılı bir şekilde tartışılmıştır.

Yedinci bölümde ise TSDP, üretim oranı maksimizasyonu amacı ile popülasyon tabanlı olan SGA ve BP-BÇ algoritması ile çözülmüştür. Önceki bölümde olduğu gibi önerilen algoritmaların performansları hem birbirleri ile hem de literatürdeki mevcut diğer yöntemler ile kıyaslanmıştır.

Son olarak sekizinci bölümde, deneysel çalışmalar sonucunda elde edilen bulgular tartışılarak tezin bir özeti sunulmuştur. Bu bölümde ayrıca bu doktora tez çalışmasının literatüre katkısı vurgulanarak, gelecekte yapılabilecek çalışma önerilerine yer verilmiştir.

2. TAMPON STOK DAĞILIMI PROBLEMİ

2.1 Giriş

Tampon stok dağılımı problemi bir üretim sisteminde iş istasyonlarının arasında bulundurulması gereken stok miktarlarına odaklanmış bir tasarım problemidir. Bu tez çalışmasının amacı bozulmaların meydana geldiği seri üretim hatlarında tampon stok dağılımı probleminin çözümü için etkin algoritmalar geliştirmektir. Bu nedenle, bu bölümde öncelikle problemin ele alındığı seri üretim hatları ve özellikleri tanıtıldıktan sonra tampon stok dağılımı problemi ayrıntılı bir şekilde ele alınmıştır.

Bölüm 2.2’de üretim ve üretim sistemi tasarımı kavramlarına kısaca değinilmiştir. Ardından, Bölüm 2.3’te üretim hatlarının sınıflandırılmasına yer verilmiştir. Bölüm 2.4’te seri üretim hatlarında tampon stok kullanımına ve bazı temel formüllere yer verilmiştir. Son olarak, Bölüm 2.5’te tampon stok dağılımı probleminin özellikleri, sınıflandırılması ve problemin çözümünde kullanılan yöntemler detaylı olarak anlatılmış ve Bölüm 2.6’da bu bölümün bir özeti sunulmuştur.

2.2 Üretim ve Üretim Sistemi Tasarımı

Üretim kelimesi köken olarak Latince “ortaya çıkarmak” veya “sevketmek” anlamına gelen “*Pro Ducere*” ve “el yapımı” anlamına gelen “*Manus Factus*” kelimelerinden türemiştir (Sohlenius 2000). Üretim terminolojik olarak, önceden tanımlanmış bir programa göre belirli süreçler aracılığıyla birtakım girdilerden çıktı elde etmek anlamına gelir. Bir üretim sistemi, fiziksel imalat altyapısı, insan organizasyonu ve bilgi-kontrol mimarisinden oluşur (Wu 2001). Üretim sistemi yönetimi, bir üretim sistemini yaşam döngüsü boyunca düzenlemek ve optimize etmek için gerekli olan tasarım, uygulama ve izleme gibi ana faaliyetleri kapsayan fonksiyonel bir alandır (Wu ve Chang 2000).

Üretim, dünya ekonomisi ve refahı için vazgeçilmez bir unsurdur (Li ve Meerkov 2009). Üretimin ekonomideki önemi nedeniyle son 50 yılda bu alanda

dikkate değer sayıda çalışma yapılmıştır. Bu çalışmaların önemli bir kısmında, üretim sistemi mühendisleri ve bilim insanları fabrikaların etkinliğini artırmak üzere tasarım ve operasyon alanlarına odaklanmışlardır (Gershwin 2000, Gershwin 2018).

Üretim sistemleri teknolojinin de gelişmesiyle beraber, daha karmaşık, bilgisayar destekli ve otomasyona dayalı hale gelmiştir (Kusiak 2018). Üretim sistemi tasarlamak, şirketlerin sürdürülebilir ekonomisini doğrudan etkileyen uzun vadeli stratejik kararlardan biri olarak bilinmektedir (Waibel ve diğ. 2017). Bu yüzden, tasarım kararı şirket yöneticileri tarafından genellikle bir kez alınır. Bu tasarım kararının kalitesi üretim hattının en temel performans göstergelerinden biri olan birim zamanda üretilen ürün miktarıyla (ürün/birim zaman) ölçülür. Üretim hattının tasarımındaki ufak bir değişikliğin kârlılık üzerinde büyük etkisi vardır.

Bir üretim sistemi bir dizi makine/operatör, transfer elemanı, bilgisayar, tampon stok (*buffer*), iş yükü ve üretim için gerekli diğer bileşenlerden meydana gelir (Gershwin 1994). Birbirlerine seri şekilde bağlanmış ve tampon stoklarla ayrılmış makinelerin/hücrelerin oluşturduğu hatlara seri üretim/transfer/akış hattı denir (Bihan ve Dallery 2000, Shi ve Gershwin 2009). Kuyruk teorisi (*queueing theory*) açısından bir seri üretim hattı, sınırlı (*finite*) tampon stoktan ve art arda sıralı (*tandem*) kuyruk sisteminden meydana gelir (Dallery ve Gershwin 1992, Li ve diğ. 2016). Seri üretim hatları genellikle, elektronik ürün (Mak 1986, Matta ve diğ. 2012), otomobil (Tempelmeier 2003) ve gıda ürünleri (Liberopoulos ve Tsarouhas 2002) gibi yüksek hacimli tek ürün üreten, kesikli parça akışı olan (Dallery ve Gershwin 1992), bozulmalara maruz kalan ve sınırlı tampon stok alanı içeren otomatik hatlardır (Li ve diğ. 2009, Li ve Meerkov 2009, Alfieri ve Matta 2012). Bu tip hatlar, tüm parçaların hat boyunca sırayla işlendiği ve itme (*push*) tipi üretim sisteminin geçerli olduğu hatlardır (Papadopoulos ve Vidalis 2001).

2.3 Üretim Hatlarının Sınıflandırılması

Üretim hatları etkin bir üretim için işlenecek parçaların özellikleri de dikkate alınarak tasarlanır. Sabit veya değişken süreyle parçaların işlendiği makineler zamana veya operasyona bağlı olarak bozulmalara maruz kalabilir. Bu sebeple üretim hattındaki tampon stok alanlarında sınırsız veya sınırlı miktarda tampon stok

depolamak gerekebilir. Gerçek üretim sistemlerinde tampon stok alanı genellikle sınırlıdır. Bir üretim hattındaki makinelerin işlem süreleri birbirlerinden farklı olabileceği gibi, makineler birbirinin aynı işlem sürelerine de sahip olabilirler. İşlemlerde operatörlerin kullanıldığı hatlarda işlem süreleri değişkenlik gösterirken, makinelerin kullanıldığı otomatik hatlarda işlem süreleri genellikle aynıdır. Ayrıca parçaların hat üzerindeki akışı üretilen ürünün özelliğine göre sürekli veya kesikli olabilir (Li ve diğ. 2009) (Şekil 2.1).



Şekil 2.1: Üretim hatlarının özelliklerine göre sınıflandırılması (Li ve diğ. 2009)

Hat topolojisi: Üretim sistemleri imal edilen ürünün özelliğine ve üretim sürecine göre tasarlanırlar. En yaygın hat topolojileri seri (S), seri-paralel (S-P), montaj/de-montaj (M/D), genel ağ sistemi (GAS), esnek ve hücresel üretim hatlarıdır. Seri üretim hatları genellikle yüksek üretim hacmine sahip, tek tip ürün imal eden hatlardır ve bu hatlarda üretim süresi genellikle deterministiktir (Tempelmeier 2003).

Bozulma süresi, tamir süresi ve bozulma tipleri: Üretim sistemlerinde makinelerin bozulması öngörülebilir bir durumdur. Bir veya birden fazla makinede bozulma meydana geldiğinde ilgili makineler parçaları işleyemez ve tekrar çalışabilmeleri için tamir işlemi gerekir. Bir makine üretim hattında dört durumda bulunabilir: meşgul/çalışıyor (*busy/working*), bozulmuş/tamirde (*failed/under repair*),

bloke (*blocking*) veya açlık (*starving*) (Glassey ve Hong 1993). Dallery ve Bihan (1999) ise bloke ve açlık durumlarını âtil/boşta olarak sınıflandırıp ‘çalışıyor’ durumu içine dâhil etmişlerdir. Bir makinenin aynı zamanda hem aç hem de bloke durumda olması çok küçük bir olasılıktır ve ihmal edilebilirdir (Gershwin 1987). Makinelerin bozulmaları arasında geçen ortalama süre bozulma oranını, bozulma durumundan tekrar operasyonel duruma geçmesi arasındaki ortalama süre ise tamir oranını ifade eder. Bozulma ve tamir oranları deterministik olabileceği gibi hattın karakteristiğine göre olasılıksal bir dağılıma da sahip olabilirler. Makineler temelde iki tip bozulmaya maruz kalır: işleme bağlı bozulma (İBB; *operation dependent failure: ODF*) ve zamana bağlı bozulma (ZBB; *time dependent failure: TDF*) (Buzacott ve Hanifin 1978). İBB parçanın işlenmesinden kaynaklıdır ve makine çalışır durumdayken (malzeme kırılması vb.) meydana gelir. ZBB ise parçanın işlenmesinden bağımsız, makinenin yorulması sonucu makine âtil durumdayken bile (örneğin; elektronik kontrol sisteminde) herhangi bir zamanda meydana gelebilen arızalardır (Buzacott ve Hanifin 1978). Hanifin, 1975 yılında yaptığı doktora tezinde, Chrysler otomotiv firmasındaki gözlemleri sonucunda büyük hacimli üretimin yapıldığı seri üretim hatlarında büyük çoğunlukla İBB’lerin meydana geldiğini tespit etmiştir.

Bir seri üretim hattının performansı makinelerin bozulma ve tamir oranı gibi güvenilirlik parametreleri ile değişken işlem sürelerinden etkilenmektedir (Battini ve diğ. 2009). Rastgele işlem süreleri işgücü yoğun üretim hatlarında sıklıkla görülürken, sabit işlem süreleri daha çok otomasyona dayalı üretim hatlarında karşımıza çıkmaktadır (Li ve diğ. 2009). Bu tip üretim hatları için sabit işlem süreleri daha makul bir varsayımdır. Bir üretim hattında makineler rastgele bozuluyorsa bu tür hatlara güvenilir olmayan (*unreliable*) üretim hattı, makinelerin bozulmaya maruz kalmadığı hatlara ise güvenilir (*reliable*) üretim hattı adı verilmektedir (Spinellis ve Papadopoulos 2000). Güvenilir hatlarda duruşlar sadece makinelerin işlem süreleri arasındaki farklılıktan kaynaklanırken, güvenilir olmayan üretim hatlarında hem işlem sürelerinden hem de makine bozulmalarından kaynaklı duruşlar yaşanabilmektedir. Makinelerin güvenilirliğinden kaynaklanan farklılıklar nedeniyle hatların performans analizinde farklı yöntemler kullanılmaktadır. Güvenilir hatların (GH) performans analizleri güvenilir olmayan hatların analizinden (GOH) daha kolaydır.

İşlem/operasyon süresi: Bir makinenin bir parça üzerinde bir işlemi yapması için gerekli olan zaman olarak tanımlanır. İşlem süresi deterministik (sabit) olabileceği gibi belli bir olasılık dağılımına (üstel, geometrik, vb.) uygun rassal bir değişken ile de temsil edilebilir (Dallery ve Gershwin 1992).

Hat dengesi: Bir üretim hattındaki bütün makinelerin üretim süreleri eşitse veya bu süreler birbirinden bağımsız ve eş dağılmış rassal değişkenler ile ifade edilebiliyorsa, hat dengeli hat (*balanced line*), eğer süreler eşit değilse veya sürelerin dağılımları birbirinden farklı ise dengeli olmayan hat (*unbalanced line*) olarak isimlendirilir. Makinelerin tamamının bozulmaya maruz kalıp kalmaması hat dengesi durumunu değiştirmez.

Bloklanma durumu: Tampon stok kapasitesi gerçek üretim hatlarında sınırlı (*finite*) olduğu için bloke durumları sıkça yaşanır. Üretim hatlarında iki tip bloke durumu vardır: Servis Sonrası Bloklanma (SSB; *Blocking-after-service: BAS*) ve Servis Öncesi Bloklanma (SÖB; *Blocking-before-service: BBS*) (Perros 1989). Bir makinenin devamındaki tampon stok alanı makineden çıkan parçayı alamayacak halde tamamen doluyorsa, tampon stok alanı uygun duruma gelene kadar ilgili parça makinenin üzerinde bekler. Bu durum literatürde SSB durumu olarak tanımlanır. SÖB durumunda ise, bir makine ancak sonrasındaki tampon stok alanı uygun durumda olduğunda parçayı işlemeye başlar. Yani sonraki tampon stok alanı dolu olduğu sürece makine bloklanmış halde bekler. Üretim hatlarının çoğu SSB durumuna göre işler, bu nedenle çalışmalarda genellikle ele alınan üretim sisteminde SSB durumu olduğu varsayılır (Dallery ve Gershwin 1992).

Hat/Model doygunluğu: Genellikle, ilk makinenin önünde hammadde depolanması için sınırsız alan olduğu ve son makineden sonra bitmiş ürünlerin depolanması için sınırsız alan olduğu varsayılır. Bu varsayımın geçerli olduğu hatlar doymuş hat olarak tanımlanırken, bu üretim hattının modeli doymuş model (*saturated model*) olarak isimlendirilir. Diğer taraftan, hammaddenin gelişler arası süresinde değişkenlik var ise ilk makine aç kalabilir. Benzer şekilde, son makinenin devamındaki bitmiş ürünlerin depolanacağı alan yetersiz kalabilir ve bu nedenle son makine bloke olabilir. İlk makinenin açlık, son makinenin bloke durumu yaşandığı hatlar doymamış hat olarak tanımlanırken, bu üretim hattının modeli doymamış model (*unsaturated model*) olarak isimlendirilir (Weiss ve diğ. 2019).

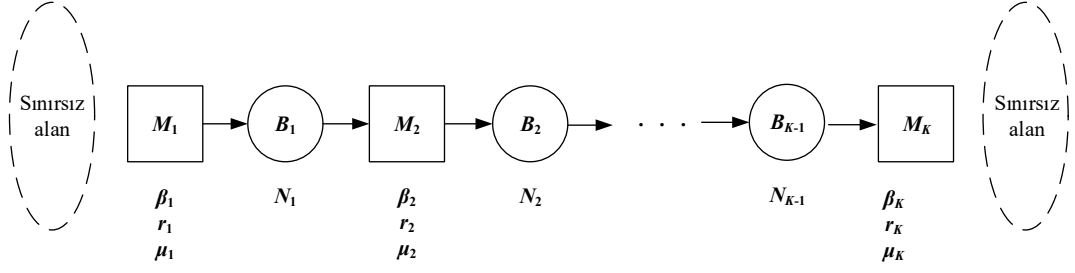
Hat büyüklüğü: Papadopoulos ve diğ. (2009) hatları iki tip olarak tanımlamışlardır: küçük boyutlu hatlar ve büyük boyutlu hatlar. Keskin bir sınır çizilmemekle beraber, maksimum 6 makineye ve maksimum 20 tampon stok miktarına kadar olan hatlar küçük boyutlu, daha çok makineye ve tampon stok miktarına sahip hatlar ise büyük boyutlu hat olarak tanımlanmaktadır.

Parça akış modeli: Üretim sistemlerinde malzemelerin hat boyunca akışı kesikli (*discrete*) veya sürekli (*continuous*) olabilir. Optimizasyon problemlerinin yapısı sürekli modellere daha uygun olmasına rağmen kesikli modeller fabrikaların kesikli yapısını daha iyi temsil etmektedir (Gershwin ve Schor 2000). Kombinatorial karmaşıklık nedeniyle kesikli optimizasyon problemlerini çözmek sürekli optimizasyon problemlerini çözmekten daha zordur.

2.4 Seri Üretim Hatlarında Tampon Stok Kullanımı

Bir seri üretim hattı birbirine seri olarak bağlanmış iş istasyonları ve tampon stok alanlarından oluşur. Şekil 2.2’de K adet iş istasyonundan oluşan ve her iş istasyonunda tek bir makinenin olduğu bir seri üretim hattı görülmektedir. Bu üretim hattında işler makineler üzerinde sıra ile işlenirler. Şekil 2.2’de kare şeklinde resmedilen öğeler makineleri (*machine: M*), daire şeklindeki öğeler ise tampon stok alanlarını (*buffer: B*) göstermektedir. Ek olarak, N_i tampon stok alanı B_i ’nin kapasitesini göstermektedir. Makinelere dair diğer notasyonlar ise aşağıdaki gibidir:

- β_i , i . makinenin bozulma oranını; $MTBF_i$ i . makinenin bozulmalar arası geçen ortalama süreyi (*MTBF: mean time between failures*)
- r_i , i . makinenin tamir oranını; $MTTR_i$ i . makinenin tamiri için harcanan ortalama süreyi (*MTTR: mean time to repair*)
- μ_i , i . makinenin işlem oranını göstermektedir.



Şekil 2.2: K makine ve $K-1$ tampon stok alanından oluşan bir seri üretim hattı

- i . makinenin izolasyondaki etkinliği/kullanılabilirliği (*efficiency in isolation: availability*) Denklem 2.1'deki gibi hesaplanmaktadır:

$$e_i = \frac{r_i}{\beta_i + r_i} \quad (2.1)$$

Güvenilir makineler için $e_i=1$ 'e eşittir, güvenilir olmayan makineler için ise $e_i < 1$ 'dir. Ayrıca, güvenilir makineler için i . makinenin izolasyondaki etkinliği/kullanılabilirliği ilgili makinenin işlem oranına eşittir (Sabuncuoğlu ve diğ. 2006).

- i . makinenin ortalama etkin servis oranı (*mean effective service rate*) Denklem 2.2'deki gibi hesaplanmaktadır:

$$\rho_i = \frac{\mu_i r_i}{\beta_i + r_i} \quad (2.2)$$

$\mu_i=1$ olması durumunda i . makinenin izolasyondaki etkinliği ortalama etkin servis oranına eşit demektir.

Bir üretim hattında, bütün tampon stok alanları sınırsız kapasiteye sahip olursa ($\mu_i=1$ için), hattın maksimum üretim oranı hattaki makinelerden en küçük ortalama etkin servis oranına (Denklem 2.3) sahip olan makineye eşit olur (Buzacott 1967):

$$f(\infty) = \frac{1}{1 + \max\{\beta_i/r_i\}} \quad i = 1, 2, \dots, K \quad (2.3)$$

Eğer bir üretim hattındaki makineler arasında hiç tampon stok alanı bulunmazsa açlık ve bloke durumları oluşabilir ve bu da sistemin durmasına neden

olur. Tampon stok bulundurulmasına izin verilmeyen bir hattın ($\mu_i=1$ için) üretim oranı Denklem 2.4'teki gibi hesaplanır (Buzacott 1968):

$$f(0) = \frac{1}{1 + \sum_i^K \frac{\beta_i}{r_i}} \quad (2.4)$$

2.5 Tampon Stok Dağılımı Problemi

Bir üretim hattının tasarımı temelde iş istasyonlarına atanacak sunucuların (*servers*) (operatörler/makineler/hücreler), her bir tampon stok alanına atanacak tampon stok miktarlarının büyüklüğünün ve istasyonlara yapılacak iş yükü atamalarının belirlenmesini kapsar (Papadopoulos ve diğ. 2009). Tampon stok dağılımı problemi bu üç temel tasarım probleminden biridir. Hattın performansını en üst düzeye çıkaracak şekilde üretim hattındaki iş istasyonları/makinelerin arasına atanacak tampon stokların büyüklüklerinin ve yerlerinin optimum değerini bulmayı amaçlayan problem literatürde tampon stok dağılımı problemi (*Buffer Allocation Problem: BAP*) olarak isimlendirilmektedir.

Tampon stok kullanımının amacı makinelerin mümkün olduğunca birbirinden bağımsız çalışmasını sağlamak ve böylelikle üretim hattından mümkün olan en yüksek üretim miktarını elde etmektir. Tampon stok makinelerin bozulma ve tamir süreleri üzerinde etkili değildir ve sadece bloke ve açlık olasılıklarını azaltmaya yardımcı olur (Kiesmüller ve Sachs 2020). Öte yandan tampon stok alanı kullanımı fiziksel alan gerektirmektedir ve bu da ek sermaye maliyetine yol açmaktadır. Ayrıca tampon stoklar süreç içi stok (*work-in-process: WIP*) miktarının artmasına sebep olduğu için bu stok miktarlarının çok fazla olması istenmez. Dolayısıyla hattın performansı ile tampon stok kullanmanın getirdiği maliyetler arasında bir denge kurmak önemlidir. Bu nedenle, üretim hatları tasarlanırken tampon stok bulundurup bulundurmama kararı hem maliyet hem de hattın beklenen performans göz önünde bulundurularak verilmelidir.

2.5.1 Tampon Stok Dağılımı Probleminin Özellikleri

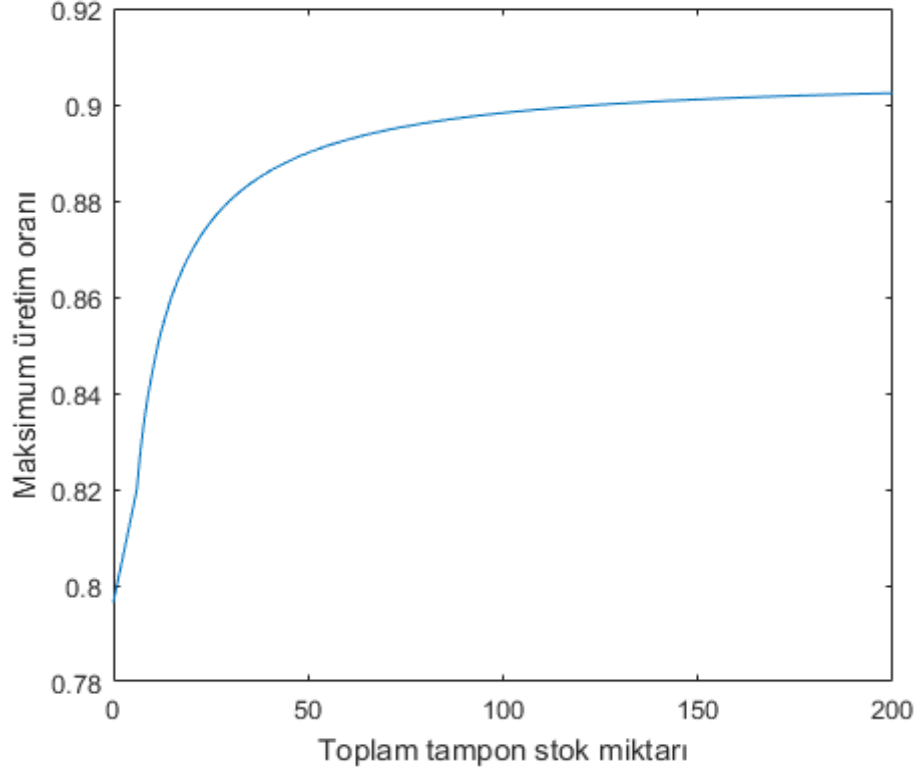
TSDP'nin doğası gereği klasik arama algoritmalarının problemi çözmeye başarısız olmasına neden olan bazı özellikleri vardır (Park, 1993):

Süreklilik (*Continuity*): Herhangi bir tampon stok alanının kapasitesindeki küçük değişiklik üretim oranının değişmesine neden olmaktadır (Gershwin ve Schor 2000). Park (1993) farklı tampon stok alanlarında tampon stok miktarının artırılması durumunda üretim oranının her zaman kesin bir şekilde artmayabileceğini, hatta bazen üretim oranı grafiğinin bir veya birden fazla kez durağan şekilde seyredebileceğini vurgulamıştır. Üretim oranındaki bu durağanlığın TSDP'nin global optimum çözümünü bulmayı zorlaştırdığını ifade etmiştir.

Monotonluk (*Monotonicity*): Tampon stok miktarındaki artış üretim hattının performansını monoton bir şekilde artırmaktadır (Okamura ve Yamashina 1977). Ayrıca, tampon stoku sınırsız kullanmak üretim oranında azalan bir artışa sebep olmakla birlikte, belli bir tampon stok miktarından sonra üretim oranı değişmemektedir (Conway ve diğ. 1988).

İçbükeylik (*Concavity*): Üretim oranı tampon stok konfigürasyonunun içbükey bir fonksiyonudur (Meester ve Shanthikumar 1990). Yani tampon stok boyutundaki bir birim artışa bağlı olarak üretim oranındaki artış, tampon stok boyutu arttıkça azalır (Gershwin ve Schor 2000).

Şekil 2.3'te 3 makineli bir hat üzerinde Gershwin ve Schor (2000) tarafından üretim oranı maksimizasyonu için ve Wang ve diğ. (2016) tarafından kâr maksimizasyonu için önerilen yöntemlere benzer basit bir gradyan yöntemi kullanılarak elde edilen bir grafik verilmiştir (veriler için bkz. Bölüm 7.4.2). Bu grafiğin yatay eksenini toplam tampon stok miktarını, dikey eksenini ise ilgili tampon stok miktarlarına karşı gelen üretim oranını göstermektedir. Grafik oluşturulurken üretim oranının kullanılan toplam tampon stok miktarı ile nasıl değiştiğini gözlemlemek amacıyla ile tampon stok miktarları 0 ile 200 arasında birer adet artırılmıştır. Şekil 2.3'ten görüldüğü gibi, toplam tampon stok miktarı artarken üretim oranı monoton (*monotonicity*), iç bükey (*concavity*) ve sürekli (*continuity*) bir şekilde artmaktadır (Gershwin ve Schor 2000).



Şekil 2.3: 3 makineli hat için toplam tampon stok miktarı ve maksimum üretim oranı ilişkisi

Literatürde büyük boyutlu seri üretim hatları için monotonluk ve içbükeylik özellikleri Gershwin ve Schor (2000), Li ve Meerkov (2009) ve Wang ve diğ. (2016)'nin çalışmalarında kanıtlanmıştır. Toplam tampon stok miktarındaki her artış üretim oranını belli bir seviyeye kadar azalan bir oranda sürekli şekilde artırır.

2.5.2 Tampon Stok Dağılımı Probleminin Sınıflandırılması

TSDP literatüründe 4 ana performans ölçütü ve bunlarla ilişkili olarak tanımlanmış gelir ve maliyet ölçütleri bulunmaktadır (Weiss ve diğ. 2019):

- (i) Hattın üretim oranı (Th): birim zamanda son makineden (M_K) çıkan bitmiş ürün miktarını
- (ii) Süreç içi stok (WIP): WIP_i , tampon stok alanı i 'de depolanan ortalama iş parçasının sayısını göstermek üzere, bu tampon stok alanında ve diğer komşu

tampon stok alanlarında hattın akış modeline bağlı olarak depolanan ortalama iş parçası sayısı,

(iii) Çevrim süresi (*cycle time*): iş parçalarının hatta geçirdiği ortalama süre,

(iv) Ortalama sipariş bekleme süresi, siparişlerin gecikmesi ve siparişlerin beklenmeden karşılanma olasılığı dâhil olmak üzere doymamış talebe sahip hatlar için müşteri hizmetlerinin ölçümü.

Bu dört performans ölçütü maliyetlerle ilişkilendirilerek TSDP kâr maksimizasyonu ve maliyet minimizasyonu şeklinde de ifade edilebilmektedir.

Burada Gershwin ve Schor (2000) tarafından yapılan ve TSDP literatüründe genel kabul görmüş olan tanıma dayanarak TSDP'nin iki matematiksel modeli tanıtılacaktır. Bunlardan ilki literatürde dual problem ya da BAP1 olarak bilinen üretim oranı maksimizasyonu (*throughput maximization*) problemidir. İkincisi ise primal problem ya da BAP2 olarak bilinen hattaki toplam stok miktarının minimize (*total buffer size minimization*) edildiği problemidir.

Dual Problem-Üretim Oranı Maksimizasyonu: Verilen belirli bir miktardaki toplam tampon stok kısıtı altında üretim oranını maksimize edecek tampon stok konfigürasyonunu bulmayı hedefleyen problem dual problem olarak isimlendirilmektedir. Dual problemin matematiksel modeli aşağıda verilmiştir.

Maks $Th(N)$

$$\sum_{i=1}^{K-1} N_i = N \quad (2.5)$$

$N_i \geq 0$ ve tam sayı

Burada K makine sayısını, N toplam tampon stok miktarını, N_i negatif olmayan tamsayı olmak koşuluyla her bir alandaki tampon stok miktarını, $Th(N)$ üretim oranını göstermektedir.

Primal Problem-Toplam Tampon Stok Minimizasyonu: Hattan beklenen belirli bir üretim oranının altına düşmeyecek şekilde hattaki toplam tampon stok miktarını minimize etmeyi amaçlayan bu problemin matematiksel modeli aşağıdaki gibidir.

$$\begin{aligned} \text{Min } N &= \sum_{i=1}^{K-1} N_i \\ Th(N) &\geq Th^* \end{aligned} \quad (2.6)$$

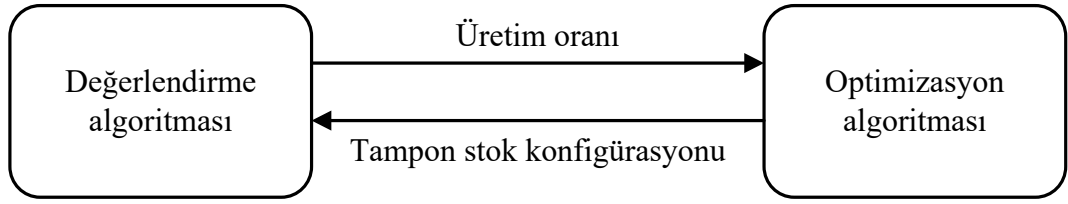
$N_i \geq 0$ ve tam sayı

Burada, K makine sayısını, N toplam tampon stok miktarını, N_i negatif olmayan tamsayı olmak koşuluyla her bir alandaki tampon stok miktarını, $Th(N)$ üretim oranını ve Th^* hattan beklenen minimum üretim oranını göstermektedir. Yani, gerçek üretim oranı $Th(N)$, beklenen üretim oranı Th^* 'den küçük olmamalıdır.

2.5.3 Tampon Stok Dağılımı Probleminin Çözümünde Kullanılan Yöntemler

TSDP literatüründe iki temel araştırma konusu vardır: performans değerlendirme ve optimizasyon (Weiss ve Stoltetz 2015). Performans değerlendirme çalışmaları tampon stok kullanımının izin verildiği hatlarda hattın performans ölçütlerini hesaplama yöntemlerine odaklanırken, optimizasyon çalışmaları ise belirli kısıtlar altında en iyi tampon stok konfigürasyonunu elde etmeye odaklanır.

TSDP, üretim oranı maksimizasyonu olarak ele alındığında doğrusal olmayan bir tamsayılı sırt çantası problemidir (Smith ve Chikhale 1995, Singh ve Smith 1997, Chaharsooghi ve Nahavandi 2003, Smith ve Cruz 2005) ve *NP-Zor (NP-Hard)* problem sınıfındadır (Dolgui ve diğ. 2013, Demir ve diğ. 2014). Ayrıca, değişken üretim süreleri ve makine bozulmaları sebebiyle problem doğası gereği stokastik bir yapıya sahiptir. Diğer yandan, Chow (1987)'un da belirttiği gibi tampon stok miktarları ile üretim oranı arasında cebirsel bir ilişki bulunmaması problemi daha da zorlaştırmaktadır. Bu zorluğun üstesinden gelmek için TSDP'nin çözümünde bir değerlendirme algoritması ve bir optimizasyon algoritması Şekil 2.4'te gösterildiği gibi, yinelemeli bir biçimde ve kapalı formda çalıştırılmalıdır (Papadopoulos ve diğ. 2009). Değerlendirme algoritması ile belirli bir tampon stok konfigürasyonu kullanılarak hattın performansı hesaplanırken, optimizasyon algoritması ile optimum ya da en iyi tampon stok konfigürasyonu belirlenmeye çalışılır.



Şekil 2.4: TSDP'nin genel çözüm süreci

Değerlendirme Algoritmaları:

Üretim hatlarının performanslarının kesin (*exact*) yöntemlerle değerlendirilmesi sadece küçük boyutlu hatlarda (en fazla 4 makine) ve belli varsayımlar altında mümkündür (Smith ve Cruz 2005, Li ve Meerkov 2009). Daha büyük hatların performanslarını değerlendirmede yaklaşık yöntemler (*approximation methods*) veya simülasyon kullanılmaktadır.

TSDP literatüründe en çok kullanılan yaklaşık analitik yöntemler ayrıştırma yöntemi (*decomposition method: AY*), toplama yöntemi (*aggregation method*), ve genişleme yöntemidir (*expansion method*).

AY, bir seri üretim hattını iki makineli alt hatlara bölerek bütün bir hattın performansını bulmaya yarayan ve literatürde en çok kullanılan analitik değerlendirme yöntemidir. TSDP literatüründe ayrıştırma yöntemini kullanan çok sayıda çalışma mevcuttur. Gershwin ve Schor (2000), Shi ve Men (2003), Tempelmeier (2003), Noureifath ve diğ. (2005), Nahas ve diğ. (2006), Shi ve Gershwin (2009), Massim ve diğ. (2010) ve Demir ve diğ. (2011, 2012) TSDP çözümünde değerlendirme yöntemi olarak AY'yi kullanmışlardır.

Toplama yönteminde ise, tüm hat yine 2 makineli hatlara bölünür ancak ayrıştırma yönteminin tersine her seferinde bu ikili hatları birbirini eklemek suretiyle tüm hat için performans analizi gerçekleştirilir (De Koster 1987). Bu yöntemi kullanan çalışmalara örnek olarak Dolgui ve diğ. (2002, 2007) ve Qudeiri ve diğ. (2008) gösterilebilir.

Genişleme yöntemi ise ayrıştırma ve toplama yöntemlerinden farklı olarak bozulmaların olmadığı üretim hatlarında ve genel dağılıma sahip işlem sürelerinin olduğu durumlarda kullanılan bir yaklaşık analitik yöntemdir (Kerbache ve Smith

1987). Smith ve Cruz (2005) ve Cruz ve diğ. (2012) bu yöntemi seri-paralel üretim hatlarının performansını değerlendirmede kullanmışlardır.

Kuyruk teorisine ve Markov zincirlerine (*Markov chains*) dayanan analitik yöntemler ancak hattaki makinelerin özelliklerine dair belirli varsayımlar geçerli olduğu takdirde kullanılabilir. Varsayımların sağlanamadığı gerçek hayat problemlerinin modellenmesinde simülasyon daha iyi bir seçenektir. Simülasyon kompleks sistemleri modellemek için çok güçlü bir araçtır ancak değerlendirme süresi analitik yöntemlere göre uzundur (Mohtashami 2014, Zhang ve diğ. 2014, Costa ve diğ. 2015, Köse ve Kılınçcı 2015, Köse ve diğ. 2015).

Yukarıda bahsedilen yöntemler dışında literatürde performans değerlendirmede Altıparmak ve diğ. (2002, 2007) ve Matta ve diğ. (2012) tarafından geliştirilen meta-modeller de mevcuttur.

Optimizasyon Algoritmaları:

TSDP'nin çözüm uzayı hattaki makine sayısı ve dağıtılacak toplam tampon stok miktarı arttıkça üstel bir şekilde genişlemektedir. Örneğin; 10 makine ve 50 tampon stoktan oluşan bir hat için toplam uygun tampon stok konfigürasyonu sayısı Denklem 2.7'den 1916797311 olarak elde edilirken (Demir ve diğ. 2014), 20 makine ve 100 toplam tampon stok miktarına sahip bir hat için bu değer $7,84 \times 10^{20}$ 'dir. Çözüm uzayı bu şekilde genişleyen kombinatoriyal optimizasyon problemlerinin çözümü için genel ve/veya probleme özgü sezgisel optimizasyon algoritmalarını kullanmak kaçınılmazdır.

$$\binom{N + K - 2}{K - 2} = \frac{(N + 1)(N + 2) \dots (N + K - 2)}{(K - 2)!} \quad (2.7)$$

TSDP'nin çözümünde kesin yöntemler problemin karmaşıklığı ve üstel olarak artan çözüm uzayı nedeniyle sadece küçük boyutlu problemlere uygulanabilmektedir (Smith ve Cruz 2005). Problemin boyutu arttıkça TSDP'nin çözümünde probleme özgü geliştirilmiş arama algoritmaları ve meta-sezgisel algoritmalar daha yaygın olarak kullanılmaktadır. Genel olarak, TSDP'nin çözümünde kullanılan optimizasyon yöntemleri şu şekilde sıralanabilir: tüm çözümlerin değerlendirildiği sayma yöntemi (*complete enumeration*), doğrusal programlama, karma/tam sayılı programlama,

doğrusal olmayan programlama, dinamik programlama, probleme özgü geliştirilmiş arama algoritmaları ve meta-sezgisel algoritmalar.

Genetik algoritma (GA) (Dolgui ve diğ. 2002, 2007, Amiri ve Mohtashami 2012, Papadopoulos ve diğ. 2013, Tsadiras ve diğ. 2013, Nahas ve diğ. 2014, Köse ve Kılınçcı 2015, Tiacci 2015, Zandieh ve diğ. 2017), tabu arama (TA) (Shi ve Men 2003, Demir ve diğ. 2011, Papadopoulos ve diğ. 2013, Demir ve diğ. 2012, Costa ve diğ. 2015), tavlama benzetimi (TB) (Spinellis ve diğ. 2000, Spinellis ve Papadopoulos 2000, Altıparmak ve diğ. 2002, Nahas ve diğ. 2006, 2009, Papadopoulos ve diğ. 2013), karınca kolonisi optimizasyonu (KKO) (Nourelfath ve diğ. 2005, Nahas ve diğ. 2009, Nahas ve Nourelfath 2018) ve parçacık sürü optimizasyonu (PSO) (Narasimhamu ve diğ. 2014, Xuemei ve diğ. 2017, Zandieh ve diğ. 2017) TSDP'yi çözmek için literatürde kullanılmış başlıca meta-sezgisel algoritmalarıdır.

Optimizasyon algoritmalarına ilişkin daha detaylı bilgiler üçüncü bölümde sunulan literatür araştırmasında verilmiştir.

2.6 Özet

Bu tez çalışmasında bozulmaya maruz kalan seri üretim hatlarında tampon stok dağılımı problemi ele alınmıştır. Tez boyunca ele alınan tüm problemler aşağıdaki varsayımlar altında çözülmüştür:

- Bütün parçalar M_1, M_2, \dots, M_K makinelerinde ilk giren ilk çıkar (*first in first out*) prensibine göre sırayla işlenmektedir.
- Birinci makinenin önünde ve hattın sonunda sınırsız tampon stok alanı olduğu varsayılmaktadır. Bu sebeple birinci makinede asla açlık durumu yaşanmaz ve son makinede de asla bloke durumu yaşanmaz (Gershwin ve Berman 1981).
- Makineler sadece çalışıyor iken bozulur, boшта iken bozulma durumu söz konusu değildir ve makinelerin güvenilirlik parametreleri geometrik dağılıma uymaktadır.
- Makinelerin durum geçiş (*state transition*) süreci Markov özelliğine sahiptir. Yani, bir makinenin gelecekteki durumlarının olasılık dağılımları, önceki olaylardan bağımsız ve sadece mevcut duruma bağlıdır (Zhou ve diğ. 2018).

- Makineler arası parça taşıma/transfer süresi çalışma süresine göre ihmal edilebilir.
- Bütün makinelerin işlem süresinin deterministik ve 1 birim olduğu varsayılmaktadır.
- Bir makine bozulduğunda bekleyen parçalar SSB mekanizmasına göre işlenmektedir.

Yukarıdaki varsayımlara sahip seri üretim hatlarında TSDP, önerilen tüm algoritmalarla üretim oranı maksimizasyonu (dual problem) için çözülmüştür. Ayrıca önerilen DKA algoritması literatürde Demir ve diğ. (2011) tarafından önerilen bir ikili arama algoritması ile birleştirilerek toplam tampon stok minimizasyonu (primal problem) için de uygulanmıştır.

3. LİTERATÜR ARAŞTIRMASI

3.1 Giriş

Tampon stok dağılımı problemi ilk defa Koenigsberg (1959) tarafından ele alınmıştır. Koenigsberg, 1959 yılında yaptığı çalışmada “*üretim hatlarının yöneylem araştırması ve yönetim bilimi alanında yapılan bilimsel çalışmalarda ihmal edildiğini*” vurgulamıştır. Koenigsberg (1959) ayrıca, modern sanayi kuruluşlarının karmaşık matematiksel analizler yapmadan yüksek verimliliğe ulaştığını belirtmektedir. Yazar, çalışmasında bu eksikliklere dikkat çekerek tampon stokun üretim sistemlerinin verimi üzerindeki etkisini irdelemiştir. Çalışmada ele aldığı farklı topolojilerdeki üretim hatlarını makinelerin sayısı, tampon stokların konumu ve boyutlarına göre incelemiştir. Yazar, üretim hattı tasarımında terminolojik bir tanımlama yaparak bu alanın önünü açmıştır. Daha sonra 1962 yılında Barten, incelediği hatlar için simülasyon ve yaklaşık yöntemler kullanarak, bütün tampon stok alanlarındaki tampon stok miktarlarının eşit olduğu ve makinelerin işlem sürelerinin normal dağılıma sahip olduğu varsayımları altında optimum tampon stok miktarı ile maliyet fonksiyonu arasında bir regresyon modeli sunmuştur. 1990’lı yıllardan sonra ise TSDP alanındaki çalışmalar yoğunlaşmış ve üretim sistemlerinin analitik modellemesini ele alan çok sayıda detaylı çalışma yayımlanmıştır (Dallery ve Gershwin 1992, Buzacott ve Shanthikumar 1993, Papadopoulos ve diğ. 1993, Papadopoulos ve Heavey 1996).

TSDP ile ilgili ilk detaylı literatür araştırması Demir ve diğ. (2014) tarafından sunulmuştur. Daha sonra 2019 yılında Weiss ve diğ. (2019) tarafından TSDP’nin formülasyonlarının ve çözüm algoritmalarının sınıflandırıldığı yeni bir literatür araştırması yayımlanmıştır. Bu bölümde Demir ve diğ. (2014)’nin önerdiği sınıflandırma metodolojisi kullanılarak son on yılda bu alanda yapılmış çalışmalar Tablo 3.1’deki gibi listelenmiştir. Bu tez çalışmasının temel hedefi TSDP için meta-sezgisel yöntemler kullanarak etkin algoritmalar geliştirmek olduğu için bu bölümde TSDP alanında yapılmış çalışmalar problemin çözümünde kullanılan yöntemlere göre incelenmiştir.

Bir sonraki bölümde TSDP’de kullanılan optimizasyon algoritmaları yöntemlere göre alt başlıklar halinde kronolojik olarak incelenmiştir. Bölüm 3.3’te

TSDP'nin çözümünde kullanılan değerlendirme algoritmalarına değinilmiştir. Son olarak, literatür incelemesi sonucunda elde edilen bulgular özetlenmiştir.

Tablo 3.1: TSDP literatürüne genel bakış

Yazarlar	Hattın Güvenilirliği	Hat Topolojisi				Amaç Fonksiyonu					Çözüm Yöntemi	
		S	S-P	GAS	M/D	1	2	3	4	5	Değerlendirme algoritması	Optimizasyon algoritması
Demir ve diğ. (2011)	GOH	✓				✓	✓				AY	TA
Demir ve diğ. (2012)	GOH	✓				✓					AY	Adaptif TA
Matta ve diğ. (2012)	GOH	✓					✓				Kriging yaklaşımı	VTO
Amiri ve Mohtashami (2012)	GOH		✓			✓	✓				Simülasyon	Melez GA-Yerel geliştirme yöntemi
Papadopoulos ve diğ. (2013)	GH	✓				✓	✓				AY, Markovian algoritma	GA, TB, MA, TA
Hillier (2013)	GH				✓					✓	Simülasyon	Sezgisel algoritma
Tsadiras ve diğ. (2013)	GH	✓				✓					YSA	YSA
Mohtashami (2014)	GOH				✓	✓		✓			Simülasyon	GA, MGA, BOSGA-II
Nahas ve diğ. (2014)	GOH	✓	✓			✓		✓			AY	GA
Narasimhamu ve diğ. (2014)	GH			✓		✓					GY	PSO
Ouazene ve diğ. (2014)	GOH	✓				✓					Markov süreci	Doğrusal olmayan programlama
Zhang ve diğ. (2014)	GOH	✓				✓	✓				Simülasyon	Melez sezgisel algoritma
Costa ve diğ. (2015)	GH	✓					✓				Simülasyon	Paralel TA
Kolb ve Göttlich (2015)	GOH	✓				✓	✓			✓	Simülasyon	SKP
Köse ve Kılınçcı (2015)	GH, GOH	✓				✓					Simülasyon	GA, TB, GAA
Köse ve diğ. (2015)	GOH	✓				✓	✓				Simülasyon	GA, TB, TA
Li ve diğ. (2015)	GOH	✓				✓					AY	HA, Yerel arama
Tiacci (2015)	GOH				✓					✓	Simülasyon	GA
Weiss ve Stolletz (2015)	GOH	✓					✓				Simülasyon	Benders AY
Smith (2016)	GH			✓		✓					Ortalama değer analizi	Doğrusal olmayan SKP
Shi ve Gershwin (2016)	GOH	✓							✓		Segmentasyon yaklaşımı	Sezgisel algoritma

Tablo 3.1: TSDP literatürüne genel bakış (devam)

Yazarlar	Hattın Güvenilirliği	Hat Topolojisi				Amaç Fonksiyonu					Çözüm Yöntemi	
		S	S-P	GAS	M/D	1	2	3	4	5	Değerlendirme algoritması	Optimizasyon algoritması
Wang ve diğ. (2016)	GH			✓					✓		BY, AY	Gradyan arama
Alferi ve diğ. (2016)	GH	✓					✓				Simülasyon	Sütun oluşturma yöntemi
Su ve diğ. (2017)	GOH	✓			✓	✓	✓				Ayrıştırma uzantılı Markov yöntemi	TA-BOSGA-II
Xuemei ve diğ. (2017)	GH	✓				✓		✓			Simülasyon	GA-PSO
Dolgui ve diğ. (2017)	GOH		✓			✓		✓			Simülasyon	Basit evrimsel çok amaçlı optimizasyon yöntemi
Zandieh ve diğ. (2017)	GOH			✓	✓	✓	✓			✓	Simülasyon	GA, PSO
Weiss ve diğ. (2018)	GOH	✓					✓				Örnekleme	Kural tabanlı yerel arama
Nahas ve Nourelfath (2018)	GOH	✓						✓			AY	GA, KKO, Doğrusal olmayan meta-sezgisel
Zhou ve diğ. (2018)	GOH	✓				✓				✓	BY, AY	GA, İTA, PSO, DTA, PSO-DTA
Pedrielli ve diğ. (2018)	GH	✓						✓		✓	Simülasyon	Matematiksel programlama, Sezgisel alg.
Smith (2018)	GOH	✓	✓	✓		✓					Sıralı ağ ayrıştırma yöntemi	Doğrusal olmayan karışık tamsayılı programlama
Motlagh ve diğ. (2019)	GOH	✓				✓	✓	✓			Simülasyon	BOSGA, BOSGA-II
Xi ve diğ. (2019)	GH		✓				✓				Genelleştirilmiş GY	Ayrıştırma-Koordinasyon Yöntemi
Romero-Silva ve Shaaban (2019)	GOH				✓	✓	✓				Simülasyon	Senaryo bazlı analiz
Lin ve diğ. (2020)	GOH	✓					✓				AY, Simülasyon	VTO, GA

Tablo 3.1: TSDP literatürüne genel bakış (devam)

Yazarlar	Hattın Güvenilirliği	Hat Topolojisi				Amaç Fonksiyonu					Çözüm Yöntemi	
		S	S-P	GAS	M/D	1	2	3	4	5	Değerlendirme algoritması	Optimizasyon algoritması
Köse ve Kılınçcı (2020)	GH, GOH	✓				✓	✓				Simülasyon	BOSGA-II, TB
Liberopoulos (2020)	GH	✓							✓		AY	Gradyan tabanlı arama, Komşuluk temelli arama
Zhang ve Matta (2020)	GOH	✓				✓				✓	Simülasyon	Karışık tamsayılı doğrusal programlama
Alfieri ve diğ. (2020)	GH	✓			✓			✓			Simülasyon	Satır-sütun oluşturma yöntemi
Lopes ve diğ. (2020)	GH				✓	✓					AY	Karışık tamsayılı doğrusal programlama

Hattın Güvenilirliği: GOH (Güvenilir Olmayan Hat), GH (Güvenilir Hat).

Hat Topolojisi: Seri Üretim Hattı (S), Seri-Paralel Üretim Hattı (S-P), Montaj/De-montaj (M/D), Genel Ağ Sistemi (GAS).

Amaç Fonksiyonu: Üretim Oranı Maksimizasyonu (1), Toplam Tampon Stok Miktarı/Süreç İçi Stok Minimizasyonu (2), Maliyet Minimizasyonu (3), Kâr Maksimizasyonu (4), Diğer amaç fonksiyonları (5).

3.2 TSDP’de Kullanılan Optimizasyon Algoritmaları

TSDP’nin çözümünde kullanılan birçok optimizasyon yöntemi mevcuttur. Bu yöntemler olası tüm çözümlerin değerlendirildiği sayma yöntemi, en iyi tampon stok konfigürasyonun nasıl oluşturulacağına dair öneriler sunan pratik kural önerileri (*rule-of-thumbs*), dinamik programlama, matematiksel programlama, sezgisel algoritmalar ve meta-sezgisel algoritmalar olarak sınıflandırılabilirler.

3.2.1 Sayma Yöntemi

Sayma (*complete enumeration*) yöntemi kombinatoriyal karmaşıklık nedeniyle genellikle küçük boyutlu hatlarda olası tüm tampon stok konfigürasyonlarının değerlendirilerek optimum sonucun elde edildiği yöntemdir. Conway ve diğ. (1988), makinelerin üretim sürelerinin düzgün (*uniform*) ve üstel dağılıma sahip olduğu, güvenilir seri üretim hatlarında simülasyon ile birlikte tam sayma yöntemini kullanarak üretim oranı maksimizasyonu problemini ele almışlardır.

McClain ve Moodie (1991), sayma yöntemini kullanarak işlem sürelerinin üstel dağılıma sahip olduğu özdeş makineleri içeren dengeli bir seri üretim hattında, farklı tampon stok dağılımlarına göre üretim oranı ve süreç içi stok miktarlarını incelemişlerdir. Makine sayısı sabitken, aynı miktar toplam tampon stok sayısına sahip hatlarda tampon stokların hattın başına doğru veya eşit dağıtılması durumunda, hattın sonuna atanması durumuna göre daha yüksek üretim oranı verdiğini gözlemlemişlerdir. Diğer taraftan, daha düşük süreç içi stok için hattın sonuna doğru daha fazla tampon stok atanması gerektiği ortaya konmuştur.

Powell ve Pyke (1996) ise dengeli olmayan seri üretim hatları üzerinde tam sayma yöntemi ile hattın üretim oranını maksimize etmek için darboğaz yaşanan makinelerin tampon stok alanlarına ve hattın ortasına daha fazla tampon stok atanması gerektiği sonucuna varmışlardır.

3.2.2 Pratik Kural Önerileri

Pratik kural önerileri hattın performansını optimize edecek tampon stok kapasitesinin ve konfigürasyonunun belirlenmesi için makinelerin izolasyondaki etkinlikleri, ortalama etkin servis oranı gibi bazı temel ölçütleri ve darboğaz makineler, ters kâse fenomeni gibi olguları kullanmaya dayanmaktadır.

Hillier ve Boling (1966, 1977, 1979) değişken işlem süresine sahip dengeli olmayan hatlarda diğer bir üretim tasarım problemi olan iş yükü atama (*workload allocation*) problemini araştırmışlardır. Hillier ve Boling (1966, 1977, 1979) bu çalışmalarında hattın ortasına daha az, kenarlarına daha fazla olmak üzere hattın tamamına simetrik bir şekilde iş yükü dağıtılmasının üretim oranını artırdığını gözlemlemişlerdir. Optimum iş yükü ataması bir kâse şekline benzediği için bu atamayı kâse fenomeni (*bowl phenomenon*) olarak tanımlamışlardır.

Hillier ve diğ. (1993), değişken işlem sürelerine sahip makinelerin oluşturduğu güvenilir seri üretim hattında yaptıkları çalışmada, tampon stok dağılımının hattın ortasına daha fazla, hattın kenarlarına daha az ve simetrik olması durumunda hattın performansının arttığını gözlemlemişlerdir. Optimum tampon stok dağılımı ters kâseye benzediği için bu dağılımı ters kâse fenomeni (*inverted bowl phenomenon*) olarak isimlendirmişlerdir. Buradaki motivasyon, hattın başındaki makinenin asla açlık durumuna düşmemesi ve son makinenin asla bloklanmaması ama ortadaki makinelerin her iki durumu da yaşayabilecek olmasıdır (Conway ve diğ. 1988, Helber 2001).

Staley ve Kim (2012), Powell ve Pyke'ın 1996 yılında güvenilir-güvenilir olmayan seri üretim hatları için kullandığı 4 makineli ve 8 makineli problemleri yeniden ele almışlardır. Yazarlar, CONWIP (*CONstant Work In Process*) kontrol sistemi için tampon stokların eşit bir şekilde dağıtılmasını önermişler ve işlem sürelerinde değişkenlik arttıkça optimum hat tasarımı için tampon stokların darboğaz makineler etrafına daha çok atandığını gözlemlemişlerdir.

3.2.3 Dinamik Programlama

Seri üretim hattında toplam tampon stok miktarı kısıtı altında üretim oranını maksimize etmeye çalışan Chan ve Ng (2002), çalışmalarında farklı tampon stok atama stratejilerini dinamik programlama ile incelemişlerdir. Hattaki makinelerin işlem süreleri hat boyunca artan veya azalan sırada ise eşit tampon stok atama stratejisinin kullanılmaması gerektiği yazarlar tarafından vurgulanmıştır. Yazarlar ayrıca, Chow (1987)'un çalışmasında kullandığı tampon stok atama stratejisinin bazen iyi sonuç verse de darboğaz makinelerin çok olduğu hatlarda kullanılmaktan kaçınılmasının faydalı olacağı görüşünü öne sürmüşlerdir. Literatürde dinamik programlama yöntemini kullanan diğer bir çalışma ise Diamantidis ve Papadopoulos (2004) tarafından yapılmıştır. Yazarlar, güvenilir seri üretim hatlarında üretim oranı maksimizasyonu problemini ele almışlar ve hattın performansını değerlendirmek için toplama yöntemi kullanmışlardır. Optimizasyon algoritması olarak dinamik programlama kullanılan bu çalışmanın sonucunda yöntemin oldukça etkin olduğu ortaya konmuştur.

3.2.4 Matematiksel Programlama: Doğrusal Programlama, Karma Tamsayılı Programlama, Doğrusal Olmayan Programlama

Son yıllarda, matematiksel programlama yaklaşımlarının TSDP'yi etkin bir şekilde çözmek için daha fazla uygulandığı görülmektedir. Ouazene ve diğ. (2014), TSDP için doğrusal olmayan programlamaya dayalı bir çözüm algoritması önermiş ve güvenilir olmayan seri üretim hatlarını içeren karşılaştırma problemlerini kullanarak algoritmanın performansını üretim oranı maksimizasyonu açısından değerlendirmişlerdir. TSDP için sürekli bir model Kolb ve Göttlich (2015) tarafından sunulmuş ve yazarlar problemi sıralı karesel programlama (SKP; *Sequential Quadratic Programming -SQP*) kullanarak çözmüştür.

Weiss ve Stolletz (2015) güvenilir olmayan seri üretim hatlarında toplam tampon stok miktarını minimize etmek için karışık tamsayılı programlama ve Benders AY'yi kombinatoriyal kesmeler ile kullanmışlardır. TSDP'nin çözümünde Benders AY ilk defa maliyet minimizasyonu için Caramanis (1987) tarafından uygulanmıştır.

Smith (2016), güvenilir genel ağ sistemlerinde TSDP'yi çözmek için doğrusal olmayan SKP yöntemini kullanmıştır. Pedrielli ve diğ. (2018), TSDP ve görev atama (*task allocation*) problemlerini güvenilir seri üretim hatlarında eş zamanlı çözmek için bir karışık tamsayılı doğrusal programlama modeli önermişlerdir. Önerilen yaklaşımın TSDP ve görev atama problemini modellemek için uygun olduğu ve karmaşık yapıdaki örnekler için esneklik sağladığı gösterilmiştir. Güvenilir hatlarda farklı ağ topolojileri için eş zamanlı olarak optimum tampon stok dağılımını ve işlem oranlarını belirlemek için Smith (2018) tarafından yine doğrusal olmayan bir SKP algoritması önerilmiştir. Yazar tarafından önerilen bu optimizasyon algoritması bir kuyruk ağı ayrıştırma yöntemi ile birlikte kullanılmış ve hattın üretim oranı maksimize edilmeye çalışılmıştır.

Lopes ve diğ. (2020), eşzamanlı olmayan montaj hatlarında üretim oranını maksimize etmek üzere, TSDP, hat dengeleme ve döngüsel sıralama problemlerini birlikte ele alan bir karışık tamsayılı doğrusal programlama modeli önermişlerdir.

Alfieri ve diğ. (2020), Alfieri ve diğ. (2016) tarafından seri üretim hatları için önerilen sütun üretme algoritmasını genişleterek, güvenilir seri üretim ve montaj/de-montaj hatlarında TSDP'yi çözmek için satır-sütun oluşturma (*row-column generation*) yaklaşımı önermişlerdir. Önerilen yaklaşım, rastgele üretilen problem setleri üzerinde standart bir doğrusal programlama (*linear programming*: DP) çözücüsü (*ILOG CPLEX*) ve karışık tamsayılı doğrusal programlama ile simülasyon kullanılarak karşılaştırılmıştır. Önerilen yaklaşımın çoğu problem setinde hesaplama süresi açısından DP'den daha iyi olduğu ve karışık tamsayılı doğrusal programlama yönteminden daha sağlam olduğu kanıtlanmıştır.

Başka bir karışık tamsayılı doğrusal programlama modeli, güvenilir olmayan seri üretim hatlarında bozulmalardan kaynaklı kesinti süresini (*downtime*) azaltarak üretim oranını iyileştirmek amacıyla Zhang ve Matta (2020) tarafından önerilmiştir. Çalışmada, yazarlar tarafından iki problem tanımlanmış ve çözülmüştür. Birincisi, belirli bir bütçe kısıtı altında üretim oranı maksimizasyonu, ikincisi ise minimum maliyet kısıtı altında sabit bir üretim oranıdır. Önerilen modelin iki modern sezgisel yöntemle karşılaştırılması sonucunda, özellikle amaç fonksiyonu açısından diğer yöntemlere oranla daha iyi sonuçlar ürettiği ortaya konmuştur.

3.2.5 Sezgisel Algoritmalar

TSDP'nin yapısından ve çözüm zorluğundan dolayı literatürde probleme özgü olarak geliştirilmiş birçok sezgisel algoritma mevcuttur.

Ho ve diğ. (1979), TSDP literatüründe en çok kullanılan 5 makineli ve 7 makineli karşılaştırma problemleri için gradyan arama tabanlı bir sezgisel algoritma önermişler ve her bir tampon stok alanındaki bir birim tampon stok artışının üretim hattı etkinliği üzerindeki etkisini incelemişlerdir.

Hillier (2013), literatürde ilk defa bir montaj hattında iş yükü atama problemi ve tampon stok dağılımı problemini eş zamanlı olarak ele almıştır. Yazar, simülasyon ve sezgisel algoritmalar kullanarak üretimden elde edilen geliri maksimize etmeye çalışmıştır.

Li ve diğ. (2015), üretim oranı maksimizasyonu için AY ile birlikte yerel aramaya dayalı hızlı algoritma (*fast algorithm: FA*) olarak isimlendirdikleri sezgisel bir algoritma önermişlerdir. Karşılaştırma için indirgenmiş tavan algoritması (İTA; *degraded ceiling: DC*) ve klasik yerel arama kullanılmış ve özellikle büyük üretim hatları için önerilen FA'nın karşılaştırılan diğer algoritmalarla benzer sonuçları daha kısa sürede elde ettiği belirtilmiştir.

Matta ve diğ. (2012), Kriging (Matheron 1963) temelli bir optimizasyon algoritması ile hem hipotetik hem de gerçek seri üretim hatlarında tampon stok minimizasyonunu ele almışlardır. Yazarlar, önerilen algoritmanın deneysel sonuçlarını etkin bir şekilde ortaya koyarak, algoritmanın genellenmesinin basit olduğunu ve kullanım kolaylığı sağladığını vurgulamışlardır.

Zhang ve diğ. (2014), üretim oranı maksimizasyonu ve süreç içi stok minimizasyonu amaçlarına dayalı olarak TSDP'nin çok amaçlı bir formülasyonunu tanımlamışlar ve problemin çözümü için melez bir sezgisel algoritma önermişlerdir.

Shi ve Gershwin (2016), minimum üretim oranı kısıtı altında kâr maksimizasyonuna odaklanarak TSDP'yi çözmek için gradyan temelli bir algoritma kullanmışlardır. Uzun bir hattı optimize etmek yerine, hattı daha küçük hatlara bölmek suretiyle optimize etmişler ve bu şekilde hesaplama süresinin önemli ölçüde azaldığını

göstermişlerdir. Başka bir gradyan tabanlı algoritma makinelerin bozulmadığı bir genel ağ sisteminde Wang ve diğ. (2016) tarafından gerçekleştirilmiştir. Çalışmada üç farklı üretim hattı yerleşim yapısı kâr maksimizasyonu açısından karşılaştırılmıştır.

Genel olarak, TSDP'leri çözerken, ilk makinenin sınırsız parça/hammadde tedarikine sahip olduğu varsayılır. Literatürden farklı olarak, sınırlı parça tedarik varsayımı altındaki TSDP, Weiss ve diğ. (2018) tarafından kural tabanlı (*rule-based*) bir yerel arama algoritması önerilerek çözülmüştür. Tampon stok kapasitelerinin alt limitlerini belirlemek için önerilen bu algoritma ile sipariş politikasının optimum tampon stok atanması üzerinde önemli bir etkiye sahip olduğu gösterilmiştir. İlk makinenin önünde sınırlı tampon stok alanı varsayımı altında TSDP'yi çözen bir başka yaklaşım Xi ve diğ. (2019) tarafından sunulmuştur. Xi ve diğ. (2019), dengeli olmayan, güvenilir, büyük seri-paralel üretim hatlarında TSDP'nin çözümü için hattı daha küçük hatlara bölerek analiz etmişlerdir. Amaç belirli bir üretim oranı kısıtı altında süreç içi envanteri minimize etmektir. Shi ve Gershwin (2016)'in çalışmasında olduğu gibi büyük bir hattın küçük hatlara bölünerek optimize edilmesinin daha etkin bir yöntem olduğu gösterilmiştir.

Romero-Silva ve Shaaban (2019), farklı uzunluklarda dengeli olmayan ve güvenilir olmayan montaj hatları için simülasyon kullanarak farklı tampon stok konfigürasyonlarının etkisini analiz etmişlerdir. Yazarlar tarafından, ortalama işlem süreleri ve tampon stok dağılımı dikkate alındığında kâse fenomenine benzer yapıdaki tampon stok dağılımının daha yüksek üretim oranı sağladığı sonucuna varılmıştır.

Liberopoulos (2020), güvenilir üretim hatları için kâr maksimizasyonu problemini üç farklı işletme kontrol politikası altında ele almıştır: kurulum tampon stoku, kademeli (*echelon*) tampon stok ve CONWIP. Yazar, kademeli tampon stok politikasının belli varsayımlar altında diğer iki politikadan daha iyi tampon stok konfigürasyonları ürettiğini göstermiştir.

3.2.6 Meta-Sezgisel Yöntemler

Meta-sezgisel yöntemler TSDP'nin bir kombinatoriyal optimizasyon problemi olması ve NP-Zor yapısı nedeni ile bu problemin çözümünde en sık kullanılan yöntemlerdir.

Nahas ve diğ. (2009), güvenilir olmayan seri-paralel üretim hatlarında makine seçimi ve TSDP'yi, KKO ve TB yöntemlerini kullanarak üretim oranı maksimizasyonu açısından ele almışlar ve KKO'nun TB'den daha iyi sonuçlar ürettiğini belirtmişlerdir. Yine Nahas ve diğ. (2014) tarafından yapılan başka bir çalışmada güvenilir olmayan bir montaj/de-montaj üretim hattının optimum tasarımı için TSDP ve makine seçim problemi aynı anda ele alınarak problemin çözümü için bir GA önerilmiştir.

Demir ve diğ. (2011), güvenilir olmayan seri üretim hatlarında üretim oranı maksimizasyonu ve toplam tampon stok miktarı minimizasyonu problemlerini ayrı ayrı ele almışlardır. Değerlendirme yöntemi olarak AY'nin kullanıldığı bu çalışmada, optimizasyon yöntemi olarak TA önerilmiştir. Ayrıca, toplam tampon stok miktarı minimizasyonu için ikili arama yöntemi TA ile birlikte kullanılmıştır. Demir ve diğ. (2012), güvenilir olmayan seri üretim hatlarında üretim oranı maksimizasyonu amacı ile TSDP'yi çözmek için adaptif bir TA algoritması önermişlerdir. Yazarlar, adaptif TA algoritmasının performansının özellikle 20 ve 40 makineli hatlara sahip büyük boyutlu problemlerde daha belirgin hale geldiğini belirtmişlerdir.

Amiri ve Mohtashami (2012), TSDP'yi çok amaçlı çözmek için melez bir GA önermişlerdir. Yazarlar, üretim oranı maksimizasyonu ve toplam tampon stok minimizasyonu problemlerini güvenilir olmayan seri-paralel üretim hatlarında birlikte ele almış ve değerlendirme yöntemi olarak simülasyon kullanmışlardır.

Papadopoulos ve diğ. (2013), dengeli üretim hatlarında güvenilir makinelerle BAP1 ve BAP2'yi çözmek için GA, TA, TB, miyopik algoritma (MA) ve tam sayım yöntemlerinin performanslarını karşılaştırmışlardır. Yazarlar, 3-80 aralığında makine sayısı ve 1-120 aralığında toplam tampon stok miktarıyla 105 farklı seri üretim hattındaki tampon stok dağılımlarını incelemiş ve bir karar destek sistemi tasarlamışlardır. Yazarlar aynı yılda yaptıkları bir diğer çalışmada (Tsadiras ve diğ.

2013) yapay sinir ağı tabanlı bir karar destek sistemi önermişlerdir. Bu çalışmada BAP1 ve BAP2 problemleri dengeli ve güvenilir hatlar için çözülmüş ve önerilen çözüm yönteminin doğruluğunun %99'a yakın olduğu belirtilmiştir.

Narasimhamu ve diğ. (2014) dengeli olmayan genel ağ sisteminde TSDP'yi CONWIP kontrol sistemi ile birlikte ele almış ve problemi PSO yöntemi kullanarak çözmüşlerdir. Yazarlar çalışmalarında, değerlendirme yöntemi olarak GY kullanmışlar ve üretim oranı maksimizasyonu yönünden 2 ve 3 makineli hatları incelemişlerdir.

Mohtashami (2014), güvenilir olmayan bir montaj hattında, TSDP ve makine atama problemleri için karışık tamsayılı doğrusal olmayan programlama modeli önermiştir. Önerilen matematiksel model, melez bir GA (MGA) kullanılarak çözülmüştür. Yazar, MGA'nın klasik GA'ya ve baskın olmayan sıralamalı genetik algoritma-II'ye (BOSGA-II) göre daha etkin olduğunu göstermiştir.

Köse ve Kılınçcı (2015), büyük üretim hatlarında TSDP'yi çözmek için TB ve GA'yı melez bir şekilde kullanmışlardır. Simülasyon ile birlikte kullanılan ve GAA (*Genetic Annealing Algorithm*) olarak isimlendirdikleri melez algoritmanın hem çözüm kalitesi (üretim oranı maksimizasyonu) ve hem de hesaplama etkinliği yönünden klasik GA'dan ve TB'den daha iyi sonuçlar elde ettiğini göstermişlerdir. Aynı yıl Köse ve diğ. (2015) tarafından yapılan bir çalışmada TB, TA ve GA bir ikili arama algoritması ile birlikte kullanılarak ısı eşanjörü üreten bir firmada toplam tampon stok miktarı minimize edilmeye çalışılmıştır. Önerilen ikili-TA algoritmasının performansının diğer iki algoritmadan daha iyi olduğu gözlenmiştir.

Costa ve diğ. (2015), güvenilir seri üretim hatlarında toplam tampon stok miktarını minimize etmek için paralel bir TA (PTA) algoritması geliştirmiş ve 5-10-15 makineli dengeli-dengeli olmayan hatları simülasyon yardımı ile analiz etmişlerdir.

Tiacci (2015), karma model montaj hattı dengeleme problemini ve TSDP'yi GA tabanlı bir çözüm yaklaşımı kullanarak eş zamanlı olarak çözmüştür. Benzer şekilde Zandieh ve diğ. (2017), bakım planlaması açısından dengeli olmayan, güvenilir olmayan üretim hatlarında TSDP'yi incelemiş ve optimizasyon için PSO ve GA yöntemlerini kullanarak karşılaştırmışlardır. Yazarlar, çözüm süresi açısından PSO

algoritmasının GA'dan daha iyi olduğunu göstermişlerdir. TSDP'nin çözümünde PSO ve GA'yı melez bir şekilde kullanan başka bir çalışma Xuemei ve diğ. (2017) tarafından sunulmuştur. Bu çalışmada TSDP ve seri üretim hattı dengeleme problemini aynı anda çözmek için GA-PSO tabanlı melez bir algoritma önerilmiştir ve önerilen algoritmanın etkinliği deneysel çalışmalarla gösterilmiştir.

Dolgui ve diğ. (2017), güvenilir olmayan seri-paralel bir üretim hattında hem üretim oranı maksimizasyonu problemini hem de maliyet minimizasyonu problemini çözmüşlerdir. Yazarlar optimizasyon yöntemi olarak basit evrimsel çok amaçlı optimizasyon algoritması ve değerlendirme yöntemi olarak simülasyon kullanmışlardır.

Su ve diğ. (2017), TSDP'yi yeniden üretim (*remanufacturing*) sistemleri için Zhang ve diğ. (2014) ile aynı hedefleri (üretim oranı maksimizasyonu ve süreç içi stok minimizasyonu) kullanarak çözmüşlerdir. Önerilen çözüm yönteminde, TA ve BOSGA-II yaklaşımları birleştirilmiştir. Başka bir BOSGA-II yaklaşımı Köse ve Kılınçcı (2020) tarafından kullanılmıştır. Yazarlar, TSDP'yi hem güvenilir hem de güvenilir olmayan seri üretim hatlarında çok amaçlı (üretim oranı maksimizasyonu ve toplam tampon stok miktarı minimizasyonu) olarak çözmek için BOSGA-II ve SA algoritmalarını melezleştirmişlerdir. Benzer şekilde Motlagh ve diğ. (2019), TSDP'yi çok amaçlı çözmek için BOSGA-II ve baskın olmayan sıralamalı genetik algoritma (BOSGA; *Non-dominated Ranked Genetic Algorithm: NRGGA*) kullanmışlardır. Problem üretim oranı maksimizasyonu, toplam tampon stok minimizasyonu ve toplam maliyet minimizasyonu amaçları için çözülmüş ve önerilen algoritmanın avantajları deneysel çalışmalar ile gösterilmiştir.

Nahas ve Nourelfath (2018), TSDP'yi her bir makinede gerçekleştirilen önleyici bakım ve makine seçimi problemi ile birlikte ele almışlardır. Yazarlar, toplam maliyet minimizasyonu problemini beklenen üretim oranı kısıtı altında güvenilir olmayan seri üretim hatları için çözmüş ve çözüm için GA, KKO ve doğrusal olmayan eşik değeri içeren bir meta-sezgisel algoritma önermişlerdir.

Zhou ve diğ. (2018), literatürdeki çalışmalardan farklı olarak, üretim oranı yerine seri üretim hattının performansını kullanılabilirlik (*availability*) üzerinden ölçmüşlerdir. 5 makineden 30 makineye ve 20 tampon stoktan 360 tampon stoka kadar

olan güvenilir olmayan hatlarda optimizasyon algoritması olarak GA, İTA, PSO, dağıtım tahmini algoritması (DTA; *estimation of distribution algorithm*) ve melez PSO-DTA kullanmışlar ve bu algoritmaları hem amaç fonksiyonu hem de hesaplama zamanı (CPU) açısından karşılaştırmışlardır.

Lin ve diğ. (2020), güvenilir olmayan seri üretim hatlarında Matta ve diğ. (2012) tarafından önerilen vekil tabanlı optimizasyon (VTO; *surrogate-based optimization*) algoritmasını kullanmışlardır. Yazarlar, üretim oranı kısıtı altında toplam tampon stok miktarı minimizasyonu problemini ele almışlar ve VTO ile GA'yı karşılaştırmışlardır. Çalışmada, VTO'nun TSDP'ye uygulamak için gayet esnek olduğu ve üretim hattındaki başka problemlere de uygulanabileceği vurgulanmıştır.

3.3 TSDP'de Kullanılan Değerlendirme Algoritmaları

TSDP'nin çözümünde kullanılan değerlendirme algoritmaları analitik yöntemler ve simülasyon olmak üzere ikiye ayrılmaktadır. Analitik yöntemler, üstel veya geometrik olarak dağılmış güvenilirlik parametrelerine sahip makineler gibi belirli varsayımların geçerli olduğu durumlarda hızlı bir şekilde hattın performans ölçütlerini elde etmede kullanılırken, analitik modelleri kullanmak için gerekli varsayımların yapılamadığı durumlar ve karmaşık sistemlerin modellenmesinde simülasyon tercih edilmektedir. Belirli varsayımlar altında yaklaşık analitik yöntemler kullanılarak hızlı bir şekilde elde edilebilen hat performans ölçütlerinin simülasyon kullanılarak elde edilenlere çok yakın olduğu bilinmektedir (Gershwin 1987).

3.3.1 Analitik Yöntemler

Üretim hatlarının kesin analitik çözümleri ancak çok küçük hatlar için bulunabilmektedir (4 makineye kadar olan hatlar). Daha büyük hatlar için yaklaşık analitik yöntemlerin kullanılması gerekmektedir. TSDP literatüründe temel olarak kullanılan üç yaklaşık analitik yöntem vardır: ayrıştırma yöntemi, toplama yöntemi ve genişleme yöntemi.

Üretim sistemlerinin analizinde 1980’li yılların sonundan itibaren çok sayıda yaklaşık analitik yöntem geliştirilmiştir (Gershwin 1987, De Koster 1987, Kerbache ve Smith 1987, Dallery ve diğ. 1988, 1989, Dallery ve Frein 1993, Glassey ve Hong 1993, Burman 1995, Bihan ve Dallery 2000). Bu analitik algoritmaların büyük bir kısmı ayrıştırma yöntemine dayanmakta ve üretim hattının performansını hızlı ve doğru bir şekilde ölçmektedirler.

Kesin analitik çözümler ancak çok küçük hatlara (çoğunlukla 2-makinelik) uygulanabilmektedir (Li ve diğ. 2006). Daha büyük hatlar için yaklaşık analitik yöntemler kullanmak gerekmektedir. Çoğu yaklaşık yöntem, hattın analiz edilmesi ve performans parametrelerinin ölçümü için denklem setlerinin oluşturulması açısından BY ve AY’ye dayanır (Dallery ve Gershwin 1992). 1988 yılından sonraki diğer analitik yaklaşık yöntemler (Glassey ve Hong 1993, Bihan ve Dallery 2000) DDX algoritmasının değiştirilmesi/genişletilmesiyle elde edilmiştir.

Fuxman (1998), mümkün olan en yüksek üretim oranını korumak için senkron olmayan, güvenilir bir karma model montaj hattında gerekli minimum sayıda tampon stokun hesaplanmasına ve bunların optimum dağıtılmasına olanak sağlayan analitik bir model sunmuştur.

Kim ve Gershwin (2008) tarafından büyük hatlar için İBB tipi bozulmayı ve kaliteyi dikkate alan analitik bir yöntem geliştirilmiştir. Kalite işlemini yürüten muayene istasyonlarının konumları ve muayene istasyonlarının izlediği makine gruplarının farklılık gösterdiği 3 farklı durum için geliştirilen yöntem simülasyon ile karşılaştırılmıştır. Muayenelerin operasyonlardan ayrılmasının etkilerini ölçen bu yöntemin güvenilir olmayan seri üretim hatlarını değerlendirmede kullanılabileceği belirtilmiştir.

Xia ve diğ. (2012), Dallery ve Bihan (1999) ve Bihan ve Dallery (2000) tarafından geliştirilen değerlendirme algoritmalarını dengeli olmayan hatlar için genişletmişler ve yeni bir değerlendirme algoritması önermişlerdir.

Li ve diğ. (2016) sınırlı tampon stoka sahip güvenilir ve yaklaşık dengeli seri üretim hatlarının performansını (üretim oranı) değerlendirmek için AY temelli iki formülasyon geliştirmişlerdir. Formülasyonlar eşit tampon stok dağılımlı veya

optimuma yakın tampon stok dağılımlı makinelerin üretim sürelerinin birbirlerine çok yakın olduğu hatlara uygulanmıştır. Önerilen formülasyonlarla güvenilir ve yaklaşık dengeli seri üretim hatlarının üretim oranı değerlerinin hızlı ve doğru bir şekilde hesaplanabileceği yazarlar tarafından gösterilmiştir.

Matta ve Simone (2016), Gershwin ve Berman (1981) ve Gershwin (1994) tarafından sunulan yöntemi genişleterek, İBB ve ZBB tipi bozulmalara maruz kalan, dengeli, güvenilir olmayan seri üretim hatları için bir değerlendirme yöntemi geliştirmişlerdir. Geliştirilen değerlendirme yöntemi mevcut değerlendirme yöntemleriyle doğruluk yönünden karşılaştırılmıştır. Ayrıca hem İBB hem ZBB bozulma tiplerine maruz kalan makinelerin performanslarını değerlendirirken, İBB tipi bozulmanın neden olduğu hata yazarlar tarafından ortaya konmuştur.

Diamantidis ve diğ. (2020) güvenilir olmayan, dengeli olmayan seri-paralel üretim hatlarının performansını değerlendirmek için AY temelli bir algoritma önermişlerdir. Önerilen algoritmanın doğruluğunu 50 makineye kadar olan hatlarda ölçmek için simülasyon modelleriyle karşılaştırmışlardır. Yazarlar, önerilen algoritmanın büyük güvenilir olmayan seri-paralel üretim hatlarına uygulanabilir olduğunu ve algoritma süresinin büyük hatlarda dahi makul olduğunu vurgulamışlardır.

3.3.2 Simülasyon

Simülasyon kompleks sistemleri modellemek için güçlü bir araçtır. Fakat değerlendirme süresi analitik yöntemlere göre daha fazladır (Bulgak 2006) ve kimi zaman gerçek hayat problemleri için maliyetli olabilmektedir (Köse ve diğ. 2015). Mohtashami (2014), Zhang ve diğ. (2014), Costa ve diğ. (2015), Köse ve Kılınçcı (2015), Köse ve diğ. (2015), Pedrielli ve diğ. (2018), Motlagh ve diğ. (2019), Zhang ve Matta (2020) ve Alfieri ve diğ. (2020)'nin çalışmaları literatürde TSDP'nin çözümünde simülasyon kullanan çalışmalardır.

3.4 Özet

Bu bölümde TSDP alanında son on yılda yapılmış olan çalışmalar, problemin çözümünde kullanılan tekniklere odaklanılarak incelenmiştir. Ayrıca, Demir ve diğ. (2014) tarafından önerilen sınıflandırma metodolojisi kullanılarak çalışmaların özellikleri detaylı bir şekilde sunulmuş ve TSDP literatürü için genel bir çerçeve ortaya konmuştur.

Mevcut TSDP literatürü incelendiğinde;

- En çok kullanılan değerlendirme yönteminin simülasyon (41 çalışmanın 21'i),
- En çok kullanılan optimizasyon yönteminin yinelemeli ve popülasyon tabanlı bir meta-sezgisel olan GA ve varyasyonları (41 çalışmanın 15'i),
- En fazla incelenen hat topolojisinin seri üretim hattı (41 çalışmanın 29'u),
- En fazla ele alınan problemin üretim oranı maksimizasyonu problemi (41 çalışmanın 26'sı),
- Gerçek hayat problemlerine yakınlığı sebebiyle en çok GOH'ların (41 çalışmanın 28'i),
- En çok makinelerin çoğunluğunun işlem süresinin birbirinden farklı olduğu hatların,
- En çok dikkate alınan makine bloke durumunun SSB (41 çalışmanın 36'sı),
- En çok doymuş model varsayımı altında üretim sistemlerinin

incelenmiş olduğu görülmektedir (bkz. Tablo 3.1).

Her ne kadar literatürde TSDP en fazla bozulmaya maruz kalan seri üretim hatları için çözülmüş olsa da bu özelliklere sahip hatlarda AY ve komşuluk temelli meta-sezgisel yöntemler kullanılarak yapılan çalışmalar sınırlıdır. Ayrıca yeni geliştirilmiş olan SGA ve BP-BÇ algoritması gibi meta-sezgisel algoritmaların performanslarının bu özellikteki hatlarda TSDP'nin çözümü için test edilmesinin de önemli olduğu düşünülmektedir. Bunun yanı sıra güvenilir olmayan hatlar gerçek hayat problemlerini daha iyi yansıtan problemlerdir ve bu problemlerin çözümünde etkin algoritmalar kullanmak optimum hat tasarımı açısından oldukça önemlidir. Son yıllarda yapılan çalışmaların önemli bir kısmı TSDP'yi çok amaçlı olarak ele alsa da

problemin özellikleri ve zorluğu nedeni ile TSDP'yi tek bir amaç için etkin algoritmalarla çözmek hem akademi hem de endüstri için halen daha önemli bir araştırma konusu olarak karşımıza çıkmaktadır.

4. METODOLOJİ

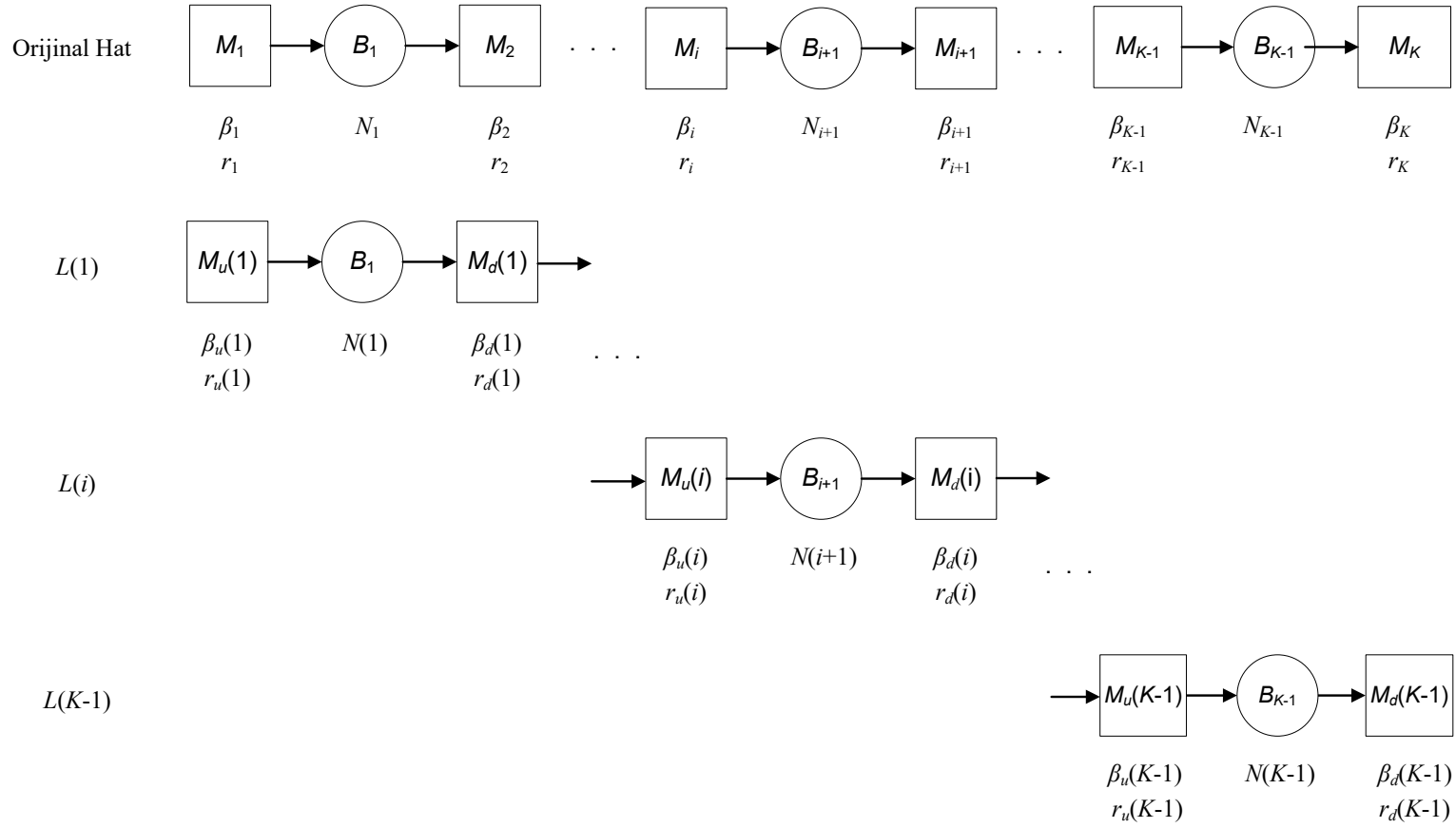
4.1 Giriş

Bu bölümde bu tez çalışmasında kullanılan yöntemler ile ilgili temel bilgilere yer verilmiştir. İlk olarak, Bölüm 4.2’de TSDP’nin çözümünde değerlendirme yöntemi olarak kullanılan ayrıştırma algoritması detaylı olarak anlatılmıştır. Ardından çalışmada kullanılan optimizasyon algoritmaları tanıtılmıştır. Bölüm 4.3 ve Bölüm 4.4’te sırası ile önce değişken komşuluk arama daha sonra büyük komşuluk arama algoritmalarının ana hatları verilmiştir. Bölüm 4.5’te savaşı genetik algoritma ve Bölüm 4.6’da ise büyük patlama-büyük çöküş algoritması anlatılmıştır. Ayrıca her bir algoritmanın literatürdeki mevcut uygulamaları ilgili bölümlerde anlatılmıştır. Son olarak Bölüm 4.7’de bu bölümün içeriği özetlenmiştir.

4.2 Ayrıştırma Yöntemi

Ayrıştırma yöntemi Gershwin tarafından 1987 yılında önerilmiş, seri üretim hatlarının performans ölçütlerini yaklaşık olarak hesaplamaya yarayan analitik bir yöntemdir. AY bir seri üretim hattını iki makineli alt hatlara bölmek suretiyle hattın tümü için performans değerlerini hesaplayan bir değerlendirme yöntemidir. Her AY temelde üç adımdan oluşur: (1) alt sistemlerin karakterize edilmesi, (2) her alt sistemin bilinmeyen parametrelerinin denklem seti halinde belirlenmesi ve (3) bu denklem setini çözecek bir algoritmanın geliştirilmesi (Papadopoulos ve Heavey 1996).

Gershwin (1987) tarafından önerilen AY, K makineden oluşan bir üretim hattının $K-1$ adet 2 makineli alt hatlara bölünerek analiz edilmesine dayanır. Şekil 4.1’de görüldüğü gibi iki makineden oluşan bu hatlar $L(i)$, $i=1, \dots, K-1$, olarak isimlendirilmektedir. Her ikili hat orijinal hattaki konumları göz önünde bulundurularak $M_u(i)$ ve $M_d(i)$ olarak isimlendirilen iki makine ve bu makinelerin arasındaki N_i kapasiteli $K-1$ adet tampon stok alanından oluşur. $M_u(i)$ ve $M_d(i)$ makinelerinin bozulma ve tamir oranları da $\beta_u(i)$, $r_u(i)$ ve $\beta_d(i)$, $r_d(i)$ ile ifade edilirler.



Şekil 4.1: Ayrıştırma Yöntemi

Bu üretim hattının ortalama üretim oranını belirlemek için, tüm hat kararlı durum davranışının belirlendiği bir Markov süreci olarak modellenmiştir. İki makineli hatların performans özellikleri tam olarak elde edilebildiğinden, ayrıştırma yöntemi, ayrıştırılmış iki makineli hatları birbirine bağlayan bir dizi denklemin türetilmesini gerektirir. Bu denklem dizisi de $L(i)$ hattının bilinmeyen parametreleri olan $\beta_u(i)$, $r_u(i)$, $\beta_d(i)$ ve $r_d(i)$ değerlerinin belirlenmesi için çözülür.

Dallery ve diğ. (1988), Gershwin (1987) tarafından önerilen AY yöntemi sonucunda ikili makinelere ait bilinmeyen parametrelerin belirlendiği doğrusal olmayan denklem sistemini çözmek üzere bir algoritma önermişlerdir. Önerilen bu algoritma yazarların soyisimlerinin baş harflerine atıfla DDX algoritması olarak anılmaktadır. Algoritma, üretim sürelerinin deterministik ve 1 birim olduğu Markovyen özellikteki güvenilir olmayan seri üretim hatlarının performans ölçütlerini hızlı bir şekilde hesaplamada oldukça etkindir. Yakınsama doğruluğu ve hızı nedeniyle bu tez çalışmasında değerlendirme algoritması olarak DDX algoritması kullanılmıştır.

DDX algoritması Algoritma 1’de verilmiştir. Her 2 makineli hattın bilinmeyen parametrelerini belirlemek için Gershwin (1987) tarafından doğrusal olmayan bir denklem seti geliştirilmiştir. Bu denklem setinin ve DDX algoritmasının ayrıntıları aşağıda verilmiştir.

$$e_i = \frac{r_i}{r_i + \beta_i}; i = 1, 2, \dots, K \quad (4.1)$$

$$Th(1) = Th(2) = \dots = Th(K - 1) \quad (4.2)$$

$$Th(i) = e_i(1 - p_b(i) - p_s(i - 1)) \quad (4.3)$$

Denklem 4.1 ile makine i 'nin izolasyondaki etkinliği hesaplanmaktadır. Denklem 4.2 parça akışının korunmasıyla ilişkilidir. Denklem 4.3 ile her bir iki-makineli hattın üretim oranı değeri hesaplanır. Denklem 4.3'teki $p_s(i)$, $M_u(i)$ makinenin açlık olasılığı ve $p_b(i)$ ise, $M_d(i)$ makinesinin bloklanma olasılığıdır. $p_s(i)$, $p_b(i)$ ve $Th(i)$ miktarları bilinmeyen dört parametrenin ($\beta_u(i)$, $r_u(i)$, $\beta_d(i)$, $r_d(i)$) birer fonksiyonudurlar. Bu değerler Gershwin ve Schick (1983) tarafından iki makineli hatlar için türetilmiş olan kararlı durum olasılıkları (*steady-state probabilities*) ve performans ölçütleri ile hesaplanabilmektedir.

Bir istasyonun aynı anda bloke ve aç kalma olasılığının sıfır olduğu varsayımına dayanan akış hızı bozta kalma süresinin hesaplanması Denklem 4.4'te verilmiştir. Denklem 4.5 ve Denklem 4.6 ise komşu iki hattın tamir oranları ile orjinal hattın tamir oranı arasındaki ilişkiyi göstermektedir.

$$\frac{\beta_d(i-1)}{r_d(i-1)} + \frac{\beta_u(i)}{r_u(i)} = \frac{1}{Th(i-1)} + \frac{1}{e_i} - 2; i = 2, \dots, K-1 \quad (4.4)$$

$$X = \frac{r_u(i)p_s(i-1)}{\beta_u(i)Th(i-1)} \quad (4.5)$$

$$Y = \frac{r_d(i-1)p_b(i)}{\beta_d(i-1)Th(i)} \quad (4.6)$$

$$\left\{ \begin{array}{l} I_u(i) = \frac{\beta_u(i)}{r_u(i)} \\ I_d(i) = \frac{\beta_d(i)}{r_d(i)} \end{array} \right. ; i = 1, 2, \dots, K-1 \quad (4.7)$$

Gershwin (1987) tarafından oluşturulan denklem seti, Denklem 4.7'den elde edilen değerler ve (4.1)-(4.7) denklemleri dikkate alınarak Dallery ve diğ. (1988) tarafından yeniden ifade edilmiştir.

$$I_u(i) = \frac{1}{Th(i-1)} + \frac{1}{e_i} - I_d(i-1) - 2; i = 2, \dots, K-1 \quad (4.8)$$

$$r_u(i) = Xr_u(i-1) + (1-X)r_i; i = 2, \dots, K-1 \quad (4.9)$$

$$X = \frac{p_s(i-1)}{I_u(i)Th(i-1)}$$

$$I_d(i) = \frac{1}{Th(i+1)} + \frac{1}{e_{i+1}} - I_u(i+1) - 2; i = 1, \dots, K-2 \quad (4.10)$$

$$r_d(i) = Yr_d(i+1) + (1-Y)r_{i+1}; i = 1, \dots, K-2 \quad (4.11)$$

$$Y = \frac{p_b(i+1)}{I_d(i)Th(i+1)}$$

Alt hatlara ait parametreler Denklem 4.8-4.11 ile elde edilebilmektedir. Dallery ve diğ. (1988), yukarıda verilen doğrusal olmayan denklem sistemini çözmek için aşağıda uygulama adımları verilen DDX algoritmasını geliştirmişlerdir (Algoritma 1). Yazarların belirttiği gibi, DDX algoritmasının hesaplama karmaşıklığı, K makine sayısını göstermek üzere, $O(K^2)$ 'dir.

Algoritma 1: DDX Algoritması (Dallery ve diğ 1988)

girdi : Makine sayısı K , tampon stok konfigürasyonu N , bozulma oranı β ve tamir oranı r

çıkıtı : Üretim oranı $Th(N)$

Adım 1: Başlatma

1 $\beta_u(1) \leftarrow \beta_1$

2 $r_u(1) \leftarrow r_1$

3 **for** ($i \leftarrow 1$ **to** $(K-1)$) **do**

4 $\beta_d(i) \leftarrow \beta_{i+1}$

5 **for** ($i \leftarrow 1$ **to** $(K-1)$) **do**

6 $r_d(i) \leftarrow r_{i+1}$

Adım 2:

7 **for** ($i \leftarrow 2$ **to** $(K-1)$) **do**

8 Denklem (4.8), Denklem (4.9) ve Denklem (4.7) kullanılarak $I_u(i)$, $r_u(i)$ ve $\beta_u(i)$ hesapla

Adım 3:

9 **for** ($i \leftarrow (K-2)$ **to** 2) **do**

10 Denklem (4.10), Denklem (4.11) ve Denklem (4.7) kullanılarak $I_d(i)$, $r_d(i)$ ve $\beta_d(i)$ hesapla

11 **Adım 2'**ye git (yakınsama oranı yakalanana kadar)

4.3 Değişken Komşuluk Arama Algoritması

Değişken komşuluk arama algoritması tek çözüme dayalı, statik amaç fonksiyonu kullanan, çeşitli komşuluk yapılarını sistematik bir şekilde değiştirerek arama yapan kombinatoriyal ve global optimizasyon problemlerini çözmeyi amaçlayan basit ve etkili bir meta-sezgiseldir (Mladenovic 1995, Mladenovic ve Hansen 1997). DKA, komşuluk yapılarını sırayla kullanarak yerel optimumdan kaçmaya çalışır (Hansen ve Mladenovic 2001, Hemmelmayr ve diğ. 2012). Bir komşuluk tabanlı arama algoritmasının tasarımında en kritik konulardan biri komşuluk yapılarının seçimidir (Ahuja ve diğ. 2002). DKA algoritmasının önemli bir özelliği komşuluk yapıları arasındaki değişim stratejisidir. Bir komşuluk yapısı yerel optimum noktaya takıldığında, arama süreci boyunca komşuluk yapıları değiştirilerek durdurma koşulu sağlanana kadar devam ettirilir (Yu ve Maghfiroh 2014, Pereira ve Vila 2015). Mevcut çözüme komşuluk yapısı uygulanarak çeşitlendirme sağlanır ve bulunan komşu çözüm yerel arama ile yoğunlaştırılarak optimum çözüm aranır (Karakostas ve diğ. 2020).

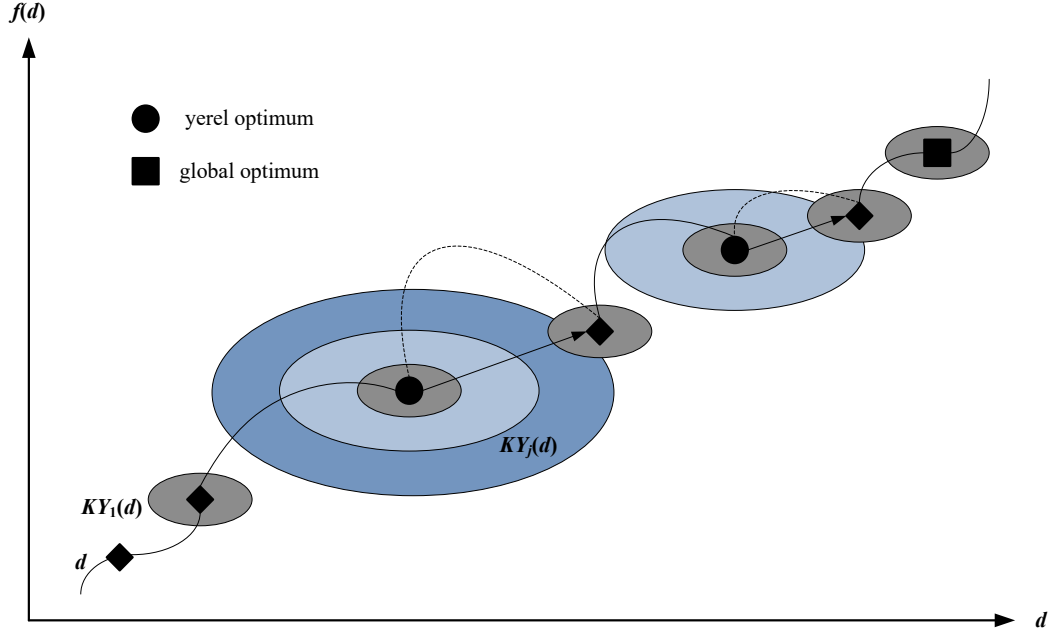
DKA ile ilgili ayrıntılı bilgi edinmek isteyen okuyucular Hansen ve diğ. (2008) tarafından yapılan literatür araştırması çalışmasına bakabilirler.

Komşuluk yapılarının uygulandığı aşama karıştırma (*shaking/perturbation*), yerel arama algoritmasının uygulandığı aşama ise yerel arama (*local search*) olarak tanımlanmaktadır. Karıştırma aşamasında komşuluk yapıları mevcut çözüm üzerinde çeşitlendirme sağlayarak yerel aramanın farklı alanlarda yoğunlaşmasına yardımcı olur. Yerel arama kombinatoriyal optimizasyon problemlerinin çözümünde sıklıkla kullanılan sezgisel bir yöntemdir ve komşuluk yapısı ile çalışan algoritmaların temelini oluşturur (Irnich ve diğ. 2006). Kombinatoriyal, doğrusal olmayan optimizasyon problemlerinin optimum ya da optimuma yakın çözümünü bulmada esnek bir yapıya sahip olan DKA'nın karıştırma ve yerel arama aşamasının temelini aşağıdaki bulgular oluşturmaktadır (Hansen ve diğ. 2017):

- Bir komşuluk yapısı kullanılarak elde edilen yerel optimum, başka bir komşuluk yapısı için yerel optimum olmak zorunda değildir.
- Global optimum, tüm komşuluk yapıları için yerel optimumdur (Smiti ve diğ. 2020).
- Deneysel çalışmalarda elde edilen bulgular, birçok problem için, yerel optimum çözümlerin tümü veya büyük bir çoğunluğunun nispeten birbirine yakın çözümler olduğunu göstermektedir (Kirkpatrick ve Toulouse 1985).

DKA'da bütün komşuluk yapılarına göre yerel optimum çözümleri bulmak için çözüm uzayında çeşitliliği artıracak şekilde hareket edilir (Yu ve Maghfiroh 2014). Yerel optimum çözümlerin kalitesini artırmak için birden fazla komşuluk yapısı kullanılır. Ayrıca, mevcut en iyi çözümün etrafında yoğunlaşarak arama yapılır.

Şekil 4.2'de bir başlangıç çözümünden komşuluk yapılarını kullanarak global optimuma ulaşırken çözüm uzayının nasıl tarandığı resmedilmiştir. Bu gösterim için Hansen ve diğ. (2008)'nin çalışmasından faydalanılmıştır.



Şekil 4.2: DKA'da çözüm uzayının taranması

Kombinatoryal optimizasyon problemlerini çözmek amacıyla DKA'nın birçok varyasyonu geliştirilmiştir (Hansen ve diğ. 2008):

Değişken Komşuluk İnişi (Variable Neighborhood Descent): Komşuluk yapılarının deterministik bir şekilde değiştirilmesi ile elde edilir. Genellikle 1 veya 2 komşuluk yapısı kullanılır.

Azaltılmış Değişken Komşuluk Arama (Reduced Variable Neighborhood Search): Komşuluk yapıları sıralı bir şekilde uygulanır ve yerel arama yapılmazsa bu varyasyon elde edilir (Hansen ve diğ. 2017). Seçilen çözümler mevcut en iyi çözümden daha iyiyse mevcut çözüm güncellenir.

Genel Değişken Komşuluk Arama (General Variable Neighborhood Search): DKA'daki yerel arama adımı Değişken Komşuluk İnişi ile değiştirilerek elde edilir.

DKA'nın diğer varyasyonları ve uygulamaları hakkında daha detaylı bilgi için Hansen ve diğ. (2008)'nin çalışması incelenebilir.

DKA'nın temel adımları (Hansen ve Mladenovic 2001) Algoritma 2'de verilmiştir. Algoritmanın başlangıcında komşuluk yapıları kümesi oluşturulur.

DKA’da komşuluk yapıları rastgele oluşturulabileceği gibi problemin yapısına özel olarak ya da melez bir şekilde oluşturulabilir. Daha sonra rassal bir şekilde uygun bir başlangıç çözümü (B^{bas}) elde edilir. Bu mevcut çözüm (B^{bas}) üzerine komşuluk yapıları kümesinden seçilen bir komşuluk yapısı ile karıştırma aşaması uygulanarak yeni bir çözüm (B^{kar}) elde edilir. Daha sonra elde edilen bu yeni çözüme yerel arama algoritması uygulanır ve B^{yer} elde edilir. Karıştırma ve yerel arama aşamalarından sonra amaç fonksiyonunda iyileşme sağlanırsa mevcut çözüm yerel optimum çözüm olarak kabul edilir ve aynı komşuluk yapısı ile karıştırma ve yerel arama aşamaları tekrar uygulanır. Temel DKA’da amaç fonksiyonunda bir iyileşme sağlanması durumunda yeni çözüm mevcut en iyi çözüm olarak kabul edilmektedir. Çözümde iyileşme sağlanmaması durumunda komşuluk yapısı değiştirilir. Komşuluk yapıları önceden belirlenen sabit bir sıraya göre uygulanabildiği gibi, rastgele veya belli mekanizmalara göre de uygulanabilmektedir. Algoritma için başlangıçta belirlenen durdurma kriteri sağlandığında, algoritma sonlandırılır. Durdurma kriteri olarak belirli bir iterasyon sayısına veya belirli bir hesaplama zamanına ulaşılması gibi kriterler kullanılabilir. Bir DKA algoritmasında sadece bir sonlandırma kriteri olabileceği gibi problemin yapısına ve koşullara göre birden fazla kriter de olabilir.

Algoritma 2: Temel DKA Algoritması (Hansen ve Mladenovic 2001)

```

1  Komşuluk yapıları kümesini seç  $KY_j (j=1, 2, \dots, j_{maks})$ 
2  Bir başlangıç çözümü bul;  $B^{bas}$ 
3  yap
4       $j=1: j_{maks}$ 
5      tekrar et
6           $B^{bas}$  ( $B^{kar} \in KY_j(B^{bas})$ ) komşuluğundan  $j$ .  $KY$  ile bir  $B^{kar}$  üret    % Karıştırma %
7           $B^{kar}$ ’a yerel arama uygula ve yerel optimum  $B^{yer}$  elde et    % Yerel Arama %
8          if  $f(B^{yer}) > f(B^{bas})$  then
9               $B^{bas} \leftarrow B^{yer}$ 
10              $j \leftarrow 1$ 
11         else
12              $j \leftarrow j + 1$                                 % Komşuluk Değiştirme %
13     Durdurma kriteri sağlanana kadar
14 end

```

DKA algoritmasının uygulama alanı ve varyasyonları çok geniştir. Komşuluk yapılarının yerel arama ile çeşitli şekillerde uygulanmasına veya diğer meta-sezgisel algoritmalarla birlikte melez olarak kullanılmasına ilişkin bilimsel literatürden birçok örnek verilebilir.

DKA ilk olarak gezgin satıcı problemlerine (GSP) (Hansen ve Mladenovic 1999, Burke ve diğ. 2001) ve GSP'nin uzantısı olan araç rotalama problemlerine (ARP) uygulanmıştır. Burke ve diğ. (2001), DKA'nın ilk klasik versiyonunda rastgele uygulanan karıştırma aşamasını yönlendirilmiş şekilde uygulamışlardır. Yazarlar, yönlendirmeli DKA algoritmasının asimetric GSP üzerinde çok iyi sonuçlar ürettiğini kanıtlamışlardır. Rousseau ve diğ. (2002) yaptıkları çalışmada farklı komşuluk yapılarını kullanarak zaman pencereli ARP için etkin sonuçlar elde etmişlerdir. Benzer şekilde zaman pencereli ARP'yi ele alan Braysy (2003), DKA'yı bir dizi yerel arama ve meta-sezgisel yöntemle karşılaştırmış ve DKA'nın diğer yöntemlerden daha iyi sonuçlar elde ettiğini göstermiştir. Polacek ve diğ. (2004) tarafından zaman pencereli ARP çok depolu olarak incelenmiş ve DKA kullanılarak komşuluk temelli bir yöntem olan TA'ya göre daha iyi sonuçlar elde edilmiştir. Büyük ölçekli ARP'lerde DKA algoritmasının etkinliği Kytöjoki ve diğ. (2007) tarafından ortaya konmuştur. Çözüm kalitesi ve hesaplama süresi açısından değerlendirilen toplamda 52 büyük ölçekli gerçek örnekte yazarlar, kıyaslama yaptıkları çalışmalarda kullanılan yöntemlere göre daha iyi sonuçlar elde etmişlerdir. Benzer şekilde Fleszar ve diğ. (2009), on altı ARP karşılaştırma problemini araç kapasitesi kısıtıyla çözmüşlerdir. Sarasola ve diğ. (2016), ARP'yi stokastik talepler ve dinamik ihtiyaçlarla birlikte değerlendirmişlerdir. Çözüm yöntemi olarak deterministik DKA ve stokastik DKA kullanılan bu çalışmada, stokastik DKA'nın çok fazla rastgele bilgi almasından dolayı deterministik DKA'dan daha iyi sonuçlar verdiği gösterilmiştir. Kapasite kısıtlı ARP'yi ele alan bir diğer çalışma Amous ve diğ. (2017) tarafından sunulmuştur. Yazarlar, DKA ve DKA'nın farklı bir varyasyonunu önermiş ve literatürdeki karşılaştırma problemleri üzerinde uygulamışlardır. Ferreira ve diğ. (2018) çalışmalarında DKA ve TA algoritmalarını melez bir şekilde çoklu zaman pencereli ARP'ye uygulamışlardır. Uygulanan problem setleri üzerinde elde edilen sonuçlar melez algoritmanın literatürdeki en iyi sezgiseller ile rekabet edebilir seviyede olduğunu göstermiştir. ARP'nin farklı kısıtlarla ele alınan versiyonları üzerindeki DKA uygulamaları için Xu ve Cai (2018) ile Derbel ve diğ. (2019)'nin çalışmalarına bakılabilir.

ARP dışında DKA algoritmasının başarılı bir şekilde yaygın olarak kullanıldığı problemler şu şekilde sıralanabilir: Personel planlama problemi (Burke ve diğ. 2008, Burke ve diğ. 2010, Rahimian ve diğ. 2017), sipariş gruplama/sıralama problemi (Menendez ve diğ. 2017^{a,b}), *p*-meydan problemi (Hansen ve diğ. 2001, Garcia Lopez

ve diğ. 2002, Irawan ve diğ. 2020), p -merkez problemi (Mladenovic ve diğ. 2003^a, Fleszar ve Hindi 2008, Elshaikh ve diğ. 2015), Weber problemi (Brimberg ve diğ. 2000), veri madenciliği (Belacel ve diğ. 2004, Mucherino ve Liberti 2013), kutu paketlenme/sırt çantası problemi (Fleszar ve Hindi 2002, Hemmelmayr ve diğ. 2012, Taşgetiren ve diğ. 2015), sınav çizelgeleme problemi (Fonseca ve Santos 2014, Borchani ve diğ. 2017), çizelgeleme problemi (Kocatürk ve Özpeynirci 2014, Moslehi ve Khorasanian 2014, Pacheco ve diğ. 2018) ve tasarım problemi (Costa ve diğ. 2002, Lapierre ve diğ. 2004, Mladenovic ve diğ. 2003^b, Meric ve diğ. 2004, Loudni ve diğ. 2006). Brimberg ve diğ. (2003) ve Aloise ve diğ. (2006) ise petrol sektöründeki gerçek problemlere DKA yöntemini uygulamışlardır. Mevcut DKA literatürü incelendiğinde DKA algoritmasının daha önce herhangi bir üretim sistemi tasarımı problemine uygulanmadığı görülmüştür.

4.4 Büyük Komşuluk Arama Algoritması

Büyük komşuluk arama, kısıt programlamayla yerel aramanın etkinliğini birleştiren, kombinatoriyal optimizasyon problemlerini çözmeyi amaçlayan, Shaw (1997) tarafından önerilmiş bir sezgisel algoritmadır. BKA algoritması ilk olarak Shaw (1997) tarafından ARP'yi çözmek için kullanılmıştır. Temel mantığı mevcut bir çözüme bir çıkarma (*removal*) ve bir ekleme (*insertion*) operatörünü aşamalı olarak uygulamak suretiyle yinelemeli olarak mevcut çözümde iyileşme sağlamaktır. Mevcut çözümün bir parçası çıkarma operatörü ile çözümden çıkarılmakta ve sonrasında ekleme operatörüyle bu parça mevcut çözümü iyileştirecek şekilde tekrardan çözüme dahil edilmektedir (Azi ve diğ. 2014).

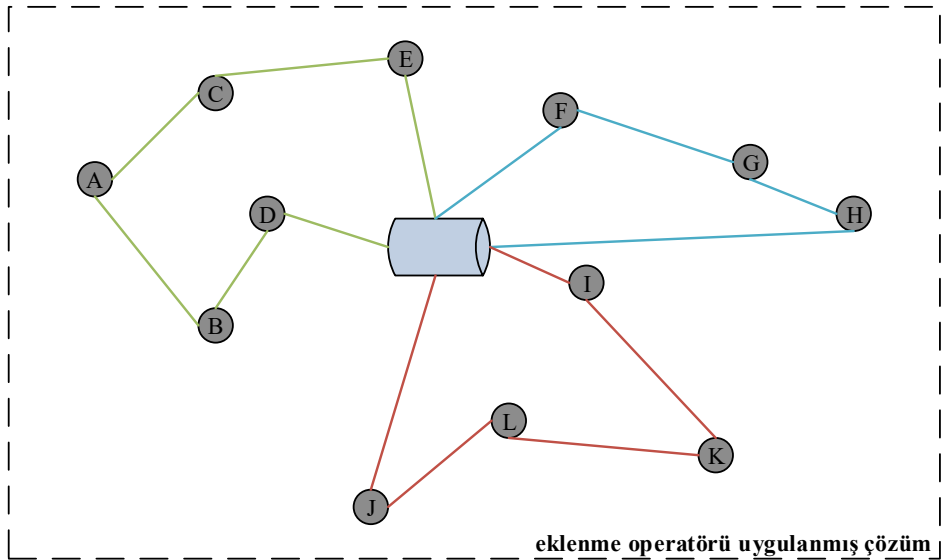
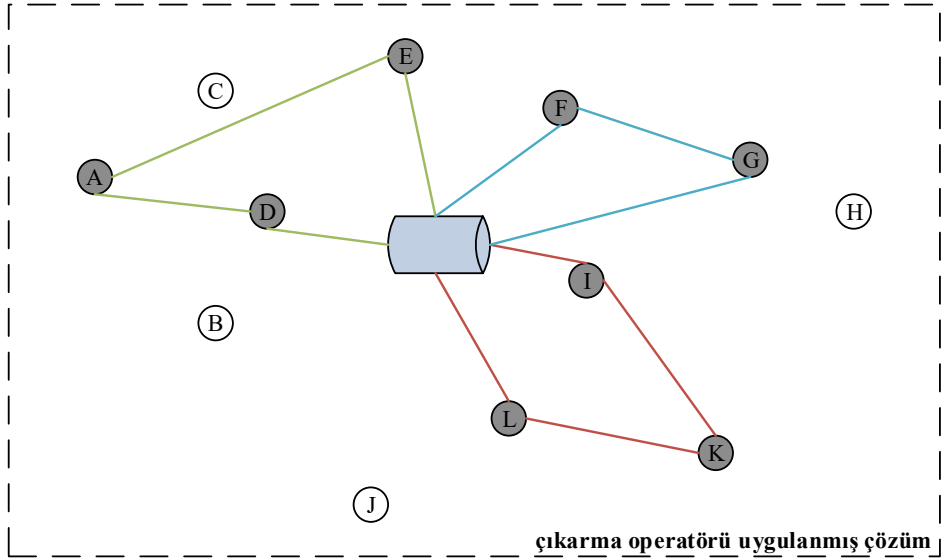
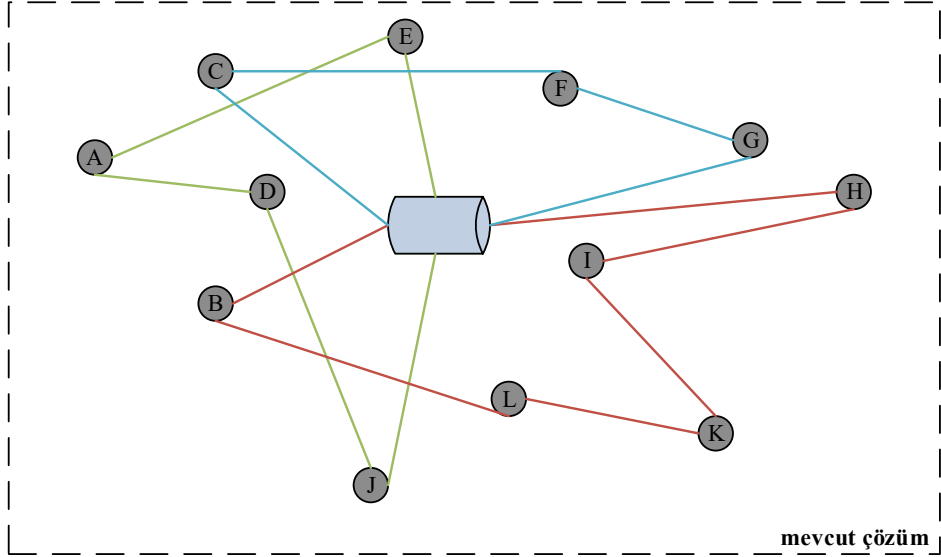
BKA algoritması da DKA gibi çözüm uzayını aramak için bir komşuluk mekanizması kullanır. BKA algoritmasında üretilen bir başlangıç çözümü çıkarma operatörü ile dejenere edilip daha sonra ekleme operatörüyle tamamlanarak yeni bir mevcut çözüm elde edilir. Bu bir dizi çıkarma ve ekleme işlemine dayalı arama prosedürü belirli bir durdurma koşulunu sağlayana kadar devam eder (Pisinger ve Ropke 2010, Azi ve diğ. 2014, Hof ve Schneider 2019).

BKA, çok büyük ölçekli komşuluk arama algoritmaları sınıfında bir sezgiseldir (Ahuja ve diğ. 2002, Ropke ve Pisinger 2006^b, Pisinger ve Ropke 2010). Büyük ölçekli

komşuluklarla arama yapmanın hem yerel optimum çözüm elde etmede hem de mevcut çözümlerden daha iyi çözümler üretmede etkin olabileceği görüşü hakimdir (Pisinger ve Ropke 2010). Ayrıca, başlangıç çözümünün kalitesinin düşük olduğu durumlarda adaptif bir şekilde uygulanan BKA ile bu çözümden kolayca uzaklaşılabilir (Demir ve diğ. 2012).

Çıkarma ve ekleme operatörlerinin çeşitli şekillerde uygulanması veya operatörlerin algoritma içinde bir ağırlığa sahip olması BKA algoritmasına adaptif bir yapı kazandırmaktadır (Azi ve diğ. 2012). Bu ağırlık, ilgili operatörün mevcut çözümü iyileştirip iyileştirmeme durumuna göre algoritma ilerledikçe sürekli olarak güncellenmekte ve mevcut çözümde daha fazla iyileştirme yapan operatör algoritma boyunca daha fazla uygulanmaktadır (Azi ve diğ. 2014).

Şekil 4.3'te bir ARP üzerinde ekleme ve çıkarma operatörlerinin nasıl uygulandığı gösterilmiştir. Bu şekil, mevcut çözümden yeni çözüm oluşturma adımlarını göstermesi açısından önemlidir. 12 müşteri ve tek depodan oluşan örnekte farklı renklerle gösterilen 3 rota oluşturulmuştur. J-D-A-E (yeşil), C-F-G (mavi) ve H-I-K-L-B (kırmızı) rotaları *mevcut çözüm* olarak belirlenmiştir. Daha sonra mevcut çözüme çıkarma operatörü uygulanarak B, C, H, J müşterileri çözümden atılmış ve *çıkarma operatörü uygulanmış çözüm* elde edilmiştir. Son olarak, *çıkarma operatörü uygulanmış çözüme* eksik müşteriler farklı rotalara dâhil edilerek eklenmiş ve *ekleme operatörü uygulanmış çözüm* elde edilmiştir. Bu durumda yeni rotalar D-B-A-C-E, F-G-H ve I-K-L-J olmuştur. Bu gösterim için Pisinger ve Ropke'nin (2010) çalışmasından faydalanılmıştır.



Şekil 4.3: Kapasite kısıtlı ARP için çıkarma ve ekleme operatörü uygulama örneği

Çıkarma operatörü uygulanırken, mevcut çözümden çıkarılacak düğümlerin belirlenmesi stokastik bir temele dayanır (Pisinger ve Ropke 2010). Literatürde şimdiye kadar farklı türdeki problemlere birçok çıkarma operatörü uygulanmıştır. Bu operatörler rastgele çıkarma (*random removal*) (Ropke ve Pisinger 2006^b, Pepin ve diğ. 2009, Sze ve diğ. 2017, Zulj ve diğ. 2018, Avcı ve Avcı 2019), en kötü değer/mesafe/zaman çıkarma (*worst removal*) (Ropke ve Pisinger 2006^a, Pepin ve diğ. 2009, Demir ve diğ. 2012, Akpınar 2016, Sze ve diğ. 2017, Zulj ve diğ. 2018, Avcı ve Avcı 2019), Shaw çıkarma (*Shaw removal*) (Shaw 1997, 1998, Ropke ve Pisinger 2006^b, Zulj ve diğ. 2018, Avcı ve Avcı 2019), yakınlık/zaman/talep-tabanlı çıkarmalar (*proximity-based, time-based, demand-based removals*) (Ropke ve Pisinger 2006^b, Pisinger ve Ropke 2007, Demir ve diğ. 2012, Avcı ve Avcı 2019) ve komşuluk/bölge/düğüm çıkarmalar (*neighborhood, zone, node removals*) (Demir ve diğ. 2012) olarak sıralanabilir.

Ekleme operatörü ise çıkarma aşamasında mevcut çözümden çıkarılan tüm düğümlerin etkin bir şekilde çözüme yeniden katılmasını sağlar ve komşu çözümler bu şekilde üretilmiş olur (Milthers 2009). Çıkarma operatöründe olduğu gibi, literatürde birçok farklı ekleme operatörü problemin yapısına uygun olarak adaptif bir şekilde uygulanmıştır. Bu operatörler rastgele ekleme (*random insertion*) (Avcı ve Avcı 2019), açgözlü ekleme (*greedy insertion*) (Demir ve diğ. 2012, Sze ve diğ. 2017, Chen ve diğ. 2018, Zulj ve diğ. 2018, Avcı ve Avcı 2019), pişmanlık ekleme (*regret insertion*) (Ropke ve Pisinger 2006^a, Azi ve diğ. 2010, Azi ve diğ. 2014, Akpınar 2016, Sze ve diğ. 2017, Chen ve diğ. 2018, Zulj ve diğ. 2018, Avcı ve Avcı 2019), Shaw ekleme (*Shaw insertion*) (Shaw 1997, 1998), gürültü fonksiyonu ile ekleme (*noise function insertion*) (Demir ve diğ. 2012) ve bölge tabanlı ekleme (*zone-based insertion*) (Ribeiro ve Laporte 2012, Chen ve diğ. 2018) olarak sayılabilir.

BKA'nın temel adımları Algoritma 3'te verilmiştir. Rastgele oluşturulan uygun bir başlangıç çözümüne (B^{bas}) sırayla çıkarma ve ekleme operatörleri uygulanarak yeni bir çözüm (B^{veni}) elde edilir. Yeni çözüm için literatürde farklı kabul kriterleri mevcut olmakla beraber, temel olarak amaç fonksiyonunda bir iyileşme sağlanması durumunda yeni çözüm mevcut en iyi çözüm (B^{en}) olarak kabul edilir (Ropke ve Pisinger 2006^a, Akpınar 2016). Bu adımlar önceden belirlenmiş bir durdurma kriteri sağlanana kadar tekrar edilir.

Algoritma 3: Temel BKA Algoritması (Pisinger ve Ropke 2010)

```
1 Bir başlangıç çözümü bul;  $B^{bas}$ 
2  $B^{en} \leftarrow B^{bas}$ 
3 tekrar et
4    $B^{yeni} \leftarrow B^{bas}$ 
5    $B^{bas}$ 'a çıkarma( $\zeta$ )/ekleme( $e$ ) operatörünü uygula;  $e(\zeta(B^{bas}))$  ve  $B^{yeni}$ 'yi üret
6   if  $f(B^{yeni}) > f(B^{en})$  then
7      $B^{en} \leftarrow B^{yeni}$ 
8   end
9    $B^{bas} \leftarrow B^{en}$ 
10 kadar (Durdurma kriteri)
11 döndür  $B^{en}$ 
```

Adaptif Büyük Komşuluk Arama (ABKA; *Adaptive Large Neighborhood Search: ALNS*) algoritması ise BKA algoritmasında birden fazla çıkarma-ekleme operatör çiftinin farklı şekillerde ve ağırlıklarda uygulanabildiği BKA algoritmasının genişletilmiş bir versiyonudur (Ropke ve Pisinger 2006^b, Pisinger ve Ropke 2010, Azi ve diğ. 2012, Azi ve diğ. 2014). Literatürde BKA algoritmasının adaptif olarak uygulandığı ve başarılı sonuçların elde edildiği birçok çalışma mevcuttur. Algoritmaya adaptif bir yapı kazandırmak için DKA'da komşuluk yapılarının, BKA'da ise çıkarma-ekleme operatörlerinin seçimlerinde rulet tekeri gibi yöntemler kullanılmaktadır (Hemmelmayr ve diğ. 2012).

BKA, ARP'ye ilk kez 1997 yılında Shaw tarafından uygulanmıştır. Yazar, BKA algoritmasının yöneylem araştırması alanındaki diğer meta-sezgisel algoritmalarla son derece rekabetçi olduğunu vurgulamıştır. Shaw (1997)'in bu görüşünü destekleyecek şekilde, BKA algoritmasının çözüm kalitesi açısından literatürdeki diğer algoritmalarla daha iyi sonuçlar elde ettiği, deneysel çalışmalarla ortaya konmuştur. Ropke ve Pisinger (2006^a), zaman pencereli topla-dağıt ARP için ABKA yöntemi önermişlerdir. Yazarlar, önerdikleri yöntemi literatürdeki 350'den fazla karşılaştırma problemine uygulamış ve çözüm kalitesi açısından ele alınan problemlerin yarısından daha fazlasında daha iyi sonuç elde etmişlerdir. Yazarlar yine aynı yıl yaptıkları diğer bir çalışmada (Ropke ve Pisinger 2006^b), zaman pencereli topla-dağıt ARP için BKA'nın farklı bir versiyonunu önermişlerdir. Literatürden alınan 338 karşılaştırma probleminin yaklaşık %67'sinden daha iyi sonuçlar elde edilmiştir. Pisinger ve Ropke (2007) zaman pencereli ARP, kapasite kısıtlı ARP, çok depolu ARP, mevkiye bağlı ARP ve açık ARP'leri topla-dağıt ARP'ye dönüştürmüş

ve Ropke ve Pisinger (2006^a) tarafından önerilen ABKA yöntemiyle çözüm üretmeye çalışmışlardır. Ele alınan bütün ARP türü problemler için literatürde bilinen en iyi yöntemlerin yaklaşık %38'inden daha iyi sonuçlar elde etmişlerdir. Azi ve diğ. (2014) yaptıkları çalışmada, dayanıksız malları taşıyan her bir aracın iş günü boyunca birden fazla güzergâh üzerinden teslimat işlemleri gerçekleştirdiği ve yeni müşteri taleplerinin dinamik olarak ortaya çıktığı ARP'yi ele almışlardır. Yazarlar, hizmet verilen müşteri sayısını maksimize etmek ve araçların kat ettiği toplam mesafeyi minimize etmek için ABKA önermişlerdir. Farklı yapılarında çıkarma ve ekleme operatörünün kullanıldığı çalışmada, yalnızca müşterilerin dâhil edildiği klasik operatörlerin aksine, müşteri, güzergâh ve iş günü seviyelerindeki operatörlerin uygulanmasının çok faydalı olduğu gösterilmiştir. Müşterilere toplam varış süresinin minimizasyonu olan kümülatif kapasite kısıtlı ARP için Ribeiro ve Laporte (2012) tarafından bir ABKA önerilmiştir. Yazarlar, ABKA'yı yakın zamanda geliştirilen iki yöntemle karşılaştırmak için literatürdeki problemleri ele almışlar ve ABKA'nin diğer iki yöntemden daha iyi sonuçlar sergilediğini göstermişlerdir. Akpınar (2016), kapasite kısıtlı ARP için KKO algoritması ve BKA'yı melez bir şekilde (KKO-BKA) uygulamayı önermiştir. Yazar, KKO'nun çözüm üretme mekanizmasını kullanarak iyi seviyede bir çeşitlendirme ile BKA'nin etkinliğini artırmayı amaçlamıştır. Chen ve diğ. (2018), dinamik ARP'yi sınırlı sayıda araç ve zor-zaman pencereli olarak incelemişlerdir. Alinaghian ve Shokouhi (2018) çok depolu, çok kompartımanlı ARP için matematiksel bir model sunmuş ve toplam araç sayısını ve araçların kat ettiği toplam mesafeyi minimize etmek için DKA ve ABKA algoritmalarını melez bir şekilde kullanmışlardır. Sarasola ve Doerner (2020), yaptıkları çalışmada müşterinin ilk aldığı teslimat ile son aldığı teslimat arasında geçen âtil süreyi ve maliyeti minimize etmek için bir ABKA algoritması önermişlerdir ve ele aldıkları örneklerde âtil süreyi yaklaşık %80'e kadar, maliyeti ise yaklaşık %10'a kadar azaltmışlardır.

ARP dışında ABKA algoritmasının başarılı bir şekilde yaygın olarak kullanıldığı problemler şu şekilde sıralanabilir: Çevre kirliliği rotalama problemi (Demir ve diğ. 2012, Demir ve diğ. 2014, Franceschetti ve diğ. 2017), periyodik konum rotalama problemi (Hemmelmayr ve diğ. 2017), sipariş gruplama problemi (Zulj ve diğ. 2018), çoklu gezici tamirci problemi (Avcı ve Avcı 2019), kesikli kısıtlı sırt çantası problemi (Hifi ve diğ. 2014), rihtım atama-rihtım vinci atama problemi (Iris ve diğ. 2017) ve elektrikli araç sıralama problemi (Wen ve diğ. 2016).

Literatürde ABKA algoritmasının kesikli optimizasyon problemlerine başarılı bir şekilde uygulanması, bu algoritmanın TSDP'nin çözümü için de iyi sonuçlar üretebileceği beklentisi bu tez çalışmasının motivasyonlarından biri olmuştur.

4.5 Savaşçı Genetik Algoritma

GA, Holland tarafından 1975 yılında geliştirilen evrimsel bir algoritmadır ve o tarihten beri birçok kombinatoriyal optimizasyon problemini çözmek için etkin bir şekilde kullanılmaktadır. Klasik bir GA genellikle rastgele oluşturulmuş bir popülasyonla başlar. Popülasyondaki bireyler kromozom olarak adlandırılırlar ve probleme özgü olarak kodlanırlar. GA optimizasyon süreci boyunca temel olarak üç operatör kullanır: seçim (*selection*), çaprazlama (*crossover*) ve mutasyon (*mutation*). Çaprazlama, farklı çözümler arasında bilgi alışverişi yapmak için uygulanırken, mutasyon popülasyondaki çeşitliliği artırır. İlk popülasyonun rastgele oluşturulup kromozomların belli özelliklere göre seçilip değerlendirilmesinden sonra, popülasyondaki kromozomlara farklı oranlarda çaprazlama ve mutasyon operatörleri uygulanır. Bu şekilde yeni popülasyon oluşturulur. Bu süreç, bir sonlandırma kriteri sağlanana kadar devam eder. GA ve uygulamaları hakkında daha detaylı bilgi için Taşan ve Tunalı (2008), Güner Gören ve diğ. (2010) ve Gen ve Lin (2014)'in çalışmalarına bakılabilir.

SGA, Eksin ve Erol (2001) tarafından önerilen GA-tabanlı yeni bir algoritmadır. Eksin ve Erol (2001) bu algoritmanın etkinliğini Sphere, Rosenbrock, Ackley vb. gibi matematiksel fonksiyonlar üzerinde test etmiş ve SGA'nın arama kapasitesi bakımından tatmin edici sonuçlar ürettiğini gözlemlemişlerdir. Erol ve Eksin'in 2006 yılındaki çalışmasında belirtildiği üzere, SGA, GA'ların iki temel operatörünü birleştirdiğinden, klasik GA'lara kıyasla hesaplama miktarını büyük ölçüde azaltmakta ve arama sürecini hızlandırmaktadır (Erol ve Eksin 2006).

Standart GA'da seçim operatörü ile popülasyondaki en iyi bireyler (probleme özel bilgileri içeren değişkenler) kopyalanmış olur. Algoritma ilerledikçe, her yinelemede üretilen popülasyon içinde amaç fonksiyonu değeri (uygunluk değeri: *fitness value*) açısından benzer veya aynı bireylerin seçilme olasılığı artacağı için çaprazlama operatörünün etkinliği azalacaktır. Bu sebeple popülasyonun yeniden

üretilme aşaması çeşitliliği artıracak ve yerel optimumdan kaçmayı sağlayacak şekilde yeniden uyarlanmalıdır. Bu uyarlamayı da içeren SGA aşağıdaki adımlardan oluşur (Eksin ve Erol 2001):

Adım 1: P adet bireyden oluşan popülasyon rastgele oluşturulur.

Adım 2: Oluşturulan popülasyon içinden iki birey rastgele seçilir.

Adım 3: Seçilen bireylerin uygunluk değerleri problemin özelliğine göre hesaplanır. Daha sonra, uygunluk değerlerinin göreceli farkı (Δ_r) Denklem 4.12'ye göre hesaplanır:

$$\Delta_r = \frac{|f(d_1) - f(d_2)|}{f(d_1) + f(d_2)} \quad (4.12)$$

Buradaki $f(d_1)$ ve $f(d_2)$ sırasıyla birinci (d_1) ve ikinci bireyin (d_2) amaç fonksiyonu uygunluk değerleridir.

Adım 4: Seçilen iki bireyin uygunluk değerleri karşılaştırılır. Eğer göreceli fark çok büyükse, kısmi üzerine yazma (*partial overwrite*) operatörü uygulanır. Uygunluk değerine göre daha az uygun olan (maksimizasyon probleminde daha küçük uygunluk değerine sahip olan, minimizasyon probleminde daha büyük uygunluk değerine sahip olan) birey bazı bilgilerini kaybeder. Yani bireyin yapısı değiştirilir. Eğer göreceli fark küçükse, klasik düzgün çaprazlama (*uniform crossover*) operatörü uygulanır. Bu koşulları matematiksel olarak ifade etmek gerekirse (ϑ ; $[0, 1]$ aralığında rastgele üretilen bir sayı olmak üzere);

- $f(d_1) > f(d_2)$ ve $\vartheta < \Delta_r$ sadece d_2 için kısmi üzerine yazma uygulanır
- $f(d_1) > f(d_2)$ ve $\vartheta > \Delta_r$ d_1 ve d_2 için birlikte düzgün çaprazlama uygulanır
- $f(d_1) < f(d_2)$ ve $\vartheta < \Delta_r$ sadece d_1 için kısmi üzerine yazma uygulanır
- $f(d_1) < f(d_2)$ ve $\vartheta > \Delta_r$ d_1 ve d_2 için birlikte düzgün çaprazlama uygulanır

Adım 5: Seçilen iki bireye popülasyon büyüklüğüyle ters orantılı olarak ($1/P$ oranında) mutasyon uygulanır. Bu da algoritmanın başında bazı parametrelerin atanmasına gerek olmadığı anlamına gelir.

Adım 6: Belirli bir durdurma kriteri sağlanana kadar *Adım 2*'ye geri dönülerek algoritma tekrar edilir.

Adım 4'teki operatörler, ele alınan problemin maksimizasyon problemi olduğu varsayımıyla uygulanmaktadır. Algoritmanın temel adımlarından da anlaşılacağı üzere, SGA'da her yinelemede sadece bir seçim, bir çaprazlama ve bir mutasyon operatörü uygulanmaktadır. Bu şekilde, her yinelemede 2 birey oluşturulması ve çaprazlama-mutasyon operatörlerinden sonra da 2 yeni birey oluşturulmasıyla toplamda 4 birey oluşturulur ve 4 bireyin uygunluk değerleri hesaplanır. Bu da klasik GA'ya göre daha az sayıda uygunluk değeri hesaplanması demektir.

Eksin ve Erol (2001), bir dizi matematiksel fonksiyonun değerlendirilmesinde klasik GA ile SGA'yı karşılaştırmışlardır. Sonuçlar SGA'nın arama kapasitesi bakımından klasik GA'dan daha iyi sonuçlar ürettiğini göstermiştir. Chuang ve diğ. (2008) ise mikroçip veri teknolojisi alanında gen özellik seçimi ve sınıflandırması için SGA tabanlı yöntem önermişlerdir. Önerilen yöntem literatürden alınan 10 mikroçip veri setine uygulanmıştır ve elde edilen sonuçlar önerilen yöntemin gen seçiminde sınıflandırma hatasını en aza indirdiğini göstermiştir. Benzer şekilde, Chuang ve diğ. (2012) tarafından yapılan başka bir çalışmada gen özellik seçimi ve sınıflandırması problemi ele alınmıştır. Yazarlar, ilgili problem için hem ikili PSO hem de SGA önermişlerdir ve birinci çalışmayla paralel sonuçlar elde etmişlerdir.

Literatürden de görüldüğü üzere, SGA daha önce bir üretim sistemi tasarımı problemine uygulanmamıştır. SGA her yinelemede algoritmanın başlangıcında üretilen popülasyon sayısından bağımsız sadece dört bireyin (kromozomun) amaç fonksiyonunu değerlendirdiği için arama süreci oldukça hızlıdır. SGA'da çaprazlama ve mutasyon operatörleri basit şekilde uygulanabilmektedir. GA'ya oranla arama sürecinin daha hızlı olması, yani hızlı yakınsama özelliği ve kolay uygulanabilir yapısı nedeni ile SGA'nın performansının TSDP'nin çözümündeki etkinliğinin araştırılması ve problemin yapısına uygun bir şekilde uyarlanması bu tez çalışmasının amaçlarından biridir.

4.6 Büyük Patlama – Büyük Çöküş Algoritması

Büyük patlama-büyük çöküş algoritması, SGA'nın yavaş yakınsama dezavantajının üstesinden gelmek için, Erol ve Eksin (2006) tarafından evrenin evriminden esinlenilerek geliştirilmiş, biyotaklit (*biomimicry*) bir meta-sezgisel arama yöntemidir. BP-BÇ algoritması popülasyon tabanlı ve yinelemelidir. Algoritma, çok az parametre gerektirdiğinden uygulanması da çok basittir. Ayrıca, algoritma, kısa hesaplama süresinde hızlı yakınsaması nedeniyle tercih edilmektedir. Algoritma, evren hakkındaki Büyük Patlama (*Big Bang*) ve Büyük Çöküş (*Big Crunch*) teorisinden esinlenmiştir. BP-BÇ algoritmasının TA algoritmasında olduğu gibi arama süreci ile ilgili bilgileri kullanan büyük çöküş aşamasına sahip olması, algoritmayı güçlendiren ana özelliklerdendir.

Erol ve Eksin (2006), önerdikleri BP-BÇ algoritmasını matematiksel fonksiyonlar üzerinde test etmişlerdir. Deneysel çalışma sonucunda BP-BÇ algoritmasının, SGA'dan hem yakınsama hem de amaç fonksiyonu açısından daha iyi sonuçlar ürettiği gözlenmiştir.

BP-BÇ algoritması büyük patlama ve büyük çöküş isimli iki temel aşamadan oluşmaktadır. Algoritma, başlangıç popülasyonunun rastgele oluşturulması açısından GA'ya benzemektedir. Büyük patlama aşamasında, düzgün (*uniform*) şekilde aday çözümler arama uzayına dağıtılır. Büyük çöküş aşamasında ise arama uzayına dağılan çözümler tek bir çözüme dönüştürülür. Bu aşama yakınsama ya da yoğunlaştırma operatörü olarak da adlandırılır. Buradaki dönüştürme, aday çözümler arasında ya kütle merkezine (*center of mass*) ya da amaç fonksiyonunun değerine göre en uygun çözümü seçmek olarak tanımlanabilir. Yinelemeli arama boyunca, büyük patlama-genişleme (*expansion*) ve büyük çöküş-daralma (*contraction*) adımları art arda kullanılır. Bu adımlar önceden belirlenmiş bir durdurma koşulu sağlanana kadar tekrar eder (Erol ve Eksin 2006). BP-BÇ algoritmasının büyük patlama aşamasında popülasyon üretme mekanizması ve büyük çöküş aşamasındaki en iyi (ya da merkezdeki) bireyi seçme mekanizması değiştirilerek uygulanabilmektedir.

Bir yinelemede bir büyük patlama ve bir büyük çöküş aşaması uygulanır. Büyük patlama aşamasında, ilk yinelemede, popülasyon büyüklüğü (P) kadar aday çözüm, probleme özgü sınırlandırılmış arama uzayına dağıtılır. Bu aşamayı büyük

çöküş aşaması takip eder ve P aday çözüm tek bir çözüme/kütle merkezine indirgenir. Erol ve Eksin (2006) algoritmayı ilk defa önerdikleri çalışmada, kütle merkezini amaç fonksiyonunun tersi olarak ifade etmişlerdir. Kütle merkezi (d^m) Denklem 4.13'e göre aşağıdaki gibi hesaplanmaktadır (Erol ve Eksin 2006):

$$\vec{d}^m = \frac{\sum_{i=1}^P \frac{1}{f(d)^i} \vec{d}^i}{\sum_{i=1}^P \frac{1}{f(d)^i}} \quad (4.13)$$

d^m vektörel kütle merkezini, d^i P boyutlu arama uzayında üretilen vektörel bir noktayı, $f(d)^i$ bu noktanın amaç fonksiyonu ile elde edilmiş uygunluk değerini ve P ise popülasyon büyüklüğünü göstermektedir.

Büyük çöküş aşamasından sonraki aşamada, ikinci yinelemede, yine büyük patlama aşaması kullanılarak yeni P adet aday çözüm üretilir. Yeni aday çözümler üretilmesinde birkaç farklı alternatif olmakla birlikte, bir önceki aşamada elde edilen kütle merkezi olarak (d^m) ifade edilen noktanın kalıtsal özelliği kullanılarak yeni çözümler elde etmek global optimizasyon açısından tercih edilmektedir. Algoritmanın yineleme sayısı arttıkça d^m noktasının etrafında yeni aday çözümler üretmek için kullanılan normal dağılımın fonksiyonuna ait standart sapma değeri de küçülür. Bu sebeple yineleme sayısı arttıkça çözümlerin dağıldığı arama uzayının sınırları daralır. BP-BÇ algoritmasının teknik detayları ve örnek bir matematiksel fonksiyon üzerinde uygulama örneği için Erol ve Eksin'in (2006) çalışması incelenebilir. BP-BÇ algoritmasının temel adımları aşağıdaki gibidir (Erol ve Eksin 2006):

Adım 1: P adet aday birey içeren popülasyon arama uzayının sınırları dikkate alınarak rastgele oluşturulur.

Adım 2: Oluşturulan P adet aday bireyin uygunluk değerleri hesaplanır.

Adım 3: Denklem 4.13'e göre kütle merkezi hesaplanır ya da en iyi amaç fonksiyonu değerine sahip birey kütle merkezi olarak seçilir.

Adım 4: Bir rastgele sayı (r) kullanılarak kütle merkezinin etrafında yeni aday çözümler üretilir. Yineleme sayısı arttıkça bu sayının küçülmesini formülize eden Denklem 4.14 aşağıdaki gibidir:

$$d^{yeni} = d^m + ur/m \quad (4.14)$$

u parametrelerin üst limitini, r bir normal rastgele sayıyı, m yineleme sayısını ve d^{yeni} yeni üretilen aday çözümleri göstermektedir.

Adım 5: Durdurma kriteri sağlanana kadar *Adım 2*'ye geri dönülerek algoritma tekrar edilir.

BP-BÇ algoritması literatürde farklı problemler üzerinde kullanılmıştır. Alataş (2011), BP-BÇ algoritmasının aşamalarındaki işlevleri değiştirerek algoritmanın 3 farklı varyasyonunu (düzgün BP-BÇ, düzgün BP-kaotik BÇ ve BP-kaotik BÇ) geliştirmiştir. Yazar, 4 farklı matematiksel fonksiyonu 2 farklı durdurma kriteriyle simule etmiş ve düzgün BP-kaotik BÇ varyasyonu kıyaslanan bütün varyasyonlar arasında en iyi sonucu vermiştir. Literatür incelendiğinde BP-BÇ algoritmasının inşaat mühendisliği alanındaki sürekli optimizasyon problemlerinde sıklıkla kullanıldığı görülmüştür. Kafes yapılarının optimum tasarımında Kaveh ve Talatahari (2009) tarafından BP-BÇ algoritması PSO ile birlikte uygulanarak melez bir algoritma geliştirilmiştir. Geliştirilen melez algoritma, BP-BÇ algoritması, GA, KKO, PSO ve harmoni arama algoritması ile karşılaştırılmıştır. Melez BP-BÇ algoritmasının karşılaştırılan diğer algoritmalara göre etkinlik ve sağlamlık açısından daha iyi olduğu belirtilmiştir. Yazarların 2010 yılında yaptıkları ve melez BP-BÇ algoritmasını kullandıkları diğer bir çalışmada (Kaveh ve Talatahari 2010), Schwedler ve yivli kubbelerin optimum tasarımı ele alınmıştır. Çelik çerçeve yapılarının koda dayalı tasarım optimizasyonu için Hasançebi ve Azad (2012) tarafından BP-BÇ algoritmasının büyük patlama aşamasında üstel dağılım kullanan farklı bir varyasyonu (üstel BP-BÇ) önerilmiştir. Klasik BP-BÇ algoritmasının çelik çerçevelerin tasarım optimizasyonunda etkin çözümler üretememesinden dolayı etkili bir modifikasyonla sağlam sonuçların elde edilmesi sağlanmıştır. Beton çerçevelerin mevzuata uygun şekilde tasarlanması ve dolayısıyla toplam maliyet ve CO₂ emisyonunun minimizasyonu için Camp ve Huq (2013) tarafından melez BP-BÇ algoritması kullanılmıştır. Melez BP-BÇ algoritması hem GA hem de TB ile karşılaştırılmış, melez BP-BÇ algoritmasının daha iyi sonuçlar ürettiği gösterilmiştir. BP-BÇ algoritmasını PSO ile melez bir şekilde uygulayan başka bir çalışma Fernandez Caban ve Masters (2018) tarafından sunulmuştur. Çalışmada mukavemet ve servis

edilebilirlik gereksinimleri karşılanırken ağırlığı en aza indiren büyük ölçekli çerçeve yapıları incelenmiştir.

Yapısal sistemlerin tasarımında parametre tahmini için bir BP-BÇ algoritması Tang ve diğ. (2010) tarafından önerilmiştir. Yüksek boyutlu ve çok modlu olarak formüle edilen problem için gürültülü sinyaller ve sınırlı çıktı verisi gibi parametreler ele alınmış ve önerilen algoritma ile GA ve PSO'ya göre daha iyi sonuçlar elde edilmiştir.

BP-BÇ algoritmasının diğer yaygın kullanım alanı ise elektrik-elektronik mühendisliği alanıdır (Desai ve Prasad 2013, Biradar ve diğ. 2016). Genç ve diğ. (2013), BP-BÇ algoritmasını bir yerel arama algoritmasıyla güçlendirerek GA, PSO gibi meta-sezgisel algoritmalarla kıyaslamışlardır. Sedighizadeh ve diğ. (2013), dengeli ve dengeli olmayan dağıtım sistemlerini bulanık bir çerçevede çok amaçlı çözmek için BP-BÇ algoritması ve PSO'nun melez bir versiyonunu kullanmışlardır.

Literatür araştırmasından da görüleceği üzere üretim sistemleri alanında BP-BÇ algoritması ile yapılmış bir çalışmaya rastlanmamıştır. Algoritmanın büyük patlama aşamasında kullanılan parametreler ile çeşitlilik artırılabilmekte ve büyük çöküş aşamasında popülasyondaki en iyi bireyin seçimi ile yakınsama özelliği güçlendirilebilmektedir. Algoritmanın basit yapısı ve TA algoritmasındaki gibi belirli bilgileri depolayarak arama esnasında kullanma özelliğinden dolayı TSDP'nin çözümünde etkili olması beklenmektedir. Bu amaçla bu tez çalışması kapsamında BP-BÇ algoritmasının performansı TSDP için test edilmiştir.

4.7 Özet

Bu bölümde, TSDP'nin çözümünde kullanılan değerlendirme yöntemi olan ayrıştırma yöntemi detaylı olarak anlatılmıştır. Ardından çalışmada kullanılan optimizasyon algoritmaları; değişken komşuluk arama algoritması, büyük komşuluk arama algoritması, savaşçı genetik algoritma ve büyük patlama-büyük çöküş algoritması alt bölümlerde tanıtılmıştır. Ayrıca her bir algoritmanın literatürdeki mevcut uygulamaları ilgili alt bölümlerde verilmiştir.

TSDP'yi çözmek için literatürde birçok meta-sezgisel algoritma kullanılmış olsa da en uygun olan algoritmayı bulma sorusu henüz tatmin edici bir şekilde cevaplanmamıştır. Ayrıca, optimum çözüm arama süresinde çeşitliliğin sağlanması, erken yakınsama ve parametre değerlerinin belirlenmesi meta-sezgisel algoritmalar için hala araştırılması gereken konular arasındadır. Bu nedenle, TSDP'nin çözümünde daha önce kullanılmayan komşuluk tabanlı arama algoritmaları olan değişken komşuluk arama algoritması ve büyük komşuluk arama algoritması ile popülasyon tabanlı olan savaşıcı genetik algoritma ve büyük patlama-büyük çöküş algoritması problemin çözümünde kullanılmıştır.

5. TAMPON STOK DAĞILIMI PROBLEMİ İÇİN DEĞİŞKEN KOMŞULUK ARAMA TABANLI BİR ÇÖZÜM YAKLAŞIMI

5.1 Giriş

Bu tez çalışmasının amacı üretim hatlarında tampon stok dağılımı problemine meta-sezgisel yöntemlerle çözüm üretmektir. Bu amaçla, tezin giriş bölümünde de belirtildiği üzere komşuluk arama tabanlı ve popülasyon tabanlı meta-sezgisel yöntemler kullanılmıştır. Bu bölümde üretim hatlarında tampon stok dağılımı probleminin çözümü için önerilen değişken komşuluk arama algoritmasının ayrıntılarına değinilecektir.

Makinelerin bozulmaya maruz kaldığı üretim hatlarında tampon stok dağılımı problemi üretim oranının maksimizasyonu (*Problem 1*) ve toplam tampon stok miktarı minimizasyonu (*Problem 2*) için çözülmüştür. Bu problemin yapısına uygun bir DKA algoritması geliştirilmiştir. İyi bir başlangıç çözümünün arama için harcanacak eforu azaltacağı göz önünde bulundurularak iki başlangıç sezgiseli önerilmiş ve bu sezgisellerin arama performansı üzerindeki etkisi gözlenmiştir. Önerilen DKA algoritmasının performansı literatürde mevcut test problemleri üzerinde denenerek etkinliği kanıtlanmıştır.

Bölümün geri kalanı şu şekilde organize edilmiştir. Bölüm 5.2’de önerilen başlangıç sezgiselleri tanıtılmıştır. Bölüm 5.3’te önerilen DKA algoritması ayrıntılı olarak anlatılmıştır. Bölüm 5.4’te önerilen algoritmanın ve başlangıç sezgisellerinin performansı deneysel çalışmalar ile test edilmiştir. Bölüm 5.5’te çalışmadan elde edilen bulgular sunulmuş ve son olarak Bölüm 5.6’da yapılan çalışmanın sonuçları özetlenmiştir.

5.2 Önerilen Başlangıç Çözüm Sezgiselleri

Literatürden de bilindiği üzere bir sezgisel algorithmada iyi bir başlangıç çözümü ile en iyi/optimum çözüme ulaşmak daha az arama eforu gerektirmektedir. Bu

nedenle bu tez çalışması kapsamında rastgele başlangıç çözümünün yanı sıra tampon stok dağılımı problemine özgü olarak iki farklı başlangıç çözümü üretme yöntemi önerilmiştir.

5.2.1 Rastgele Başlangıç Çözümü

Rastgele başlangıç çözümü üretilirken, $K-1$ tampon stok alanına her biri negatif olmayan tamsayı olmak koşuluyla toplam tampon stok miktarını (N) sağlayacak şekilde tampon stok değerleri rastgele atanır. Çalışmanın geri kalanında, rastgele çözüm ile başlayan DKA algoritması DKA-Rastgele olarak isimlendirilecektir.

5.2.2 Modifiye Edilmiş Kâse Fenomeni Sezgiseli

Hillier ve diğ. (1993) tarafından önerilen Ters Kâse Fenomeni, belirli toplam tampon stok sayısı (N) ve makine sayısına (K) göre hattın ortalarına doğru daha fazla, hattın kenarlarına doğru daha az tampon stokun hattın geneline simetrik bir şekilde atanması durumunda hattın üretim oranının arttığını çeşitli çalışmalarla ortaya koymuş bir yöntemdir. Oluşan tampon stok konfigürasyonunun ters kâseye benzemesi nedeniyle yöntem literatürde “ters kâse fenomeni” olarak isimlendirilmiştir. Bu tez çalışması kapsamında Hillier ve diğ. (1993) tarafından önerilen “ters kâse fenomeni” yöntemi güvenilir olmayan hatlar için modifiye edilerek kullanılmıştır. Önceki çalışmalarda (Hillier ve diğ. 1993, Hillier 2000), Ters Kâse Fenomeni yaklaşımı değişken üretim süresi içeren güvenilir üretim hatlarında kullanılmıştır. Genellikle, optimum tampon stok konfigürasyonlarının $\{N_1, N_1 + 1, \dots, N_1 + 1, N_1\}$, $\{N_1, N_1 + 2, \dots, N_1 + 2, N_1\}$, $\{N_1, N_1 + 3, \dots, N_1 + 3, N_1\}$ ve $\{N_1, N_1 + 0,03KN_1, \dots, N_1 + 0,03KN_1, N_1\}$ şeklinde olduğu görülmüştür. Bu yöntem, bu tez çalışması kapsamında ilk defa güvenilir olmayan hatlar için bir başlangıç tampon stok konfigürasyonu oluşturmak üzere bir başlangıç sezgiseli olarak önerilmiştir. Tezin ilerleyen bölümlerinde bu sezgisel Modifiye Edilmiş Kâse Fenomeni (MEKF) Sezgiseli olarak anılacaktır.

MEKF sezgiseli ile tampon stok konfigürasyonu oluşturulurken aşağıdaki formül kullanılmaktadır:

$$N = \{N_1, 2N_1, 3N_1, \dots, 3N_1, 2N_1, N_1\} \quad (5.1)$$

Bu formülde, N_1 ilk tampon stok alanındaki (B_1) tampon stok miktarını göstermektedir ve toplam tampon stok kapasitesinin N 'lerin katsayıları toplamına bölünmesiyle elde edilmiştir. Elde edilen tampon stok konfigürasyonundaki her bir değer en yakındaki tamsayı değerine yuvarlanarak geçici bir tampon stok konfigürasyonu elde edilir. Daha sonra elde edilen geçici toplam tampon stok konfigürasyonundaki toplam stok miktarının N değerini sağlayıp sağlamadığının kontrolü yapılır. Eğer geçici toplam tampon stok konfigürasyonundaki değerlerin toplamı N 'den büyükse hattın başındaki ve sonundaki değerler fark kadar azaltılır. Tercihen önce hattın sonundaki tampon stok değeri, daha sonra hattın başındaki tampon stok değeri azaltılır. Diğer taraftan, eğer geçici toplam tampon stok konfigürasyonunun toplam değeri N 'den küçükse hattın orta noktasındaki tampon stok değerleri N değerini sağlayacak şekilde artırılır. Bu başlangıç sezgiseli ile başlayan DKA algoritması, tezin geri kalanında DKA-MEKF olarak isimlendirilecektir.

MEKF sezgiselinin kavramsal algoritması Algoritma 4'te verilmiştir. Örneğin; 5 makineli bir hatta toplam 31 tampon stok atanacaksa Denklem (5.1)'e göre ilk geçici konfigürasyon $\{5, 10, 10, 5\}$ şeklinde elde edilir. Elde edilen geçici tampon stok konfigürasyonunda toplam tampon stok miktarı 30 olduğu için, N sayısına ulaşmak için gerekli tampon stok miktarı 2. tampon stok alanına eklenerek başlangıç çözümü $N^{bas} = \{5, 11, 10, 5\}$ olarak elde edilir.

Algoritma 4: Modifiye Edilmiş Kase Fenomeni Sezgiseli

girdi : makine sayısı (K), toplam tampon stok miktarı (N)

çıkıtı : başlangıç tampon stok konfigürasyonu (N^{bas} ; çözüm uygunsa *DOĞRU*, değilse *YANLIŞ*)

```
1   $N \leftarrow (0, 0, \dots, 0) \leftarrow (N_1, N_2, \dots, N_{K-1})$  ayarla
2   $N^{mekf} \leftarrow (0, 0, \dots, 0) \leftarrow (N^{mekf}_1, N^{mekf}_2, \dots, N^{mekf}_{K-1})$  ayarla
3  tamponStokSayac  $\leftarrow 0$  ayarla
4  for ( $i \leftarrow 1$  to ( $K-1$ )) do
5      if  $i \leq ((K-1)/2)$  then
6           $\lfloor$  tamponStokSayac  $\leftarrow$  tamponStokSayac + 1
7      else if  $i = ((K-1)/2) + \text{mod}(K, 2)$  then
8           $\lfloor$  tamponStokSayac  $\leftarrow$  tamponStokSayac
9      else if  $i > ((K-1)/2)$  then
10          $\lfloor$  tamponStokSayac  $\leftarrow$  tamponStokSayac - 1
11     end
12      $N_i \leftarrow$  tamponStokSayac
13 end
14 TOPLAM_tamponStokSayac  $\leftarrow$  TOPLAM( $N_i$ ) ayarla
15 for ( $i \leftarrow 1$  to ( $K-1$ )) do
16      $N^{mekf}_i \leftarrow N_i / \text{TOPLAM\_tamponStokSayac} * N$ 
17      $N^{mekf}_i \leftarrow (N^{mekf}_i)$ 
18     if ( $N^{mekf}_i \leq 0$ ) then
19          $\lfloor$  döndür YANLIŞ
20     end
21 end
22  $FARK \leftarrow N - \text{TOPLAM}(N^{mekf}_i)$ 
23 if  $FARK < 0$  then
24      $HB \leftarrow 0$  (hattın başındaki tampon stok değerleri)
25      $HS \leftarrow 0$  (hattın sonundaki tampon stok değerleri)
26     if  $N^{mekf}_{K-1} \neq 0$  then
27          $HB \leftarrow \lfloor (FARK)/2 \rfloor$ 
28          $HS \leftarrow \lfloor (FARK)/2 \rfloor$ 
29         for ( $i \leftarrow K-1$  to ( $K-1 + HS + 1$ )) do
30              $\lfloor N^{mekf}_i \leftarrow N^{mekf}_i - 1$ 
31         end
32         for ( $i \leftarrow 1$  to ( $-HB$ )) do
33              $\lfloor N^{mekf}_i \leftarrow N^{mekf}_i - 1$ 
34          $\lfloor$  end
35     else if  $N^{mekf}_{K-1} = 0$  then
36          $0 \leftarrow$  sifirSayac
37         for ( $i \leftarrow 1$  to ( $(K-1)/2$ )) do
38             if  $N^{mekf}_i < 1$  then
39                  $\lfloor$  sifirSayac  $\leftarrow$  sifirSayac + 1
40             end
41         end
42         for ( $i \leftarrow (1 - \text{sifirSayac})$  to ( $1 - \text{sifirSayac} + FARK + 1$ )) do
43              $\lfloor N^{mekf}_i \leftarrow N^{mekf}_i - 1$ 
44          $\lfloor$  end
45     end
46 else
47     for ( $i \leftarrow ((K-1)/2)$  to ( $((K-1)/2) + FARK - 1$ )) do
48          $\lfloor N^{mekf}_i \leftarrow N^{mekf}_i + 1$ 
49     end
50 end
51  $N^{bas} \leftarrow N^{mekf}$ 
52 döndür DOĞRU
```

5.2.3 PaVi Sezgiseli

Papadopoulos ve Vidalis (2001) güvenilir olmayan üretim hatları için Ters Kâse Fenomenine benzer bir tampon stok dağılımı oluşturma yöntemi önermişlerdir. Önerdikleri yöntemde hattın ortasına daha fazla, kenarlarına daha az tampon stok dağıtılırken üretim hattında darboğaza neden olan makineler de dikkate alınmaktadır. Bu amaçla makineler, Denklem 2.2 ile belirlenen ortalama etkin servis oranlarına (ρ_i) göre sıralanmakta ve daha sonra önceden tanımlanmış bazı kurallara göre makinelerin arasına tampon stok atamaları yapılmaktadır.

Papadopoulos ve Vidalis (2001) tarafından önerilen yöntem bu tez çalışmasında sadece bir başlangıç çözümü elde etmek için kullanılmıştır. Çalışmanın geri kalanında bu başlangıç yöntemi PaVi sezgiseli olarak anılacaktır. Bu başlangıç sezgiseli ile başlayan DKA algoritması, tezin geri kalanında DKA-PaVi olarak isimlendirilecektir. Yöntemin ayrıntıları için Papadopoulos ve Vidalis (2001)'in çalışmasına bakılabilir. Burada yöntemin uygulanışı 5 makine ve 31 toplam tampon stok atamasının yapıldığı bir örnek üzerinde anlatılmıştır (veriler için bkz. Bölüm 5.4.1).

PaVi sezgiselini uygulayabilmek için öncelikle her makinenin ortalama etkin servis oranı Denklem (2.2) ile aşağıdaki gibi hesaplanır:

$\rho_1 = 0,6452$, $\rho_2 = 0,8978$, $\rho_3 = 0,6471$, $\rho_4 = 0,7586$, $\rho_5 = 0,7879$ şeklinde hesaplanır.

Daha sonra sırasıyla aşağıdaki adımlar uygulanır:

Adım 1: Makineler (M_1, M_2, \dots, M_K) ortalama etkin servis oranlarına göre artan sırada (M_1', M_2', \dots, M_K') sıralanır.

$$M_1' = M_1 \quad M_2' = M_3 \quad M_3' = M_4 \quad M_4' = M_5 \quad M_5' = M_2.$$

Adım 2: 'İyi' bir başlangıç tampon stok konfigürasyonu belirlemek için yazarlar tarafından doğrusal tampon stok dağılımı şeması (*linear buffer allocation scheme*: LBAS; DTSPŞ) olarak isimlendirilen bir prosedür aşağıdaki gibi uygulanır:

(a) Tampon stok alanı hattın merkezine doğruysa ve sonraki makine M_i ' ise, bu alana $2(K+1-i)$ puan atanır.

(b) Tampon stok alanı hattın sonuna doğru ve sonraki makine M_i ' ise, bu alana $2(K+1-i) - 1$ puan atanır.

(c) Merkezi tampon stok alanı bir taneysse K puan, merkezi tampon stok alanı iki taneysse her birine $K/2$ puan atanır.

Yukarıda belirtilen prosedür örnek verilere uygulandığında tampon stok alanlarının toplam puanı aşağıdaki gibi hesaplanır:

(a) Tampon stok alanı B_1 'e $2(5+1-1) = 10$ puan atanırken, tampon stok alanı B_2 'ye $2(5+1-2) = 8$ puan atanır.

Tampon stok alanı B_3 'e $2(5+1-2) = 8$ puan atanırken, tampon stok alanı B_4 'e $2(5+1-4) = 4$ puan atanır.

Tampon stok alanı B_2 'ye $2(5+1-5) = 2$ puan atanırken, tampon stok alanı B_3 'e $2(5+1-3) = 6$ puan atanır.

(b) Tampon stok alanı B_1 'e $2(5+1-5)-1 = 1$ puan atanırken, tampon stok alanı B_4 'e $2(5+1-3)-1 = 5$ puan atanır.

(c) İki tane merkezi tampon stok alanı vardır. Tampon stok alanı B_2 'ye $5/2 = 2,5$ puan atanırken, tampon stok alanı B_3 'e $5/2 = 2,5$ puan atanır.

Her bir tampon stok alanının puanı sırasıyla 11, 12,5, 16,5 ve 9 olarak elde edildikten sonra toplam tampon stok sayısı ($N = 31$) her bir tampon stok alanının hesaplanmış puanı dikkate alınarak dağıtılır ve başlangıç tampon stok değerleri aşağıdaki gibi elde edilir:

$$N_1 = 6,95 \approx 7 \quad N_2 = 7,91 \approx 8 \quad N_3 = 10,44 \approx 10 \quad N_4 = 5,69 \approx 6$$

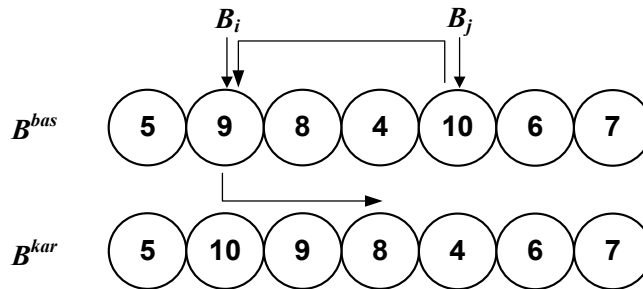
Adım 3: Bu yöntemi kullanmamızdaki amaç sadece 'iyi' bir başlangıç çözümü elde etmek olduğu için Papadopoulos ve Vidalis (2001) tarafından önerilen yöntemdeki üçüncü adım olan optimizasyon adımı bu tez çalışmasında uygulanmamıştır.

5.3 Önerilen DKA Algoritması

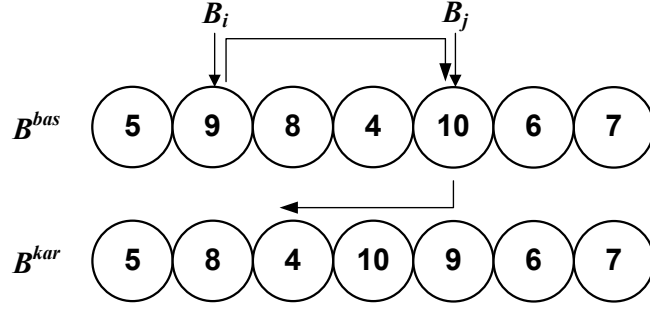
5.3.1 Komşuluk Yapıları

DKA algoritmasının temelini komşuluk yapıları oluşturmaktadır. Kullanılacak olan komşuluk yapısı, kaç tane komşuluk yapısı kullanılacağı ve bu komşuluk yapısının hangi sırada kullanılacağı algoritmanın etkinliğini belirleyen en önemli parametrelerdir. Bu çalışmada, TSDP çözümü için 4 farklı komşuluk yapısı kullanılmıştır. Bu komşuluk yapıları sağ rotasyon (*right rotation*), sol rotasyon (*left rotation*), ters çevirme (*inversion*) ve yer değiştirmedir (*swap*). DKA'daki komşuluk yapıları tek çözüm üzerinde işlem yaptığından GA'daki mutasyon operatörüne benzemektedir. Bu komşuluk yapıları kullanılırken, toplam tampon stok sayısı sabit kalacak şekilde tampon stokların yerleri değiştirilmektedir.

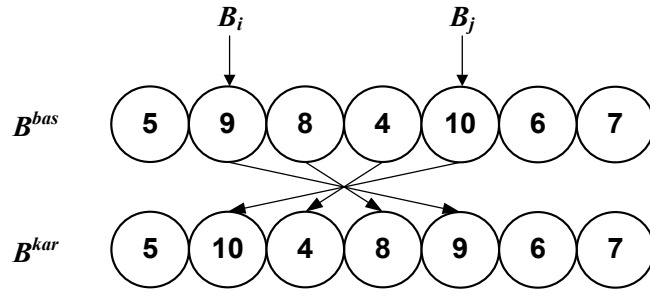
Şekil 5.1-5.4'te 7 tampon stok alanının olduğu ve toplam 49 tampon stoğun dağıtılacağı bir problemde komşuluk yapılarının nasıl uygulandığı gösterilmiştir. Bu gösterimlerde B^{bas} başlangıçtaki tampon stok dağılımını gösterirken, B^{kar} komşuluk yapıları uygulandıktan sonra elde edilen tampon stok konfigürasyonunu göstermektedir. Sağ rotasyon ve sol rotasyonda, iki tampon stok alanı (B_i ve B_j) rastgele seçilir. Ardından, seçilen tampon stok alanlarıyla başlayıp biten tampon stok konfigürasyon aralığı sağa (Şekil 5.1) veya sola (Şekil 5.2) döndürülür. Ters çevirmede ise mevcut tampon stok konfigürasyonunda rastgele seçilen iki alan arasındaki tampon stok alanları tersine çevrilir (Şekil 5.3). Son olarak yer değiştirmede, Şekil 5.4'te gösterildiği gibi mevcut tampon stok konfigürasyonunda rastgele seçilen iki konumdaki tampon stok değerleri karşılıklı olarak değiştirilir.



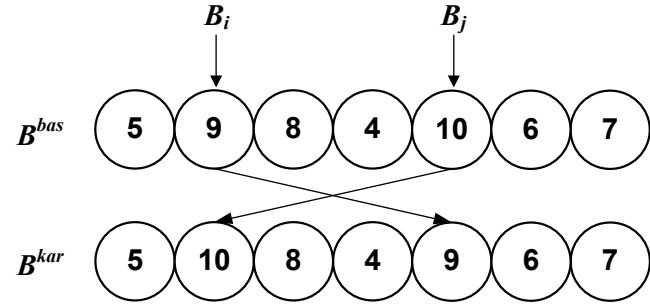
Şekil 5.1: Sağ rotasyon komşuluk yapısı



Şekil 5.2: Sol rotasyon komşuluk yapısı



Şekil 5.3: Ters çevirme komşuluk yapısı



Şekil 5.4: Yer değiştirme komşuluk yapısı

Örnek başlangıç tampon stok konfigürasyonuna (B^{bas}) sırasıyla bütün komşuluk yapıları uygulandığında farklı tampon stok konfigürasyonları (B^{kar}) elde edildiği görülmektedir. Bu komşuluk yapıları rulet tekeri yöntemi ile algoritma boyunca adaptif bir şekilde kullanılmaktadır. Algoritmanın başlangıç aşamasında, her komşuluk yapısının seçim olasılığı eşittir ve rulet tekeri yöntemi kullanılarak rastgele seçilmektedir. Her yinelemede, amaç fonksiyonunun değerini (üretim oranı değeri) iyileştiren komşuluk yapısı kaydedilmektedir. Bu şekilde üretim oranı değerinde iyileştirme sağlayan komşuluk yapısının sonraki yinelemeler için seçim olasılığı

artırılmaktadır. Ayrıca, amaç fonksiyonunu iyileştiren komşuluk yapılarının çeşitliliği artırılarak farklı yerel optimum noktalarda kaliteli çözümlere ulaşma olasılığının da artırılması beklenmektedir. Algoritmanın başlangıcında belli sayıda yinelemede kullanılmayan komşuluk yapısının rulet tekeri yöntemine göre belli yinelemeden sonra düşük olasılıkla da olsa kullanılması söz konusudur. Birden fazla komşuluk yapısının kullanılması kalitesiz çözüm üretilmesinin önüne geçilmesine ve dolayısıyla algoritmanın yerel optimuma takılmasını da engellemektedir (Muelas ve diğ. 2013, Mladenovic ve diğ. 2020).

5.3.2 Yerel Arama

DKA algoritmasında başlangıç çözümüne komşuluk yapıları uygulandıktan sonra elde edilen çözüme bir yerel arama algoritması uygulanır. Bu tez çalışmasında yerel arama algoritması olarak Artırma-Azaltma (Increment-Decrement: *IncDec*) stratejisi olarak isimlendirilen bir yerel arama operatörü önerilmiştir. Bu stratejide sırayla iki tampon stok çifti $((N_i, N_j), (i \neq j))$ ve $i, j \in (1, \dots, K-1)$ seçilir ve ilk seçilen tampon stokun kapasitesi belli bir miktar artırılırken toplam tampon stok sayısını sabit tutabilmek için diğeri aynı miktarda azaltılır. Bu miktarın değeri $(id \in \mathbb{Z}^+)$, algoritmanın başlangıç aşamasında belirlenir ve algoritma ilerledikçe problemin büyüklüğüne bağlı olarak sistematik bir şekilde değiştirilir. Bu değer, tez çalışmasında ele alınan büyük boyutlu problemler (20, 30 ve 40 makineli) için Demir ve diğ. (2012) tarafından önerildiği gibi toplam tampon stok kapasitesinin %1'i olarak belirlenmiştir.

IncDec stratejisine göre oluşan bütün aday tampon stok konfigürasyon (ATSS) sayısı Denklem 5.2 ile hesaplanmaktadır. Makine sayısı 5 olan bir TSDP için *IncDec* stratejisine göre oluşan ATSS değeri Denklem 5.2'ye göre $P(5-1, 2) = P(4, 2) = 12$ 'dir. P , burada permütasyon işlemini ifade etmektedir. Algoritma 5'te *IncDec* stratejisinin uygulama adımları verilmiştir.

$$ATSS = P(K - 1, 2) \quad (5.2)$$

Tüm aday çözümler bu şekilde değerlendirildikten sonra, en iyi üretim oranı değerini üreten tampon stok konfigürasyonu, bir sonraki yineleme için yeni çözüm olarak seçilmektedir.

Algoritma 5: IncDec Stratejisi

girdi : N^{kar} , $Th(N^{kar})$, $IncDec$ değeri (id), K
çıktı : Yerel optimum tampon stok konfigürasyonu; N^{ver}

- 1 $N^{kar} \leftarrow (N^{kar}_1, N^{kar}_2, \dots, N^{kar}_{K-1})$ ayarla
- 2 $N^{IncDec} \leftarrow (N^{IncDec}_1, N^{IncDec}_2, \dots, N^{IncDec}_{K-1})$ ayarla
- 3 $IncDec$ işlemi uygulanacak tampon stok çifti sayısını ($P(K-1, 2) \leftarrow ATSS$) hesapla
- 4 $IncDec$ işlemi uygulanacak tampon stok çiftlerini belirle; $i \in N^{kar}_i, j \in N^{kar}_j; i \neq j$
- 5 Problem boyutuna göre id değerini ayarla
- 6 $ATSS$ tampon stok çifti indisi l olsun ve $l \leftarrow 1$ ayarla
- 7 $N^{ver} \leftarrow N^{kar}$ olarak ayarla
- 8 **for** ($l \leftarrow 1$ to $ATSS$) **do**
- 9 $N^{IncDec} \leftarrow N^{kar}$
- 10 **for** ($i \leftarrow 1$ to $K-1$; $j \leftarrow 2$ to $K-1$) **do**
- 11 $N^{IncDec}_i \leftarrow N^{IncDec}_i + id$
- 12 $N^{IncDec}_j \leftarrow N^{IncDec}_j - id$
- 13 **if** $Th(N^{IncDec}) > Th(N^{kar})$ **then**
- 14 $N^{kar} \leftarrow N^{IncDec}$
- 15 **end**
- 16 **end**
- 17 **for** ($j \leftarrow 1$ to $K-1$; $i \leftarrow 2$ to $K-1$) **do**
- 18 $N^{IncDec}_i \leftarrow N^{IncDec}_i + id$
- 19 $N^{IncDec}_j \leftarrow N^{IncDec}_j - id$
- 20 **if** $Th(N^{IncDec}) > Th(N^{kar})$ **then**
- 21 $N^{kar} \leftarrow N^{IncDec}$
- 22 **end**
- 23 **end**
- 24 **end**
- 25 $N^{ver} \leftarrow N^{kar}$
- 26 id değerini güncelle
- 27 $l \leftarrow l + 1$

5.3.3 Kabul ve Durdurma Kriteri

Önerilen DKA algoritmasında elde edilen yeni çözümler amaç fonksiyonunda iyileşme sağladığı takdirde kabul edilir. Algoritma izin verilen maksimum yineleme sayısına ulaştıktan sonra sonlandırılmaktadır.

Önerilen DKA algoritması Algoritma 6’da verilmiştir. Başlangıç çözümü Bölüm 5.2’de ayrıntıları verilmiş olan başlangıç sezgisellerinden biri kullanılarak oluşturulur. Daha sonra mevcut çözüme, algoritmanın karıştırma aşamasında rulet tekeri yöntemiyle belirlenen bir komşuluk yapısı uygulanır. Ardından ilgili komşuluk yapısı kullanılarak elde edilen çözüme IncDec stratejisi uygulanarak yerel arama aşaması tamamlanır. Yerel arama sonucunda mevcut çözümden daha iyi bir çözüm elde edilirse, mevcut komşuluk yapısının seçim olasılığı artırılır ve mevcut en iyi çözüm güncellenir. Maksimum üretim oranı değerini veren en iyi tampon stok konfigürasyonu bir sonraki yineleme için mevcut çözüm haline gelir. Bu şekilde arama prosedürü, sonlandırma kriterine ulaşılan kadar devam eder.

Algoritma 6: Üretim Oranı Maksimizasyonu İçin Önerilen DKA Algoritması

girdi : KY, IncDec değeri, K

çıkıtı : Maksimum üretim oranına sahip tampon stok konfigürasyonu; N^{en}

- 1 Bir başlangıç sezgiseli kullanılarak başlangıç çözümünü oluştur; N^{bas}
 - 2 Komşuluk yapıları kümesini seç
 - 3 Her komşuluk yapısının seçim olasılığını eşitle
 - 4 Yineleme i olsun ve $i \leftarrow 1$ ayarla
 - 5 Problem boyutuna göre IncDec değerini ayarla
 - 6 $N^{en} \leftarrow N \leftarrow N^{bas}$ olarak ayarla
 - 7 Maksimum izin verilen iterasyon sayısını K olarak ayarla
 - 8 **tekrar et**
 - 9 | Rulet tekeri yöntemine göre bir KY seç,
 - 10 | N^i ’ye seçilmiş KY’yi uygula ve N^{kar} elde et % Karıştırma %
 - 11 | N^{kar} ’a IncDec stratejisini uygula ve N^{veni} elde et % Yerel Arama %
 - 12 | **if** $f(N^{veni}) > f(N)$ **then**
 - 13 | | $N \leftarrow N^{veni}$
 - 14 | | **if** $f(N) > f(N^{en})$ **then**
 - 15 | | | $N^{en} \leftarrow N$
 - 16 | | Mevcut KY’nin seçim olasılığını artır
 - 17 | IncDec değerini güncelle
 - 18 | $i \leftarrow i + 1$
 - 19 durdurma kriteri sağlanana kadar
-

5.4 Deneysel Çalışmalar

Önerilen DKA algoritmasının performansı karşılaştırma problemleri üzerinde test edilmiştir. Algoritma MATLAB (R2016a)’da kodlanmıştır. Tüm deneyler, 8 GB RAM’li 3.40 GHz Intel (R) i7 işlemciye sahip bir bilgisayarda gerçekleştirilmiştir. Karşılaştırma problemlerinin ayrıntıları ve önerilen DKA algoritmasının performans

analizi aşağıdaki alt bölümlerde sunulmuştur. Bu bölümdeki tablolarda makinelerin güvenilirlik parametreleri (ortalama tamir süresi ($MTTR = 1/r_i$) ve bozulmalar arasında geçen ortalama süre ($MTBF = 1/\beta_i$)) verilmiştir. Algoritma için belirlenen maksimum yineleme sayısı yapılan ön testlerden sonra her bir problem için makine sayısı (K) olarak belirlenmiştir. Farklı başlangıç sezgisellerine sahip önerilen DKA algoritması 10 kez çalıştırılmıştır ve 10 tekrar sonucunda elde edilen ortalama değerler üzerinde bir karşılaştırma yapılmıştır. Karşılaştırmada kullanılan literatürdeki yöntemler için sadece ilgili makalelerde raporlanan sonuçlar açısından değerlendirme yapılmıştır. Ortalama üretim oranı, optimum çözüme ulaşmak için gereken ortalama değerlendirme sayısı ve CPU süresi verileri ile literatürde yayımlanmış önceki çalışmalar ve önerilen algoritmanın performansı karşılaştırılmıştır.

Deneysel çalışmalar iki alt bölüm halinde sunulmuştur. İlk bölümde üretim oranı maksimizasyonu için önerilen DKA algoritmasının performansı test edilmiştir. İkinci bölümde ise TSDP'yi toplam stok minimizasyonu amacı ile çözmek üzere önerilen DKA algoritması Demir ve diğ. (2011) tarafından önerilen bir ikili arama algoritması ile birleştirilmiştir ve önerilen bu algoritmanın etkinliği de yine mevcut karşılaştırma problemleri üzerinde test edilmiştir.

5.4.1 Üretim Oranı Maksimizasyonu

5 Makineli Hat: Önerilen algoritmanın performansı ilk olarak 5 makineli ve tampon stok miktarının 31 olarak verildiği küçük bir örnek üzerinde test edilmiştir. Problem için tampon stok alanlarının alt sınırı 4 olarak verilmiştir. Bu problem ilk olarak Ho ve diğ. (1979)'nin çalışmasında önerilmiştir. Daha sonra Gershwin ve Schor (2000) ve Demir ve diğ. (2011) tarafından da kullanılmıştır. Ho ve diğ. (1979) ile Gershwin ve Schor (2000) problemin çözümü için gradyan temelli sezgisel algoritmalar önerirken Demir ve diğ. (2011) bir tabu arama algoritması önermişlerdir. Problemdaki makinelerin güvenilirlik özellikleri Tablo 5.1'de verilmiştir.

Tablo 5.1: 5 Makineli hattın güvenilirlik parametreleri (Ho ve diğ. 1979)

Makine (i)	1	2	3	4	5
$1/r_i$	11	19	12	7	7
$1/\beta_i$	20	167	22	22	26

Problemin optimum çözümünde $\{7, 10, 10, 4\}$ tampon stok konfigürasyonu ile elde edilen üretim oranı 0,4943'tür. Önerilen DKA algoritması da daha önceki çalışmalarda olduğu gibi optimum çözüme ulaşmıştır. Algoritma rastgele, PaVi ve MEKF başlangıç sezgiselleri için sırasıyla ortalama 22,1, 16,4 ve 11,1 değerlendirmeden sonra optimum çözüme ulaşmıştır. Optimum çözümü elde etmek için gereken ortalama CPU süresi ise tüm başlangıç sezgiselleri için 0,3 saniyeden daha azdır.

9 Makineli Hat: Bu problemde, farklı güvenilirlik parametrelerine sahip 9 makineli bir seri üretim hattı dikkate alınmıştır (Tablo 5.2). Bu problem seti ilk olarak Shi ve Men (2003) tarafından önerilmiş ve daha sonra Demir ve diğ. (2011) tarafından kullanılmıştır. Tüm senaryolar için izin verilen maksimum toplam tampon stok sayısı 160 olarak belirlenmiştir. Shi ve Men (2003) bu problemi çözmek için NP/TS adını verdikleri TA'ya dayalı melez bir algoritma, Demir ve diğ. (2011) ise DTL-TS olarak adlandırdıkları başka bir TA algoritması kullanmışlardır.

Önerilen DKA algoritması, Tablo 5.2'de de görüldüğü üzere TA tabanlı yöntemlerden daha iyi üretim oranı değerleri üretmiştir. DKA-Rastgele'nin en iyi çözüme ulaşmak için gereken ortalama değerlendirme sayısı 227,2 ve 354,5 aralığında iken, DKA-PaVi için 210,7 ile 288,5 arasında ve DKA-MEKF için 140,4 ile 190,0 arasında bulunmuştur. Algoritmanın sonlandırma kriteri sabit yineleme sayısı olduğu için, tüm başlatma sezgiselleri için ortalama CPU süresinin yaklaşık 2,6 saniye olduğu gözlenmiştir.

Tablo 5.2: 9 makineli hattın karşılaştırmalı sonuçları

Senaryo	Güvenilirlik Parametreleri		Üretim Oranı				
	β_i	r_i	NP/TS	DTL-TS	DKA-Rastgele	DKA-PaVi	DKA-MEKF
1	0,3	0,05	0,108143	0,108147	0,108243	0,108243	0,108243
2	0,3	0,10	0,200250	0,200255	0,200357	0,200357	0,200357
3	0,3	0,20	0,345491	0,345495	0,345580	0,345580	0,345580
4	0,3	0,30	0,452074	0,452077	0,452151	0,452151	0,452151
5	0,3	0,40	0,532002	0,532006	0,532091	0,532091	0,532091
6	0,4	0,05	0,088777	0,088782	0,088857	0,088857	0,088857
7	0,4	0,10	0,166232	0,166236	0,166322	0,166322	0,166322
8	0,4	0,20	0,293041	0,293046	0,293199	0,293199	0,293199
9	0,4	0,30	0,390814	0,390819	0,390881	0,390881	0,390881

Büyük boyutlu problemler: Bu örnekte 20 ve 40 makineli üretim hatları ele alınmıştır. Li ve diğ. (2015) tarafından önerilen bu problem setinin çözümü için yazarlar tarafından FA olarak adlandırılan bir algoritma geliştirilmiştir. 20 makineli hat için makinelere ait güvenilirlik parametreleri Tablo 5.3'te verilmiştir. 40 makineli hat ise özellikleri Tablo 5.3'te verilen bu 20 makineli hattın tekrar edilmesiyle oluşturulmuştur. Toplam tampon stok sayısı için kısıt 20 makineli hat için 400, 40 makineli hat için 400, 800 ve 1600 olarak belirlenmiş ve bu 4 farklı senaryo için problem çözülmüştür. Karşılaştırmalı sonuçlar Tablo 5.4'te sunulmuştur.

Tablo 5.3: 20 makineli hattın güvenilirlik parametreleri (Li ve diğ. 2015)

Makine (<i>i</i>)	1	2	3	4	5	6	7	8	9	10
β_i	0,011	0,005	0,022	0,025	0,052	0,021	0,017	0,013	0,011	0,016
r_i	0,039	0,016	0,166	0,085	0,186	0,105	0,116	0,136	0,043	0,139
Makine (<i>i</i>)	11	12	13	14	15	16	17	18	19	20
β_i	0,019	0,014	0,013	0,025	0,014	0,001	0,006	0,013	0,059	0,038
r_i	0,074	0,099	0,111	0,178	0,094	0,015	0,061	0,046	0,184	0,134

Tablo 5.4'ten görülebileceği gibi, önerilen DKA algoritması Li ve diğ. (2015) tarafından elde edilen en iyi çözümden daha iyi sonuçlar vermiştir. Önceki problemlere kıyasla önerilen DKA algoritmasının literatürdeki diğer yöntemlere göre üstünlüğünün büyük boyutlu örnekler için daha belirgin olduğu görülmüştür. Önerilen DKA-MEKF algoritması, ele alınan her bir problem seti için FA'dan sırasıyla %2,39, %1,52, %4,09 ve %3,97 daha iyi üretim oranı değerleri üretmiştir. Dahası, önerilen MEKF başlangıç sezgisel yöntemi, yalnızca çözüm kalitesi açısından değil, aynı zamanda hızlı yakınsama açısından da diğer başlangıç sezgisellerinden daha iyi bir performans göstermiştir. DKA-Rastgele'de 1. problem seti için gereken ortalama CPU süresi 69 iken, DKA-PaVi için 62 ve DKA-MEKF için 38 olarak kaydedilmiştir. 40 makineli problem setleri içinse, DKA-Rastgele'de gereken ortalama CPU süresi 532 ile 1311 aralığında iken, DKA-PaVi için 478 ile 986 arasında ve DKA-MEKF için 349 ile 840 arasında değişmiştir. Bununla birlikte, FA incelenen tüm problem setleri için birkaç saniye içinde en iyi çözümlere ulaşmıştır. Çözüm sürelerindeki farklılığın en önemli nedeni, kullanılan ayrıştırma yöntemidir. Li ve diğ. (2015), üretim hattının performansını belirlemek üzere değerlendirme sayısını önemli ölçüde azaltan 3 makineli-2 tampon stoklu bir ayrıştırma yöntemi önermişlerdir. Bu da TSDP'nin çözüm süresini önemli ölçüde azaltmıştır. Öte yandan, önerilen DKA algoritmasının

yerel arama adımımda IncDec stratejisinin uygulanması ile mevcut çözümün tüm komşu çözümleri değerlendirilmekte, bu da önerilen algoritmanın çözüm süresini arttırmaktadır.

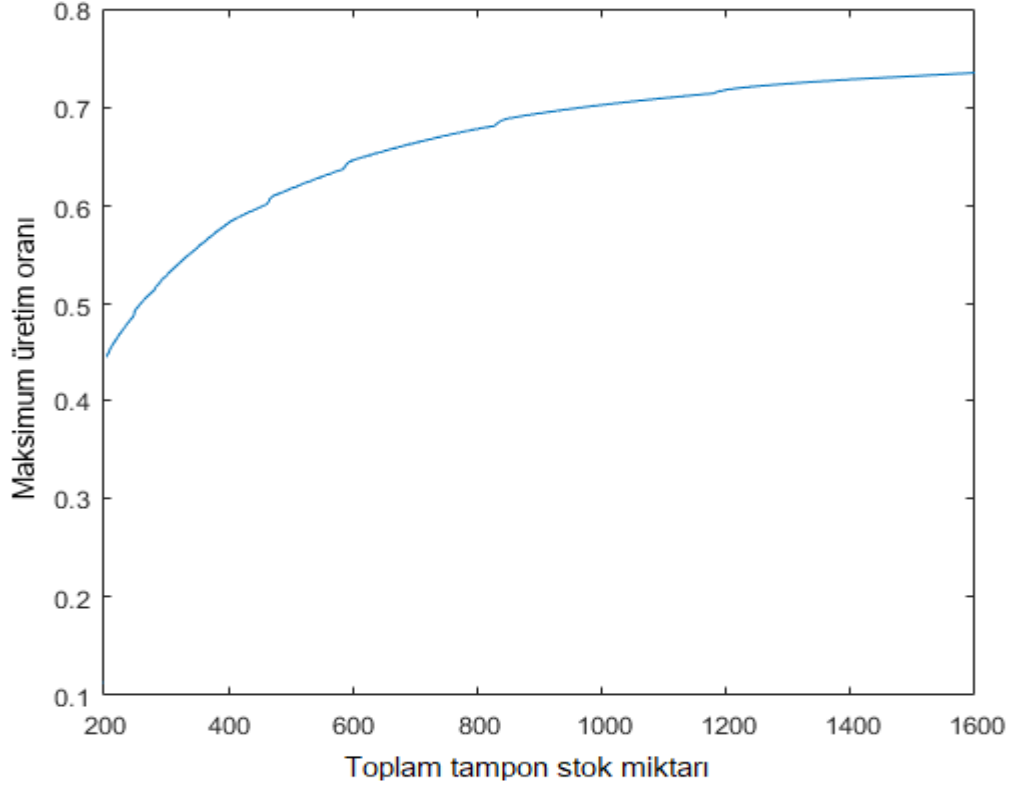
TSDP'nin bir tasarım problemi olduğu ve tasarım problemlerinin uzun vadeli stratejik problemler sınıfında olduğu göz önünde bulundurulduğunda, problemin çözüm süresinden makul bir sürede kaliteli çözümler elde etmeye odaklanması gerektiği unutulmamalıdır. Talbi (2009) tarafından da belirtildiği gibi tasarım problemleri için çözüm kalitesi çözüm süresinden daha önemlidir. Bu bağlamda, önerilen DKA algoritmasının makul süreler içinde FA'dan daha iyi çözümler ürettiği söylenebilir.

Son olarak, Şekil 5.5'te önerilen DKA-MEKF algoritmasıyla 40 makineli hat için toplam tampon stok miktarının farklı değerleri ile elde edilen maksimum üretim oranı grafiği verilmiştir. Bu grafik, bir üretim hattı tasarımcısının istenen üretim oranı değerini sağlayacak toplam tampon stok miktarını belirlemesine yardımcı olabilecek bir grafikdir. Toplam tampon stok miktarının artmasıyla, hattın üretim oranı da monoton olarak artmakta ve ortaya çıkan grafik, Şekil 5.5'te de görülebileceği gibi içbükey hale gelmektedir.

Tablo 5.4: 20 ve 40 makineli hatların karşılaştırmalı sonuçları

Problem seti	(K, N)	FA (Li ve diğ. (2015))		DKA-Rastgele			DKA-PaVi			DKA-MEKF		
		En İyi Üretim Oranı	Ort. Değ. Say.	Ort. Üretim Oranı	Ort. Değ. Say.	Ort. İyileşme (%) **	Ort. Üretim Oranı	Ort. Değ. Say.	Ort. İyileşme (%) **	Ort. Üretim Oranı	Ort. Değ. Say.	Ort. İyileşme (%) **
1	(20, 400)	0,662264	100	0,666298	3356	0,61	0,674265	3137	1,81	0,678092	2757	2,39
2	(40, 400)	0,571864	-	0,574597	9769	0,48	0,579772	8225	1,38	0,580575	7961	1,52
3	(40, 800)	0,651023	-	0,664865	12774	2,13	0,673412	11323	3,44	0,677652	10568	4,09
4	(40, 1600)	0,707086	-	0,718958	14819	1,68	0,729187	13601	3,13	0,735147	12084	3,97

**FA'ya göre ortalama iyileşme (%) = $100 \times [Th(DKA) - Th(FA)] / Th(FA)$



Şekil 5.5: 40 makineli hat için önerilen DKA-MEKF'nin performans değerlendirmesi

Üretim oranının maksimizasyonu için yapılmış tüm deneylerin sonuçları Tablo 5.5'te özetlenmiştir. Tabloda, her bir problem seti için 10 tekrar sonucunda elde edilen üretim oranı değerlerindeki standart sapmalar ve algoritmanın karıştırma aşamasında mevcut çözümü iyileştiren komşuluk yapılarının oranı sunulmaktadır. Tablo 5.5'te görüldüğü gibi, tekrarlar arasındaki sapmalar çok küçüktür, yani önerilen algoritma her çalışmada hemen hemen aynı üretim oranı değerlerini üretmektedir. Standart sapma değerleri aynı zamanda önerilen DKA algoritmasının sağlamlığını (*robustness*) da göstermektedir. Genel olarak, uygulanan komşuluk yapılarının iyileştirme oranları dikkate alındığında, sağ ve sol rotasyon komşuluk yapıları ortalama %28 ile aynı orana sahiptir. Bununla birlikte, bu oran ele alınan problem setine bağlı olarak değişmektedir. Öte yandan, tüm deneyler göz önüne alındığında, ters çevirme ve yer değiştirme komşuluk yapıları mevcut çözümü sırasıyla ortalama %16 ve %24 oranında iyileştirmektedir. Bu deneylerden de görüldüğü üzere, sağ ve sol rotasyon komşuluk yapılarını uygulamak TSDP'yi çözmeye ters çevirme ve yer değiştirme komşuluk yapılarını kullanmaktan daha etkilidir.

Tablo 5.5: Üretim oranı maksimizasyonu için deneysel çalışmaların özeti

Problem seti	Başlangıç sezgiseli	Standart sapma	Komşuluk yapılarının başarıyla uygulama oranları (%)			
			Sağ rotasyon	Sol rotasyon	Ters çevirme	Yer değiştirme
(5, 31)	Random	0,0000	34	16	16	34
	PaVi	0,0000	50	50	0	0
	MEKF	0,0000	0	0	0	0
(9, 160)	Random	0,0000	18	27	31	24
	PaVi	0,0000	22	42	17	19
	MEKF	0,0000	30	30	16	24
(20, 400)	Random	0,0073	29	28	20	23
	PaVi	0,0032	26	18	18	38
	MEKF	0,0027	24	30	14	32
(40, 400)	Random	0,0085	36	14	14	36
	PaVi	0,0074	28	28	17	27
	MEKF	0,0071	26	35	18	21
(40, 800)	Random	0,0042	26	25	15	34
	PaVi	0,0076	19	43	15	23
	MEKF	0,0108	32	31	20	17
(40, 1600)	Random	0,0090	25	31	19	25
	PaVi	0,0008	31	21	21	27
	MEKF	0,0021	40	26	10	24

Şimdiye kadar sunulan deneysel sonuçlar, önerilen DKA-MEKF algoritmasının literatürde bildirilen diğer yöntemlerle aynı veya daha iyi sonuçlar ürettiğini göstermiştir. Yöntemlerle elde edilen ortalama üretim oranı değerleri arasındaki farklar küçük görüldüğünden, ortalama üretim oranı değerleri arasındaki farkın istatistiksel olarak anlamlı olup olmadığını belirlemek üzere Wilcoxon işaretli sıra testi (*Wilcoxon signed-rank test*) kullanılmış ve aşağıdaki hipotez $\alpha=0,05$ anlamlılık düzeyinde test edilmiştir:

$$H_0: Th(\text{Literatürdeki En İyi}) - Th(\text{DKA-MEKF}) = 0$$

Tablo 5.6: Wilcoxon işaretli-sıra testi

H_0 hipotezi	Test	Anlamlılık (p)	Karar
Literatürdeki en iyi sonuç ve DKA-MEKF arasındaki farkların medyanı 0'a eşittir.	İlişkili örneklem Wilcoxon işaretli-sıra testi	0,001	H_0 hipotezi reddedilir.

Tablo 5.6'da Wilcoxon işaretli sıra testinin sonucu görülmektedir. Wilcoxon işaretli sıra testi ele alınan bütün karşılaştırma problemleri için üretim oranı değerleri arasındaki farkın %95 güven seviyesinde istatistiksel olarak anlamlı olduğunu ortaya koymuştur. Wilcoxon testinin sonucu da dikkate alındığında önerilen DKA-MEKF algoritmasının ele alınan problemler için literatürdeki diğer yöntemlerden daha iyi üretim oranı değeri ürettiğini söylemek mümkündür. Ayrıca önerilen algoritmanın

etkinliđi, makinelerin güvenilirlik parametrelerinin birbirinden farklı olduđu büyük boyutlu hatlarda öne çıktıđı görölmektedir. Öte yandan, algoritmanın üstünlüđü, bu bölümde sunulan deneysel sonuçlardan da görüldüđu gibi, büyük N/K oranına sahip hatlar için daha belirgindir. Özetle, modifiye edilmiş ters kâse fenomeni sezgisel yöntemiyle başlayan DKA algoritmasının, TSDP'nin çözümünde etkin bir algoritma olduđunu söylemek mümkündür.

5.4.2 Toplam Tampon Stok Miktarı Minimizasyonu

Bu bölümde TSDP literatüründe primal problem olarak bilinen ve Bölüm 2.5.2'de tanımlanmış olan toplam tampon stok minimizasyonu problemi ele alınmış ve problemin çözümü için önerilen DKA algoritması, Demir ve diğ. (2011) tarafından önerilen bir ikili arama (*binary search*) algoritmasıyla birlikte kullanılmıştır. Önerilen ikili-DKA algoritması, önceden belirlenmiş olan bir hedef üretim oranı değeri elde edilene kadar bir alt sınır (AS) ve bir üst sınır (ÜS) arasında Algoritma 7'de gösterildiđi gibi arama yapmaktadır. Algoritmanın her adımında elde edilen toplam tampon stok miktarı (N) için DKA algoritması çalıştırılmakta ve mevcut N değeri ile elde edilebilecek maksimum üretim oranı elde edilmeye çalışılmaktadır. Algoritma hedeflenen üretim oranını sağlayacak minimum toplam tampon stok değerine ulaşıncaya sonlandırılmaktadır.

Algoritma 7: Toplam Tampon Stok Minimizasyonu İçin Önerilen İkili-DKA Algoritması

girdi : KY, IncDec değeri, $AS = 0$, $ÜS = M$, hedef üretim oranı değeri (Th^*)

çıkıtı : Minimum toplam tampon stok miktarına sahip en iyi tampon stok konfigürasyonu B^{en}

```

1  Toplam tampon stok miktarı  $N$ 'i  $N \leftarrow ((AS + ÜS)/2)$  olarak ayarla
2  while ( $ÜS \neq AS$ ) do
3      Üretim oranı maksimizasyonu için DKA algoritmasını çalıştır
4      if  $Th(B^{en}) < Th^*$  then
5           $AS \leftarrow N$ 
6           $N \leftarrow ((AS + ÜS)/2)$ 
7      else
8           $ÜS \leftarrow N$ 
9           $N \leftarrow ((AS + ÜS)/2)$ 
10 end

```

Önerilen İkili-DKA algoritmasının performansı da yine literatürde mevcut karşılaştırma problemleri üzerinde test edilmiştir. İlerleyen bölümlerde bu deneysel çalışmaların sonuçlarına yer verilmiştir.

12 Makineli Hat: Bu karşılaştırma problemi ilk olarak Park (1993) tarafından önerilmiştir. 12 makineli hat için güvenilirlik parametreleri Tablo 5.7’de verilmiştir. Bu problemi çözmek için, Park (1993) tarafından iki aşamalı bir sezgisel prosedür, Gershwin ve Schor (2000) tarafından gradyan tabanlı bir arama algoritması ve son olarak Demir ve diğ. (2011) tarafından TA tabanlı bir algoritma önerilmiştir. Hedef üretim oranı değerleri, Senaryo 1 için 0,85 ve Senaryo 2 için 0,895 olarak belirlenmiştir.

Tablo 5.7: 12 makineli hat (Park 1993)

Makine (<i>i</i>)	1	2	3	4	5	6	7	8	9	10	11	12
β_i	0,037	0,015	0,02	0,03	0,03	0,01	0,02	0,02	0,02	0,03	0,03	0,01
r_i	0,35	0,15	0,4	0,4	0,3	0,2	0,3	0,3	0,4	0,4	0,3	0,25

Tablo 5.8’de 12 makineli hat için gerçekleştirilen deneysel çalışmanın sonuçları özetlenmektedir. Senaryo-1’in sonuçlarına bakıldığında önerilen İkili-DKA algoritması 87 toplam tampon stok miktarı ile literatürdeki önceki yöntemlerle aynı sonuca ulaşmaktadır. Ancak aynı toplam tampon stok miktarı ile elde edilen üretim oranı değeri diğer yöntemlere göre az da olsa daha fazladır. Senaryo-2’ye bakıldığında ise önerilen İkili-DKA algoritması rastgele bir çözümle başladığında literatürdeki mevcut çözümlere ulaşırken, önerilen İkili-DKA-PaVi ve İkili-DKA-MEKF algoritmaları, daha az toplam tampon stok miktarı ile hedef üretim oranı değerine ulaşmıştır. Bu örnekten de görüldüğü üzere iyi bir başlangıç çözümü ile aramaya başlamak daha iyi bir çözüme ulaşılmasını sağlamıştır. Bu problemin sonuçlarından yola çıkarak önerilen İkili-DKA algoritmasının toplam tampon stok miktarı minimizasyonu için etkin bir algoritma olduğunu söylemek mümkündür.

Tablo 5.8: 12 makineli hattın karşılaştırmalı sonuçları

Senaryo	Hedef Üretim Oranı Değeri	Park (1993)		Gershwin ve Schor (2000)		Demir ve diğ. (2011)		İkili-DKA-Rastgele		İkili-DKA-PaVi		İkili-DKA-MEKF	
		Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>
1	0,850	0,8505	93	0,8507	87	0,8507	87	0,8508	87	0,8508	87	0,8508	87
2	0,895	0,8950	390	0,8950	243	0,8950	242	0,8951	242	0,8950	240	0,8950	240

Büyük boyutlu üretim hattı: Bu problem setinde önerilen İkili-DKA algoritmasının etkinliğini test etmek üzere daha fazla makinenin olduğu üretim hatları ele alınmıştır. 10, 20 ve 30 makineli hatları içeren bu problem seti ilk olarak Gershwin ve Goldis (1995) tarafından önerilmiş ve daha sonra Gershwin ve Schor (2000) tarafından kullanılmıştır. İncelenen 20 ve 30 makineli hatlar aynı 10 makineli hattın bloklar halinde çoğaltılmasıyla oluşturulmuştur. 10 makineli bir bloğun güvenilirlik parametreleri Tablo 5.9’da verilmiştir. Gershwin ve Goldis (1995) problemin çözümü için birleşik bir algoritma (GG) kullanırken, Gershwin ve Schor (2000) bu problemi çözmek için gradyan tabanlı bir arama yöntemi önermiştir. Hedef üretim oranı değeri, tüm hatlar için 0,88 olarak belirlenmiştir.

Tablo 5.9: 10 makineli blok hat (Gershwin ve Goldis 1995)

Makine (<i>i</i>)	1	2	3	4	5	6	7	8	9	10
β_i	0,007	0,008	0,003	0,004	0,006	0,007	0,008	0,003	0,004	0,006
r_i	0,094	0,095	0,045	0,078	0,069	0,094	0,095	0,045	0,078	0,069

Deneyin sonuçları Tablo 5.10’da sunulmuştur. Tablo 5.10’da da görülebileceği gibi önerilen İkili-DKA algoritması ile 10 makineli hat için literatürdeki mevcut çözümlere ulaşılırken 20 ve 30 makineli hatlar için daha iyi çözümler elde edilmiştir. Ayrıca önerilen DKA algoritması MEKF sezgiseli ile başlatıldığında daha az toplam tampon stok miktarı ile daha yüksek üretim oranı değerlerine ulaşmıştır.

Tablo 5.10: Büyük boyutlu üretim hattının karşılaştırmalı sonuçları

Makine Sayısı	Gershwin ve Goldis (1995)		Gershwin ve Schor (2000)		İkili-DKA-Rastgele		İkili-DKA-PaVi		İkili-DKA-MEKF	
	Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>	Üretim Oranı	<i>N</i>
10	0,8800	443	0,8800	433	0,8801	433	0,8801	433	0,8801	433
20	0,8800	995	0,8800	995	0,8801	995	0,8801	994	0,8800	993
30	0,8800	1557	0,8800	1556	0,8801	1551	0,8802	1550	0,8804	1548

Bu bölümde sunulan deneysel sonuçlar, önerilen DKA algoritmasının daha az toplam tampon stok miktarı ile daha yüksek üretim oranı değerleri verdiğini göstermektedir. Bölüm 5.4.1’deki deneylere benzer şekilde, MEKF başlangıç sezgiseli yöntemiyle başlayan İkili-DKA algoritması en iyi sonuçları vermektedir. Önerilen İkili-DKA-MEKF algoritmasının üstünlüğü, özellikle Tablo 5.10’da görülebileceği gibi büyük hatlar için daha belirgindir. Bu nedenle, önerilen İkili-DKA-MEKF

algoritmasının bu bölümde ele alınan tüm problem setleri için TSDP'yi toplam tampon stok miktarı minimizasyonu amacı ile çözmeye en iyi yöntem olduğu söylenebilir.

5.5 Bulgular ve Tartışma

Üretim oranı maksimizasyonu ve toplam tampon stok miktarı minimizasyonu problemlerinin çözümü için yapılan tüm deneylerin sonuçları göz önünde bulundurulduğunda önerilen DKA algoritmasının, dikkate alınan tüm karşılaştırma problemleri için literatürdeki diğer yöntemlerle aynı sonucu verdiği veya literatürdeki en iyi çözümden daha iyi sonuçlar ürettiği görülmüştür. Ayrıca başlangıç sezgiseli olarak MEKF yöntemini kullanmak, özellikle büyük üretim hatları için en iyi çözüme ulaşmada gereken yineleme sayısını azaltmaktadır. Deneysel çalışmalarda önerilen PaVi başlangıç sezgiselinin darboğaz makineleri dikkate aldığı halde, hattın başına ve sonuna daha fazla tampon stok atadığı ve bu nedenle, PaVi başlangıç sezgiselinin beklendiği kadar etkin olmadığı görülmüştür. MEKF başlangıç sezgiseli tampon stokları PaVi'ye kıyasla daha eşit oranda dağıttığı için ele alınan problemlerde başlangıç sezgiseli olarak PaVi'den daha etkin olduğu gözlenmiştir.

Bu çalışmada, önerilen DKA algoritmasının etkinliğini kanıtlamak için hipotetik üretim hatları dikkate alınmıştır. Bununla birlikte, önerilen algoritma gerçek hayattaki üretim problemlerini çözmek için büyük bir potansiyele sahiptir. Bu bağlamda, önerilen algoritmanın en büyük dezavantajı, en iyi çözümü elde etmek için harcadığı CPU süresidir. Önerilen DKA algoritmasının her yinelemede birden fazla çözümü değerlendirmesi algoritmanın çözüm süresini de arttırmaktadır. Bu dezavantajın üstesinden gelmek için önerilen algoritma, diğer arama algoritmaları ile melezleştirilebilir.

5.6 Sonuç

Bu bölümde güvenilir olmayan seri üretim hatlarında TSDP'yi çözmek için bir DKA algoritması önerilmiştir. Ayrıca iki farklı başlangıç sezgiseli önerilerek, bu sezgisellerin algoritmanın performansı üzerindeki etkisi test edilmiştir. Problem iki açıdan ele alınmıştır: üretim oranı maksimizasyonu ve toplam tampon stok miktarı

minimizasyonu. Önerilen DKA algoritmasının performansı literatürde mevcut karşılaştırma problemleri üzerinde test edilmiştir.

Deneysel çalışmalar önerilen DKA algoritmasının, dikkate alınan tüm karşılaştırma problemleri için literatürdeki diğer yöntemlerle aynı sonucu veya mevcut yöntemlerden daha iyi sonuçlar verdiğini göstermiştir. Ayrıca başlangıç sezgiseli olarak MEKF yöntemini kullanmanın özellikle büyük üretim hatları için en iyi çözüme ulaşmada gereken yineleme sayısını azalttığı gözlenmiştir. Elde edilen tüm sonuçlar değerlendirildiğinde, bir tasarım problemi olan TSDP için önerilen DKA-MEKF algoritmasının problemi makul bir hesaplama süresi içinde etkin bir şekilde çözdüğü görülmüştür. Sonuç olarak, önerilen DKA-MEKF algoritmasının TSDP'yi farklı amaçlarla çözebilen etkin bir algoritma olduğunu söylemek mümkündür.

6. BOZULMAYA MARUZ KALAN ÜRETİM HATLARINDA ÜRETİM ORANI MAKSİMİZASYONU İÇİN ADAPTİF BÜYÜK KOMŞULUK ARAMA ALGORİTMASI

6.1 Giriş

Bu tez çalışmasının ana amacı güvenilir olmayan üretim hatlarında TSDP'nin çözümü için etkin algoritmalar geliştirmektir. Bu amaçla problemin çözümü için önceki bölümde bir DKA algoritması önerilmişti. Bu bölümde ise komşuluk temelli başka bir meta-sezgisel algoritma olan ABKA algoritması önerilmiştir. ABKA, BKA algoritmasının birden fazla çıkarma ve ekleme operatörünü içeren versiyonudur. Literatürde farklı problemler için ABKA'nın birçok başarılı uygulaması vardır (Akpınar 2016, Ghilas ve diğ. 2016, Iris ve diğ. 2017, Chen ve diğ. 2018, Avcı ve Avcı 2019). Bu tez çalışmasında ilk defa bir üretim hattı tasarımı problemi olan TSDP'nin çözümüne yönelik olarak bir ABKA algoritması önerilmiştir.

Önceki bölümde olduğu gibi bu bölümde de önerilen algoritmanın arama eforunu azaltmak için başlangıç sezgiselleri önerilmiş ve bu başlangıç sezgisellerinin algoritmanın performansı üzerindeki etkisi test edilmiştir. Ayrıca problemin yapısına özgü olarak iki farklı çıkarma-ekleme operatörü geliştirilmiş ve algoritma boyunca adaptif bir şekilde uygulanmıştır. Önerilen ABKA algoritmasının performansı literatürdeki mevcut karşılaştırma problemleri üzerinde test edilmiştir.

Bölümün geri kalanı şu şekilde organize edilmiştir. Sonraki bölümde ABKA algoritmasında kullanılan başlangıç sezgiselleri tanıtılmıştır. Bölüm 6.3'te önerilen ekleme-çıkarma operatörleri, Bölüm 6.4'te ise algoritmada kullanılan adaptif mekanizma anlatılmıştır. Sonraki bölümde yerel arama için kullanılan strateji açıklanmıştır. Algoritma için kabul ve durdurma kriterleri anlatıldıktan sonra Bölüm 6.8'de deneysel çalışmalar detaylı bir şekilde sunulmuştur. Bölüm 6.9'da çalışmanın bulguları üzerinde durulmuş ve son olarak Bölüm 6.10'da da bu bölümde yapılan çalışmalar özetlenmiştir.

6.2 Önerilen Başlangıç Çözüm Sezgiselleri

Önceki bölümde olduğu gibi bu bölümde de başlangıç çözümü oluşturma yöntemlerinin önerilen çözüm algoritmasının performansı üzerindeki etkisi test edilmiştir. Bu amaçla rastgele oluşturulan bir çözümle algoritmayı başlatmanın yanı sıra üç ayrı başlangıç çözümü üretme sezgiselinin çözüm kalitesi ve optimum çözüme ulaşma zamanı üzerindeki etkisi gözlenmiştir. Bölüm 5.2’de önerilen PaVi ve MEKF sezgisellerinin yanı sıra bu bölümde yine Hillier ve diğ. (1993) tarafından geliştirilen ters kâse fenomenine dayalı yeni bir başlangıç çözümü oluşturma sezgiseli önerilmiştir. Düzleştirilmiş Kâse Fenomeni (DKF) olarak isimlendirilen bu sezgiselde MEKF ile oluşturulan tampon stok konfigürasyonu, tampon stokların eşit olarak dağıtıldığı başka bir tampon stok konfigürasyonu ile birleştirilmiştir.

DKF sezgiseli ile tampon stok konfigürasyonu oluşturmak için öncelikle MEKF sezgiseli ile bir tampon stok konfigürasyonu elde edilir. Daha sonra toplam tampon stok miktarının tampon stok alanlarına eşit/düzgün dağıldığı ikinci bir tampon stok konfigürasyonu oluşturulur. Elde edilen bu iki konfigürasyon aşağıdaki formül yardımı ile birleştirilir.

$$\left(\frac{MEKF + Düzgün}{2} \right) \quad (6.1)$$

Denklem 6.1’de “Düzyün” eşit tampon stok konfigürasyonunu ifade ederken, elde edilen konfigürasyondaki değerler en yakın tamsayıya yuvarlanmaktadır. Bu başlangıç sezgiselindeki ana fikir, daha önce literatürde başarıyla uygulanmış düzyün tampon stok konfigürasyonu dağılımının MEKF’e göre daha düz bir konfigürasyonunu elde etmek ve MEKF ile elde edilecek avantajlarını birleştirmektir. Düzyün tampon stok dağılımı elde edilirken toplam tampon stok sayısı $K-1$ ’e bölünmektedir ve değerler en yakın tamsayıya yuvarlanmaktadır. Denklem 6.1’deki işlemde sonra fark değerine göre hattın ortasına ekleme yapılmakta veya öncelikle hattın sonunda ve daha sonra hattın başında olmak üzere azaltma yapılmaktadır. Bu başlangıç sezgiseli ile başlayan ABKA algoritması, tezin geri kalanında ABKA-DKF olarak isimlendirilecektir.

DKF Sezgiseli Algoritma 8’de verilmiştir. 5 makineden ve toplamda 31 tampon stoktan oluşan bir üretim hattı için $N^{bas} = \{6, 10, 9, 6\}$ başlangıç tampon stok

konfigürasyonu elde edilmektedir. Örneğin; $K=7$, $N=54$ ve $K=9$, $N=160$ için DKF sezgiseli kullanılarak oluşturulan başlangıç tampon stok konfigürasyonları sırasıyla $\{6, 9, 12, 12, 9, 6\}$ ve $\{14, 18, 22, 26, 26, 22, 18, 14\}$ 'dir.

Algoritma 8: Düzleştirilmiş Kase Fenomeni Sezgiseli

girdi : makine sayısı (K), tampon stok miktarı (N), N^{mekf}
çıkıtı : başlangıç tampon stok konfigürasyonu (N^{bas} ; çözüm uygunsa *DOĞRU*, değilse *YANLIŞ*)

- 1 $N \leftarrow (0, 0, \dots, 0) \leftarrow (N_1, N_2, \dots, N_{K-1})$ ayarla
- 2 $N^{dkf} \leftarrow (0, 0, \dots, 0) \leftarrow (N^{dkf}_1, N^{dkf}_2, \dots, N^{dkf}_{K-1})$ ayarla
- 3 TOPLAM_tamponStok $\leftarrow 0$ ayarla
- 4 **for** ($i \leftarrow 1$ **to** $(K-1)$) **do**
- 5 $N_i \leftarrow (N/(K-1))$ **then**
- 6 **end**
- 7 **for** ($i \leftarrow 1$ **to** $(K-1)$) **do**
- 8 $N^{dkf}_i \leftarrow N_i$
- 9 **end**
- 10 **for** ($i \leftarrow 1$ **to** $(K-1)$) **do**
- 11 $N^{dkf}_i \leftarrow ((N^{dkf}_i + N^{mekf}_i)/2)$
- 12 **end**
- 13 TOPLAM_tamponStok $\leftarrow TOPLAM(N^{dkf}_i)$ ayarla
- 14 $FARK \leftarrow N - TOPLAM_tamponStok$
- 15 **if** $FARK < 0$ **then**
- 16 $HB \leftarrow 0$ (hattın başındaki tampon stok değerleri)
- 17 $HS \leftarrow 0$ (hattın sonundaki tampon stok değerleri)
- 18 **if** $N^{dkf}_{K-1} \neq 0$ **then**
- 19 $HB \leftarrow \lfloor (FARK)/2 \rfloor$
- 20 $HS \leftarrow \lfloor (FARK)/2 \rfloor$
- 21 **for** ($i \leftarrow K-1$ **to** $(K-1 + HS + 1)$) **do**
- 22 $N^{dkf}_i \leftarrow N^{dkf}_i - 1$
- 23 **end**
- 24 **for** ($i \leftarrow 1$ **to** $(-HB)$) **do**
- 25 $N^{dkf}_i \leftarrow N^{dkf}_i - 1$
- 26 **end**
- 27 **else if** $N^{dkf}_{K-1} = 0$ **then**
- 28 $0 \leftarrow sifirSayac$
- 29 **for** ($i \leftarrow 1$ **to** $((K-1)/2)$) **do**
- 30 **if** $N^{dkf}_i < 1$ **then**
- 31 $sifirSayac \leftarrow sifirSayac + 1$
- 32 **end**
- 33 **end**
- 34 **for** ($i \leftarrow (1 - sifirSayac)$ **to** $(1 - sifirSayac + FARK + 1)$) **do**
- 35 $N^{dkf}_i \leftarrow N^{dkf}_i - 1$
- 36 **end**
- 37 **end**
- 38 **else**
- 39 **for** ($i \leftarrow ((K-1)/2)$ **to** $((K-1)/2) + FARK - 1$) **do**
- 40 $N^{dkf}_i \leftarrow N^{dkf}_i + 1$
- 41 **end**
- 42 **end**
- 43 $N^{bas} \leftarrow N^{dkf}$
- 44 **döndür** *DOĞRU*

6.3 Çıkarma ve Ekleme Sezgiselleri

ABKA algoritmasının temelini çıkarma-ekleme operatörleri oluşturmaktadır. Literatürde farklı problemler için geliştirilmiş birçok çıkarma-ekleme operatörü olmakla beraber TSDP için uygun bir operatör çifti bulunmamaktadır. Bu nedenle, ABKA algoritmasının temel işleyişini koruyacak şekilde ve problemin yapısına uygun olarak iki yeni çıkarma-ekleme operatör çifti önerilmiştir. Bunlardan ilki tampon stokun konumuna ve değerine dayalı olarak geliştirilmiş olan ve konum tabanlı çıkarma-ekleme operatörü adı verilen sezgiseldir. İkincisi ise makinelerin çalışma/bozulma durumlarını dikkate alarak geliştirilen ve durum tabanlı çıkarma-ekleme operatörü olarak isimlendirilen sezgiseldir.

6.3.1 Konum Tabanlı Çıkarma-Ekleme Operatörü

Bu operatör çifti tampon stokların konumuna ve değerine göre uygulanmaktadır. Operatörü uygulamak için öncelikle tampon stok konfigürasyonu aşağıdaki denklemler sırasıyla (6.2 – 6.4) kullanılarak *Sol Kenar Elemanları* (N_{LEE}), *Sağ Kenar Elemanları* (N_{REE}) ve *Merkez Elemanlar* (N_{CE}) olarak isimlendirilen alt konumlara bölünür:

$$L = \left(\frac{K}{4}\right) \quad (6.2)$$

$$\begin{aligned} N_{LEE} &= [N_1, N_L] \\ R &= (L - K) \end{aligned} \quad (6.3)$$

$$N_{REE} = [N_R, N_{K-1}]$$

$$N_{CE} = [b_{L+1}, b_{R-1}] \quad (6.4)$$

Örneğin; $N = \{5, 9, 7, 10\}$ şeklindeki bir tampon stok konfigürasyonunun *Sol Kenar Elemanları* $N_{LEE} = \{5\}$, *Sağ Kenar Elemanları* $N_{LEE} = \{10\}$ ve *Merkez Elemanları* $N_{CE} = \{9, 7\}$ olacaktır.

Konuma ve değere dayalı operatör çifti işlemi yaparken temel motivasyon yine Ters Kâse Fenomeni olmuştur. Minimum ve maksimum tampon stok değerlerinin

bulunduğu konuma göre tampon stok değerlerinde çıkarma ve ekleme yapılmaktadır. Örneğin; maksimum tampon stok değeri konfigürasyonun başında iken minimum tampon stok değeri konfigürasyonun tam ortasında ise, ortadaki tampon stok değeri daha büyük olacak şekilde baştaki stok değeri azaltılıp ortadaki tampon stoğun değeri arttırılır. Bu işlemler Şekil 6.1’de belirtilen koşullar gözetilerek gerçekleştirilir. Şekil 6.1’de N mevcut çözümü gösterirken, N^{yeni} geçici yeni çözümü göstermektedir. Bu sezgisel yöntem, özellikle rastgele başlangıç çözümü ile algoritmanın başlatıldığı durumlarda eğer konfigürasyondaki değerler dalgalı bir yapıya sahipse tampon stok konfigürasyonlarının daha düzgün dağılmasını sağlar.

Koşul 1:
EĞER (($\min(N) \in N_{LEE}$) VEYA ($\min(N) \in N_{REE}$)) VE (($\max(N) \in N_{LEE}$) VEYA ($\max(N) \in N_{REE}$))
 $\min(N) = \lfloor (\min(N) + \max(N))/2 \rfloor$;
 $\max(N) = \lceil (\min(N) + \max(N))/2 \rceil$;
 $N^{yeni} = N$;

Koşul 2:
EĞER (($\min(N) \in N_{CE}$) VE (($\max(N) \in N_{LEE}$) VEYA ($\max(N) \in N_{REE}$)))
 $\min(N) = \lfloor (\min(N) + \max(N))/4 \rfloor$;
 $\max(N) = \lceil 3(\min(N) + \max(N))/4 \rceil$;
 $N^{yeni} = N$;

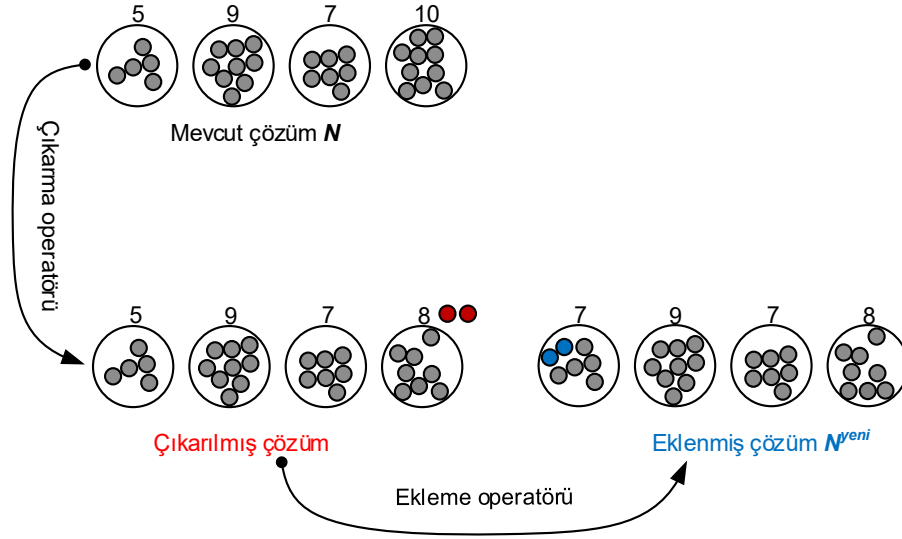
Koşul 3:
EĞER (($\min(N) \in N_{CE}$) VE ($\max(N) \in N_{CE}$))
 $\min(N) = \lfloor (\min(N) + \max(N))/2 \rfloor$;
 $\max(N) = \lceil (\min(N) + \max(N))/2 \rceil$;
 $N^{yeni} = N$;

Koşul 4:
EĞER (($\min(N) \in N_{LEE}$) VEYA ($\min(N) \in N_{REE}$)) VE ($\max(N) \in N_{CE}$)
 $\min(N) = \lfloor (\min(N) + \max(N))/4 \rfloor$;
 $\max(N) = \lceil 3(\min(N) + \max(N))/4 \rceil$;
 $N^{yeni} = N$;

Şekil 6.1: Önerilen konum tabanlı çıkarma-ekleme operatörü koşulları

Şekil 6.2’de yukarıda belirtilen koşulların 5 makineli bir hat için uygulaması resmedilmiştir. Toplam 31 tampon stok için mevcut çözümün $N=\{5, 9, 7, 10\}$ olduğu varsayılırsa konum tabanlı çıkarma-ekleme operatörünü uygulanırken öncelikle konfigürasyondaki elemanlar sol kenar, merkez ve sağ kenar elemanlar olarak sırasıyla $N_{LEE} = N_1$; $N_{CE} = [N_2, N_3]$; $N_{REE} = N_4$ şeklinde ayrılır. Daha sonra minimum ve maksimum tampon stok değerlerinin konumu belirlenir. Aynı minimum/maksimum değerden birden fazla varsa bunların biri rastgele seçilir. Örneğimizde minimum değer mevcut konfigürasyonun başında (sol kenar), maksimum değer ise sonunda (sağ kenar) bulunmaktadır. Dolayısıyla Şekil 6.1’deki birinci koşul sağlanmaktadır. 1. koşulda belirtilen çıkarma-ekleme işlemlerinin yapılması ile yeni tampon stok

konfigürasyonu $N^{yeni} = \{7, 9, 7, 8\}$ olarak elde edilir. Her yinelemede toplam tampon stok değeri kontrol edilerek toplam tampon stok değerinin sabit kalması sağlanır.



Şekil 6.2: Konum tabanlı çıkarma-ekleme operatörü: 5 makineli hat örneği

6.3.2 Durum Tabanlı Çıkarma-Ekleme Operatörü

Bu operatör çifti, hattaki makinelerin çalışma/bozulma durumunu dikkate alan bir sezgiseldir. Bir üretim hattında tüm tampon stok alanları sonsuz kapasiteye sahipse, maksimum üretim oranı Denklem 2.1 ve Denklem 2.3 ile hesaplanan hattaki makinelerden en düşük izolasyondaki etkinlik/kullanılabilirlik değerine sahip makinenin üretim oranı değerine eşittir. Bu nedenle, hattın üretim oranını mümkün olan en yüksek seviyeye çıkarmak için ortalama etkin servis oranı en küçük olan makine ($\min\{\rho_i\}, i = 1, 2, \dots, K$) asla açlık durumuna düşmemelidir. Bunun için de ilgili makinenin önüne daha fazla tampon stok atanması gerekir. Bu çıkarma-ekleme operatörünün amacı makinelerin açlık ve bloklanma olasılıklarını azaltacak şekilde tampon stokları makinelerin arasına dağıtmaktır.

Bu operatörü uygulamak için mevcut tampon stok konfigürasyonu içinde iki farklı tampon stok değeri (N_i, N_j) rastgele ($i, j; i \neq j, j > i$) seçilerek bu değerler Denklem 6.5'teki gibi toplanır ve toplam tampon stok değeri (N_{SUM}) elde edilir.

$$N_{SUM} = N_i + N_j \quad (6.5)$$

Benzer şekilde, i . ve j . tampon stokun önündeki $(i+1)$. ve $(j+1)$. makinenin ortalama etkin servis oranı Denklem 6.6'daki gibi (ρ_{SUM}) toplanır.

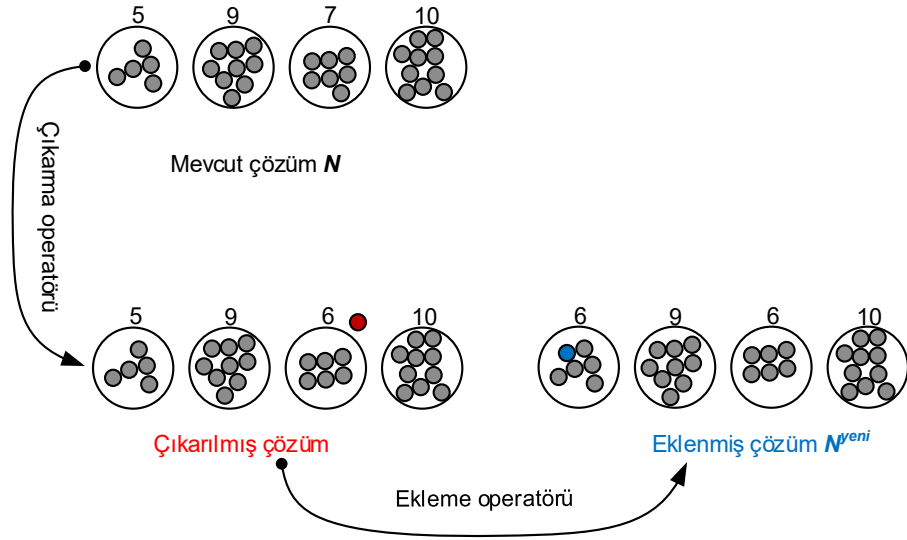
$$\rho_{SUM} = \rho_{(i+1)} + \rho_{(j+1)} \quad (6.6)$$

Daha sonra, Denklem 6.7 ve Denklem 6.8 kullanılarak i . ve j . yeni tampon stok değerleri (N'_i, N'_j) elde edilir. Hesaplanan ilk tampon stokun (i) değeri aşağı yuvarlanırken, ikinci tampon stokun (j) değeri yukarı yuvarlanır.

$$N'_i = \left\lfloor \frac{N_{SUM}\rho_{(i+1)}}{\rho_{SUM}} \right\rfloor \quad (6.7)$$

$$N'_j = \left\lceil \frac{N_{SUM}\rho_{(j+1)}}{\rho_{SUM}} \right\rceil \quad (6.8)$$

Örneğin; Şekil 6.3'teki gibi 5 makineli bir hatta toplam 31 tampon stoktan oluşan bir tampon stok konfigürasyonu $(N = \{5, 9, 7, 10\})$ ele alındığında, rastgele seçilen N_1 ve N_3 tampon stoklarının önündeki M_2 ve M_4 'ün ortalama etkin servis oranları sırasıyla 0,8979 ve 0,7586'dır (veriler için bkz. Bölüm 6.8.1). Denklem 6.5 – 6.8 sırasıyla uygulandığında elde edilen yeni tampon stok konfigürasyonu $N^{yeni} = \{6, 9, 6, 10\}$ olur. Bu işlemle birlikte N_3 tampon stok değerinden 1 çıkartılarak N_1 tampon stok değerine 1 eklenmiştir. Konum tabanlı çıkarma-ekleme operatörünün uygulanışında olduğu gibi her yinelemelerde toplam tampon stok miktarı kontrol edilerek toplam tampon stok miktarının sabit kalması sağlanmaktadır.



Şekil 6.3: Durum tabanlı çıkarma-ekleme operatörü: 5 makineli hat örneği

6.4 Adaptif Mekanizma

ABKA algoritmasında kullanılan çıkarma-ekleme operatör çiftlerinin seçimi rulet tekeri yöntemine göre yapılmaktadır. Algoritmanın başlangıcında operatör çiftlerinin seçim olasılıkları ($P_o = 1/2$) eşit olarak belirlenir. Her yinelemede operatör çiftlerinin (o_1, o_2) üretim oranı üzerinde yaptığı iyileştirme kaydedilir ve bir sonraki yinelemede yeniden seçilme olasılığı aşağıdaki formüle göre güncellenir (Ropke ve Pisinger 2006^a, Ghilas ve diğ. 2016):

$$P_j^{i+1} = \frac{D_j^i}{\sum_{i=1}^{5K} D_i}, o \in \{j = 1, 2\} \quad (6.9)$$

Denklem 6.9'da D_j^i j . operatör çiftinin i . yineleme sonucunda başarıyla uygulanma (üretim oranı üzerinde geliştirme yaptığı) sayısını göstermek üzere, P_j^{i+1} j . operatörün i . yineleme sonucundaki ($i+1$. yinelemede) seçilme olasılığıdır. Bir operatör çifti mevcut yinelemede üretim oranı üzerinde artış sağlarsa sonraki yinelemede tekrar seçilme olasılığı bu yöntem ile artırılmaktadır. İlgili operatör çifti bir yinelemede üretim oranı üzerinde iyileşme sağlamazsa Denklem 6.9'a göre D_j^i değeri ve dolayısıyla P_j^{i+1} seçim olasılığı artırılmaz. Bu şekilde, ABKA algoritması TSDP'yi çözmek için adaptif bir şekilde kullanılmıştır.

6.5 IncDec Stratejisi

Artırma-Azaltma (Increment-Decrement: *IncDec*) stratejisi ABKA algoritmasının arama sürecinde yerel optimum çözümden kaçmak için kullanılmaktadır. Eğer $2K$ kadar yineleme sonucunda üretim oranı değişmezse, yani ABKA algoritması bir yerel optimuma (*l-opt* olarak isimlendirilmektedir) ulaşırsa Bölüm 5.3.2’de anlatılan *IncDec* stratejisi uygulanarak yeni aday çözümler elde edilir. Bu stratejide iki tampon stok alanı $((N_i, N_j), (i \neq j))$ ve $i, j \in (1, \dots, K-1)$ seçilir ve ilk seçilen tampon stok kapasitesi belli bir miktar artırılırken ikincisi aynı miktarda azaltılır. Bu miktarın değeri ($id \in \mathbb{Z}^+$), algoritmanın başlangıç aşamasında belirlenir ve algoritma ilerledikçe sistematik olarak azaltılır. Deneysel çalışmalarda bu değer, büyük boyutlu problemler (20 ve 40 makineli) için Demir ve diğ. (2012) tarafından önerildiği gibi toplam tampon stok kapasitesinin %1’i olarak belirlenmiştir. *IncDec* stratejisinin uygulama adımları Algoritma 5’te verilmiştir. *IncDec* stratejisi, önerilen ABKA’da küçük boyutlu problemler için en uygun çözümü hızlı bir şekilde elde etmek için başlangıç sezgiselinden sonra bir kez uygulanmaktadır.

6.6 Kabul Kriteri

Önerilen ABKA algoritmasında sadece amaç fonksiyonunda iyileşme sağlayan çözümler değil, aynı zamanda kötü çözümler de TB (Kirkpatrick ve diğ. 1983) algoritmasına dayalı bir kritere göre kabul edilmektedir. TB’de başlangıç sıcaklığı çeşitliliği artırmak amacıyla yüksek belirlenmektedir. TB algoritmasında mevcut sıcaklığın T olduğu durumda Denklem 6.10’daki gibi v olasılık ile kötü çözümler kabul edilmektedir.

$$v = e^{-10000(Th(N) - Th(N^{yeni})) / T} \quad (6.10)$$

Denklem 6.10’da $Th(N)$ mevcut tampon stok konfigürasyonunun üretim oranını gösterirken, $Th(N^{yeni})$ çıkarma-ekleme operatör çifti uygulanmış yeni tampon stok konfigürasyonunun üretim oranını göstermektedir. Her iki tampon stok konfigürasyonunun üretim oranı değerleri $(0, 1]$ aralığında olduğundan ve farkları çok küçük değere sahip olduğundan v değerinden anlamlı bir olasılık değeri elde etmek

için fark değeri 10000 değeri ile çarpılmaktadır. T değeri mevcut sıcaklık değeridir. Başlangıç sıcaklık değeri (T_0) hattaki makine sayısı (K) olarak belirlenmiştir. ABKA algoritmasının uygulandığı problem setlerinde makine sayısı 5 ile 40 arasında değiştiğinden, soğutma oranı ($c=(K-1)/K$) 0,80 ile 0,975 arasında alınmıştır.

Bu soğutma oranı, arama alanında bir çeşitlendirme yapılmasına izin vermektedir. Her yinelemede $[0, 1]$ aralığında rastgele bir sayı (ϵ) üretilmektedir ve üretilen bu sayı ile v değeri karşılaştırılmaktadır. $v > \epsilon$ olması durumunda kötü çözüm kabul edilmektedir. Arama süreci ilerledikçe Denklem 6.10'daki formüle göre v 'nin ϵ değerinden büyük olma ihtimali giderek azalmaktadır. Bu şekilde kötü çözümlerin ilerleyen yinelemede kabul edilme şansı azalmaktadır.

6.7 Durdurma Kriteri

Amaç fonksiyonunda belirli bir yineleme sayısı boyunca iyileşme olmazsa algoritma durdurulmaktadır. Önceden yapılan pilot deneylerle bu yineleme sayısı $5K$ olarak belirlenmiştir.

Önerilen ABKA algoritması, Algoritma 9'da verilmiştir. Bölüm 6.2'de önerilen başlangıç sezgisellerinden biri kullanılarak ilk tampon stok konfigürasyonu (N^{bas}) oluşturulmaktadır. İlk tampon stok konfigürasyonunun üretim oranı hesaplandıktan sonra küçük boyutlu problemler ($6 \leq K$) için IncDec stratejisi başlangıç sezgiselinden hemen sonra uygulanır. Küçük hatlarda bu ön işlemi uygulamadaki temel amaç, önerilen başlangıç sezgiselleri ile elde edilen tampon stok konfigürasyonlarının problemin optimum çözümüne çok yakın olmasıdır. İlk tampon stok konfigürasyonu oluşturulduktan sonra bir çıkarma-ekleme operatör çifti uygulanır. Üretilen tampon stok konfigürasyonunun üretim oranı hesaplanır ve kabul kriterine göre geçerli çözüm seçilir. Yerel bir optimuma ulaşırsa, yeni bir çözüm elde etmek için IncDec stratejisi kullanılır. Ardından yeni çözüm değerlendirilir ve kabul kriterine göre güncellenir. Bu prosedür, sonlandırma kriteri karşılanana kadar devam eder.

Algoritma 9: Üretim Oranı Maksimizasyonu İçin Önerilen ABKA Algoritması

girdi : Çıkarma/ekleme operatörleri ($o \in O$), T_{bas} , c , IncDec (id) değeri

çıkıtı : Maksimum üretim oranına sahip en iyi tampon stok konfigürasyonu N^{en}

```
1 Bir başlangıç mekanizması kullanarak bir başlangıç çözümü  $N^{bas}$  üret
2 Her bir ekleme/çıkartma operatör çifti için seçim olasılıklarını ( $P_o$ ) eşitle
3  $K \leftarrow T \leftarrow T_{bas}$  sıcaklık,  $i$  iterasyon  $i \leftarrow 1$  ve soğutma oranı  $c \leftarrow (K-1)/K$  olsun
4 Ele alınan problem boyutuna göre  $id$  değerini ayarla
5  $N^{en} \leftarrow N \leftarrow N^{bas}$ 
6 Yerel optimum için izin verilen maksimum yineleme sayısını  $2K \leftarrow l-opt$  olarak ayarla
7 İzin verilen maksimum ardışık yineleme sayısı  $5K$  olarak ayarla
8 Küçük boyutlu problemler için  $N$ 'ye IncDec stratejisini uygula ve  $N^{veni}$  elde et
9 if  $Th(N^{veni}) > Th(N)$  then
10    $N \leftarrow N^{veni}$ 
11 tekrar et
12    $P_o$  olasılıkla bir ekleme/çıkarma operatör çiftini,  $o \in O$ , seç
13    $N$ 'ye  $o$  operatör çifti uygulandığında elde edilen yeni çözüm  $N^{veni}$  olsun
14   if  $Th(N^{veni}) > Th(N)$  then
15      $N \leftarrow N^{veni}$ 
16     if  $Th(N) > Th(N^{en})$  then
17        $N^{en} \leftarrow N$ 
18   else
19      $v \leftarrow e^{-10000(Th(N)-Th(N^{veni}))/T}$ 
20     Rastgele bir sayı üret  $\vartheta \in [0; 1]$ 
21     if  $v > \vartheta$  then
22        $N \leftarrow N^{veni}$ 
23   if  $l-opt \geq 2K$  then
24      $N$ 'ye IncDec stratejisi uygulayarak elde edilen yeni çözüm  $N^{veni}$  olsun
25     if  $Th(N^{veni}) > Th(N)$  then
26        $N \leftarrow N^{veni}$ 
27        $N^{en} \leftarrow N$ 
28      $T \leftarrow cT$ 
29     Kötü çözüm kabul olasılığını güncelle
30     Her bir ekleme/çıkarma operatör çiftinin seçilme olasılığını güncelle
31      $id$  değerini güncelle
32      $i \leftarrow i + 1$ 
33 durdurma kriteri sağlanana kadar
```

% Çözüm Kabulü %

6.8 Deneysel Çalışmalar

Bu bölümde önerilen ABKA algoritmasının performansı literatürdeki mevcut karşılaştırma problemleri üzerinde test edilmiştir. Önceki çalışmalarda çözüm sürelerinin verilmemesi ve daha önemlisi kullanılan donanımların farklılığı nedeniyle karşılaştırmalar ortalama üretim oranı ve değerlendirilen ortalama tampon stok konfigürasyon sayısı (algoritmanın arama süreci boyunca hesaplanan üretim oranı sayısı) üzerinden yapılmıştır. Karşılaştırmada kullanılan literatürdeki yöntemler için sadece ilgili makalelerde raporlanan sonuçlar açısından değerlendirme yapılmıştır.

Önerilen ABKA algoritması MATLAB (R2016a)'da kodlanmıştır. Tüm deneyler, 8 GB RAM'li 3.40 GHz Intel (R) i7 işlemciye sahip bir bilgisayarda gerçekleştirilmiştir. Her problem seti için 10 tekrar yapılmıştır.

Deneysel çalışma için karşılaştırma problemleri hattaki makine sayısı temel alınarak küçük, orta ve büyük boyutlu problemler olmak üzere 3 gruba ayrılmıştır. Önerilen algoritmanın performans analizi aşağıdaki alt bölümlerde ayrıntılı olarak sunulmuştur.

6.8.1 Küçük Boyutlu Problemler

5 Makineli Hat: İlk örnekte, önerilen algoritmanın etkinliği, problem özellikleri Tablo 6.1'de verilen 5 makineli üretim hattı için test edilmiştir. Her tampon stok alanı için alt sınır 4 ile sınırlandırılmış ve toplam tampon stok miktarı 31 olarak belirlenmiştir. Sonuçlar Tablo 6.2'de sunulmuştur.

Tablo 6.1: 5 Makineli hattın güvenilirlik parametreleri (Ho ve diğ. 1979)

Makine (i)	1	2	3	4	5
$1/r_i$	11	19	12	7	7
$1/\beta_i$	20	167	22	22	26

Bu problem seti ilk defa Ho ve diğ. (1979) tarafından önerilmiş ve problemin çözümü için gradyan-tabanlı bir arama algoritması kullanılmıştır. Problem daha sonra, Gershwin ve Schor (2000) tarafından yine gradyan-tabanlı bir arama algoritması ile Demir ve diğ. (2011) tarafından da tabu arama ile çözülmüştür. Tablo 6.2'den de görüleceği üzere, önerilen algoritma, önceki çalışmalarda olduğu gibi {7, 10, 10, 4} tampon stok konfigürasyonu ile 0,4943 olan optimum üretim oranı değerine ulaşmıştır.

Önerilen tüm başlangıç sezgisellerinin 10 tekrarın tümü için optimum çözümü elde ettiği gözlemlenmiştir. Önerilen ABKA-MEKF ve ABKA-DKF tüm başlangıç sezgiselleri arasında en düşük değer olan 10 değerlendirme ortalamasıyla optimum çözüme ulaşmışlardır. Bunun nedeni, MEKF ve DKF başlangıç sezgisellerinin bu problem için optimum çözüme çok yakın bir konfigürasyon üretmesidir.

Tablo 6.2: 5 makineli hattın karşılaştırmalı sonuçları

Çalışmalar/Yöntem	Tampon Stok Dağılımı (N_i)				En İyi Çözüm Say.	Ort. Üretim Oranı	Ort. Değ. Say.
	N_1	N_2	N_3	N_4			
Ho ve diğ. (1979)	5	11	8	7	-	0,4914	-
Gershwin ve Schor (2000)	7	10	10	4	-	0,4943	-
Demir ve diğ. (2011)	7	10	10	4	-	0,4943	-
ABKA-Rastgele	7	10	10	4	10	0,4943	29,7
ABKA-PaVi	7	10	10	4	10	0,4943	27,0
ABKA-MEKF	7	10	10	4	10	0,4943	10,0
ABKA-DKF	7	10	10	4	10	0,4943	10,0

Bu küçük boyutlu örneği çözmenin amacı, önerilen ABKA algoritmasının optimum çözüm bulma kapasitesine sahip olduğunu doğrulamaktır. Önerilen algoritmanın performansı hakkında daha genel yorumlar yapabilmek için, deneysel çalışma orta ve büyük ölçekli problemler ele alınarak genişletilmiştir.

6.8.2 Orta Büyüklükte Problemler

9 Makineli Hat: Bu problem seti 9 farklı senaryo içermektedir. Makinelerin güvenilirlik parametreleri değiştirilerek oluşturulmuş her bir senaryoda 9 makineden oluşan hat için toplam 160 tampon stok kısıtı altında üretim oranı maksimize edilmiştir. 9 senaryoya ait özellikler ve elde edilen sonuçlar Tablo 6.3'te verilmiştir. Bu problem seti ilk defa Shi ve Men (2003) tarafından önerilmiştir ve daha sonra Demir ve diğ. (2011) tarafından kullanılmıştır. Shi ve Men (2003) bu problem için NP/TS olarak adlandırdıkları tabu arama tabanlı melez bir algoritma önerirken, Demir ve diğ. (2011) ise DTL-TS olarak adlandırdıkları başka bir TA yöntemi kullanmışlardır.

Önceki örnekte olduğu gibi, önerilen ABKA algoritması diğer TA tabanlı yöntemlerden biraz daha yüksek üretim oranı değerine ulaşmıştır (Tablo 6.3). ABKA-DKF için en iyi çözüme ulaşmak için gereken ortalama değerlendirme sayısı 98.6 ve 135.0 aralığında iken, bu sayı ABKA-MEKF için 108,8-145,0 aralığında, ABKA-PaVi için 117,3-188,4 aralığında ve ABKA-Rastgele için 197,7-267,6 arasında değişmiştir. Bu nedenle, ABKA-DKF'nin yakınsama hızı açısından en iyi performansa sahip olduğu sonucuna varılabilir. Bu deneysel sonuçlar, önerilen DKF sezgiselinin iyi bir

başlangıç çözümü oluşturmadaki etkinliğini göstermiştir. Ayrıca, önerilen ABKA algoritmasının, tüm başlangıç sezgiselleri için 10 tekrardan 10'unda da en iyi çözümü elde ettiği ve çözüm süresinin ortalama 3,5 ile 4,8 sn arasında olduğu görülmüştür.

Tablo 6.3: 9 Makineli hattın güvenilirlik parametreleri ve karşılaştırmalı sonuçları

Senaryo	Makine Parametreleri		Shi ve Men (2003)	Demir ve diğ. (2011)	ABKA-Rastgele		ABKA-PaVi		ABKA-MEKF		ABKA-DKF	
	$1/\beta_i$	$1/r_i$	Üretim Oranı	Üretim Oranı	Üretim Oranı	Ort. Değ. Say.	Üretim Oranı	Ort. Değ. Say.	Üretim Oranı	Ort. Değ. Say.	Üretim Oranı	Ort. Değ. Say.
1	0,3	0,05	0,108143	0,108147	0,108240	223,8	0,108240	117,3	0,108240	126,1	0,108240	98,6
2	0,3	0,10	0,200250	0,200255	0,200357	241,6	0,200357	151,5	0,200357	144,5	0,200357	109,5
3	0,3	0,20	0,345491	0,345495	0,345580	204,7	0,345580	136,8	0,345580	118,5	0,345580	116,9
4	0,3	0,30	0,452074	0,452077	0,452151	215,8	0,452151	139,0	0,452151	130,3	0,452151	113,7
5	0,3	0,40	0,532002	0,532006	0,532091	207,2	0,532091	185,8	0,532091	125,4	0,532091	118,3
6	0,4	0,05	0,088777	0,088782	0,088857	243,3	0,088857	188,4	0,088857	108,8	0,088857	122,8
7	0,4	0,10	0,166232	0,166236	0,166322	245,9	0,166322	163,6	0,166322	129,8	0,166322	135,0
8	0,4	0,20	0,293041	0,293046	0,293199	267,6	0,293199	154,1	0,293199	142,6	0,293199	105,7
9	0,4	0,30	0,390814	0,390819	0,390881	197,7	0,390881	153,8	0,390881	145,0	0,390881	129,1

10 Makineli Hat: Bu problem ilk olarak Nahas ve diğ. (2006) tarafından önerilmiştir. Problem daha sonra Demir ve diğ. (2011)'nin çalışmasında kullanılmıştır. Makinelerin güvenilirlik parametreleri Tablo 6.4'te verilmiştir. Toplam tampon stok kapasitesi 270 olarak belirlenmiştir. Problemin çözümü için Nahas ve diğ. (2006) İTA önermişlerdir. Demir ve diğ. (2011) ise DTL-TS olarak isimlendirdikleri bir TA algoritması önermişlerdir. Sonuçlar Tablo 6.5'te gösterilmiştir.

Tablo 6.4: 10 Makineli hattın güvenilirlik parametreleri (Nahas ve diğ. 2006)

Makine (<i>i</i>)	1	2	3	4	5	6	7	8	9	10
$1/r_i$	7	7	5	10	9	14	5	8	10	10
$1/\beta_i$	20	33	22	22	25	40	23	30	45	20

Tablo 6.5'ten de görüleceği üzere önerilen ABKA algoritması tarafından elde edilen en iyi üretim oranı değeri, {14, 20, 30, 53, 45, 27, 23, 25, 33} tampon stok konfigürasyonu ile 0,64348'dir. Önceki örneklerde olduğu gibi, önerilen başlangıç sezgiseli DKF, daha az değerlendirme sayısı ile en iyi çözüme ulaşmıştır. Önerilen ABKA algoritmasının performansı çözüm süresi açısından değerlendirildiğinde, en iyi çözümü elde etmek için gereken ortalama sürenin 4,42-5,67 sn arasında olduğu görülmektedir.

Tablo 6.5: 10 makineli hattın karşılaştırmalı sonuçları

Çalışmalar/Yöntem	En İyi Çözüm Say.	Ort. Üretim Oranı	Ort. Değ. Say.	Ort. CPU Süresi (sn)
Nahas ve diğ. (2006)	-	0,64135	-	-
Demir ve diğ. (2011)	-	0,64135	-	-
ABKA-Rastgele	7	0,64347	467	5,67
ABKA-PaVi	10	0,64348	397	4,81
ABKA-MEKF	10	0,64348	391	4,44
ABKA-DKF	10	0,64348	383	4,42

6.8.3 Büyük Boyutlu Problemler

Bu bölümde önerilen algoritmanın etkinliği, büyük boyutlu problemler için test edilmiştir. Bu amaçla, Li ve diğ. (2015) tarafından önerilen farklı güvenilirlik parametrelerine sahip 20 ve 40 makineli üretim hatları ele alınmıştır. Problem, 20 makineli hatlar için toplam tampon stok miktarı 400 kısıtı altında, 40 makineli hatlar

içinse 400, 800 ve 1600 olarak belirlenmiş toplam tampon stok miktarları için çözülmüştür. 20 makineli hattaki makinelere ait güvenilirlik parametreleri Tablo 6.6’da verilmiştir. 40 makineli hat, 20 makineli hattın tekrar edilmesiyle oluşturulmuştur.

Tablo 6.6: 20 makineli hattın güvenilirlik parametreleri (Li ve diğ. 2015)

Makine (<i>i</i>)	1	2	3	4	5	6	7	8	9	10
β_i	0,011	0,005	0,022	0,025	0,052	0,021	0,017	0,013	0,011	0,016
r_i	0,039	0,016	0,166	0,085	0,186	0,105	0,116	0,136	0,043	0,139
Makine (<i>i</i>)	11	12	13	14	15	16	17	18	19	20
β_i	0,019	0,014	0,013	0,025	0,014	0,001	0,006	0,013	0,059	0,038
r_i	0,074	0,099	0,111	0,178	0,094	0,015	0,061	0,046	0,184	0,134

Li ve diğ. (2015) problemin çözümü için FA olarak isimlendirdikleri sezgisel bir algoritma önermişlerdir. Önerilen ABKA algoritmasında elde edilen sonuçlarla FA’nın sonuçları karşılaştırmalı olarak Tablo 6.7’de sunulmuştur. Tablo 6.7’de görüldüğü gibi, önerilen ABKA-DKF algoritması, her test problemi için üretim oranı açısından ortalama olarak sırasıyla %1,85, %1,59, %2,77 ve %3,66 daha iyi sonuçlar vermiştir. Şekil 6.4, önerilen ABKA-DKF algoritmasının çözüm kalitesi açısından diğer algoritmalar arasında en iyi çözümü ürettiğini göstermektedir. Hem Tablo 6.7 hem de Şekil 6.4’te görülebileceği gibi, önerilen ABKA-DKF’nin etkinliği büyük boyutlu örnekler için daha belirgin hale gelmiştir. Yani üretim oranındaki iyileşmeler önceki problemlere oranla daha fazladır. Ayrıca, Li ve diğ. (2015) çalışmalarında “*en iyi üretim oranının art arda beş yineleme için değişmemesi durumunda artık değişmeyeceğini*” ifade etmişlerdir. Dolayısıyla, Li ve diğ. (2015) tarafından elde edilen üretim oranı değerleri önerdikleri FA algoritması ile elde edilebilecek en iyi değerlerdir. Bu da önerilen ABKA-DKF’nin çözüm kalitesi açısından FA’dan daha iyi olduğunu göstermektedir. Önerilen ABKA-DKF algoritmasının performansı çözüm süresi açısından değerlendirildiğinde, en iyi çözümü elde etmek için gereken sürenin 20 makineli hat için ortalama 20,4-31,2 sn arasında olduğu ve 40 makineli hat için ise ortalama 58,9-141,8 sn arasında olduğu görülmektedir. Öte yandan, Li ve diğ. (2015) tarafından FA’nın çözüm süresinin birkaç saniye olduğu belirtilmiştir. Çözüm süresindeki bu farklılığın bir nedeni, kullanılan ayrıştırma yöntemindeki farklılıktır. 5. bölümdeki deneysel çalışmalarda da belirtildiği üzere Li ve diğ. (2015), üretim oranı değerlendirmelerinin sayısını önemli ölçüde azaltan üç makineli-iki tampon stok alanlı

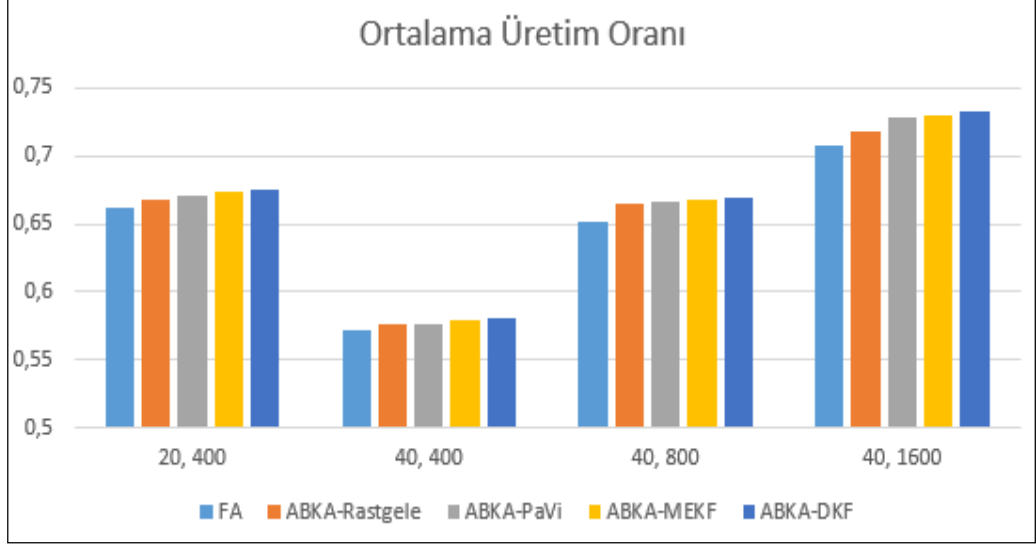
bir ayrıştırma yöntemi önermişler ve TSDP için önerdikleri FA algoritması ile beraber kullanmışlardır. Önerilen ABKA algoritmasının çözüm süresinin nispeten uzun olmasının nedeni ise IncDec stratejisi uygulanırken mevcut çözümün tüm komşu çözümlerinin değerlendirilmesidir. Bununla birlikte TSDP'nin bir tasarım problemi olduğu düşünüldüğünde, ABKA algoritmasının gerektirdiği sürelerin çok da uzun olmadığı ve iyi bir çözümün makul bir hesaplama süresi içinde elde edildiğini söylemek mümkündür. Talbi (2009)'nin de belirttiği gibi tasarım problemleri için çözüm kalitesi arama süresinden daha önemli olduğu için sonuç olarak önerilen ABKA-DKF algoritmasının TSDP'yi makul bir hesaplama zamanında çözebilecek kapasitede olduğu görülmektedir.

Tablo 6.7: 20 ve 40 makineli hatların karşılaştırmalı sonuçları

(K, N)	Li ve diğ. (2015) (FA)		ABKA-Rastgele			ABKA-PaVi			ABKA-MEKF			ABKA-DKF		
	En İyi Üretim Oranı	Ort. Değ. Say.	Ort. Üretim Oranı	Ort. Değ. Say.	Ort. İyileş. (%) **	Ort. Üretim Oranı	Ort. Değ. Say.	Ort. İyileş. (%) **	Ort. Üretim Oranı	Ort. Değ. Say.	Ort. İyileş. (%) **	Ort. Üretim Oranı	Ort. Değ. Say.	Ort. İyileş. (%) **
(20, 400)	0,662264	100	0,667560	2561	0,80	0,671502	2215	1,39	0,673134	2188	1,64	0,674547*	2173	1,85
(40, 400)	0,571864	-	0,576167	4998	0,75	0,576738	4820	0,85	0,579413	4733	1,32	0,580973*	4629	1,59
(40, 800)	0,651023	-	0,665010	7147	2,15	0,666939	5658	2,44	0,668434	5471	2,67	0,669077*	5238	2,77
(40, 1600)	0,707086	-	0,718852	8423	1,66	0,728926	7199	3,09	0,730675	7083	3,34	0,732937*	6908	3,66

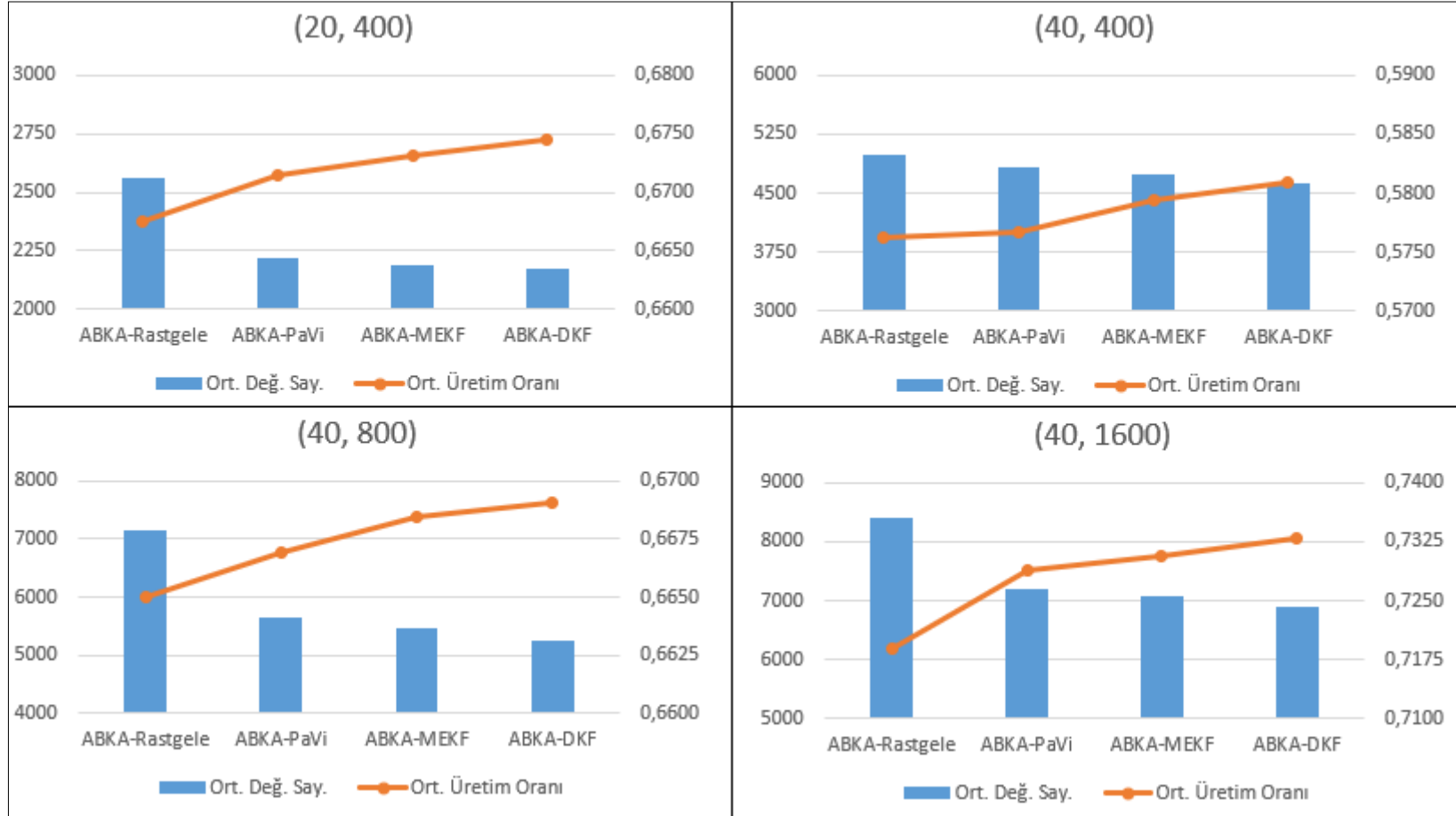
*en iyi çözüm

** İyileşme oranı (%) = $100 \times [Th (ABKA) - Th (FA)] / Th (FA)$



Şekil 6.4: Önerilen ABKA algoritması ve FA karşılaştırması

Şekil 6.5'te önerilen başlangıç sezgisellerinin performansları karşılaştırılmıştır. Grafiklerden de görüleceği gibi, önerilen DKF başlangıç sezgiseli, üretim oranı değerlendirme sayısını büyük ölçüde azaltmaktadır. Bu deneyden, algoritmayı 'iyi' bir başlangıç çözümüyle başlatmanın arama eforunu önemli ölçüde azalttığı ve daha iyi çözümler elde edildiği sonucuna varılabilir. Dolayısıyla, önerilen DKF sezgiselinin güvenilir olmayan üretim hatları için iyi bir başlangıç çözümü oluşturduğunu ve önerilen ABKA algoritması ile birlikte kullanıldığında iyi çözümlere kısa sürede ulaşabildiğini söylemek mümkündür.



Şekil 6.5: ABKA’da kullanılan başlangıç sezgisellerinin karşılaştırması

6.9 Bulgular ve Tartışma

Tablo 6.8’de bu bölümde yapılan tüm deneysel çalışmalar özetlenmiştir. Tablo 6.8’de görüldüğü gibi, önerilen ABKA-DKF algoritması literatürde mevcut en iyi çözümlerle aynı veya daha iyi sonuçlar üretmiştir. Ayrıca önerilen algoritmanın etkinliğinin problem boyutu arttıkça daha da arttığı görülmüştür. Deneysel çalışmalar, önerilen ABKA algoritmasının hesaplama karmaşıklığının, K makine sayısı olmak üzere, ortalama değerlendirme sayısı açısından $O(K^2)$ mertebesinde olduğunu göstermektedir.

Tablo 6.8’de ayrıca, tüm problem setleri ve başlangıç sezgisellerine göre önerilen çıkarma-ekleme operatörü çiftlerinin başarıyla uygulanma oranları verilmiştir. Tablo 6.8 incelendiğinde durum-tabanlı çıkarma-ekleme operatör çiftinin ortalama olarak %78,9’unda başarıyla kullanıldığı görülmüştür. Hattaki makineler bozulmaya maruz kaldığından, önerilen durum-tabanlı çıkarma-ekleme sezgiseli yöntemi, beklendiği gibi baskın operatör olarak karşımıza çıkmıştır. Ayrıca Tablo 6.8’de sunulduğu gibi problem boyutu arttıkça bu operatör çiftinin kullanımı da artmıştır.

ABKA algoritmasının yapısı çok basit olmasına rağmen komşuluk temelli olması nedeniyle her zaman yerel optimuma takılıp kalma olasılığı vardır. Önerilen ABKA algoritmasındaki bu dezavantaj, IncDec stratejisi kullanılarak aşılmıştır. Bu strateji, yalnızca yerel optimumdan kaçmak için değil, aynı zamanda çözüm kalitesini iyileştirmek için de kullanılmıştır. Öte yandan IncDec stratejisini uygularken tüm komşu çözümleri değerlendirmek, algoritmanın çözüm süresini artırmıştır.

Tablo 6.8: Deneysel sonuçların özeti

Problem Seti	Başlangıç Sezgiseli	Diğer Algoritmalara Göre İyileşme Oranı (%)	Operatörlerin Başarıyla Uygulanma Oranı	
			Konum-tabanlı (%)	Durum-tabanlı (%)
(5, 31)	Rastgele	0,0	56	44
	PaVi	0,0	0*	0*
	MEKF	0,0	0*	0*
	DKF	0,0	0*	0*
(9, 160)	Rastgele	0,04	49	51
	PaVi	0,04	26	74
	MEKF	0,04	18	82
	DKF	0,04	20	80
(10, 270)	Rastgele	0,33	28	72
	PaVi	0,33	29	71
	MEKF	0,33	25	75
	DKF	0,33	23	77
(20, 400)	Rastgele	0,80	23	77
	PaVi	1,39	17	83
	MEKF	1,64	17	83
	DKF	1,85	18	82
(40, 400)	Rastgele	0,75	13	87
	PaVi	0,85	17	83
	MEKF	1,32	18	82
	DKF	1,59	21	79
(40, 800)	Rastgele	2,15	11	89
	PaVi	2,44	18	82
	MEKF	2,67	14	86
	DKF	2,77	11	89
(40, 1600)	Rastgele	1,66	12	88
	PaVi	3,09	18	82
	MEKF	3,34	12	88
	DKF	3,66	13	87

* operatörler aracılığıyla hiçbir iyileşme elde edilmemiştir.

Şimdiye kadar sunulan deneysel sonuçlar, önerilen ABKA-DKF algoritmasının literatürdeki mevcut çözümlerle aynı veya mevcut çözümlerden daha iyi sonuçlar ürettiğini göstermiştir. Karşılaştırmalı sonuçlardan da görüleceği gibi üretim oranları arasındaki farkın az olması nedeniyle bu fark değerlerinin istatistiksel olarak anlamlı olup olmadığı test edilmelidir. Bu amaçla Wilcoxon işaretli-sıra testi (*Wilcoxon signed-rank test*) kullanılmıştır (Tablo 6.9). Aşağıdaki H_0 hipotezi $\alpha=0,05$ anlamlılık düzeyinde test edilmiştir:

$$H_0: Th(\text{Literatürdeki En İyi}) - Th(\text{ABKA-DKF}) = 0$$

Tablo 6.9: Wilcoxon işaretli-sıra testi

H_0 hipotezi	Test	Anlamlılık (p)	Karar
Literatürdeki en iyi sonuç ve ABKA-DKF arasındaki farkların medyanı 0'a eşittir.	İlişkili örneklemeler Wilcoxon işaretli-sıralar testi	0,001	H_0 hipotezi reddedilir

Wilcoxon testi sonucunda elde edilen p değeri üretim oranı değerleri arasındaki farkın %95 güven düzeyinde istatistiksel olarak anlamlı olduğunu göstermektedir (Tablo 6.9). Tüm bu analizlerin sonucu olarak önerilen ABKA-DKF algoritmasının, güvenilir olmayan üretim hatlarında TSDP'yi çözmek için etkili bir algoritma olduğunu söylemek mümkündür.

6.10 Sonuç

Bu bölümde bozulmaya maruz kalan makinelerden oluşan seri üretim hatlarında TSDP'nin çözümü için bir ABKA algoritması önerilmiştir. Bu çalışmayla literatürde ilk defa bir üretim hattı tasarım problemini çözmek için ABKA algoritması kullanılmıştır. ABKA algoritmasını problemin yapısına uyarlayabilmek için iki yeni çıkarma-ekleme operatör çifti önerilmiş ve algoritma boyunca adaptif bir şekilde uygulanmıştır. Ayrıca, ters kâse fenomenine dayalı DKF adı verilen yeni bir başlangıç sezgiseli önerilmiş ve ABKA algoritması ile birlikte etkin bir şekilde kullanılmıştır.

TSDP üretim oranının maksimizasyonu için çözülmüş ve önerilen ABKA-DKF algoritmasının performansı karşılaştırma problemleri üzerinde test edilmiştir. Deneysel çalışmalar önerilen ABKA algoritmasının, dikkate alınan tüm karşılaştırma problemleri için literatürdeki mevcut en iyi çözümleri ya da mevcut çözümlerden daha iyi çözümler ürettiğini göstermiştir. Ayrıca önerilen DKF başlangıç sezgiselinin arama eforunu önemli ölçüde azalttığı gözlemlenmiştir. Önerilen algoritmanın etkinliğinin özellikle büyük boyutlu problemlerde (20 ve 40 makineli hatlar) daha belirgin olduğu görülmüştür. Özetle, önerilen ABKA-DKF algoritmasının güvenilir olmayan üretim hatlarında TSDP'yi çözmek için oldukça etkin bir algoritma olduğu ve büyük boyutlu problemleri çözmeye literatürde mevcut diğer algoritmalarından daha iyi olduğu sonucuna varılabilir.

7. TAMPON STOK DAĞILIMI PROBLEMİNİN ÇÖZÜMÜNDE POPÜLASYON TABANLI ARAMA ALGORİTMALARI: SAVAŞÇI GENETİK ALGORİTMA VE BÜYÜK PATLAMA-BÜYÜK ÇÖKÜŞ ALGORİTMASI

7.1 Giriş

Tez çalışmasının bu bölümünde TSDP'nin çözümü için popülasyon tabanlı iki algoritma önerilmiştir: savaşçı genetik algoritma (SGA) ve büyük patlama-büyük çöküş (BP-BÇ) algoritması. Her iki algoritma da aynı yazarlar (Eksin ve Erol 2001, Erol ve Eksin 2006) tarafından literatüre tanıtılmış ve sürekli fonksiyonların optimizasyonu için başarılı bir şekilde kullanılmıştır. Bu tez çalışmasında her iki algoritma da ilk defa bir üretim sistemi tasarımı problemi olan TSDP'yi çözmek için kullanılmış ve problemin yapısına uygun olarak uyarlanmıştır.

Bölümün geri kalanı şu şekilde organize edilmiştir. Bölüm 7.2'de SGA'nın, Bölüm 7.3'te ise BP-BÇ algoritmasının TSDP'ye uyarlanması anlatılmıştır. Bölüm 7.4'te önerilen algoritmalarda kullanılan parametrelerin optimizasyonu için yapılan çalışmalara ve mevcut literatür problemleri üzerinde gerçekleştirilen karşılaştırmalı deneysel çalışmalara yer verilmiştir. Bölüm 7.5'te çalışmadan elde edilen sonuçlar tartışılarak elde edilen bulgular sunulmuştur. Son olarak, Bölüm 7.6'da bu bölümde yapılan çalışma özetlenmiştir.

7.2 Savaşçı Genetik Algoritma

Popülasyon tabanlı bir optimizasyon algoritması olan SGA, Eksin ve Erol (2001) tarafından GA'nın yakınsama sorunlarını ortadan kaldırmak ve arama sürecini hızlandırmak için geliştirilmiştir. SGA'da, seçim, çaprazlama ve mutasyon operatörleri her yinelemede sırayla uygulanır. Bu bölümde, SGA'nın TSDP'nin çözümünde nasıl kullanıldığı detaylı olarak anlatılmıştır.

Algoritmanın başlangıç aşamasında, P (popülasyon büyüklüğü) adet tampon stok konfigürasyonu rastgele oluşturulmaktadır. SGA'nın başlangıç popülasyon büyüklüğü problem boyutuna göre değişmekle birlikte, tez çalışması kapsamında ele alınan problemler için bu değer $2K - 5K$ arasında değişmektedir. Başlangıç popülasyonu oluşturulduktan sonra popülasyonun içinden iki farklı tampon stok konfigürasyonu rastgele seçilir. Seçilen konfigürasyonların üretim oranı değerleri hesaplandıktan sonra, Denklem 7.1 ile bu üretim oranları arasındaki bağıl fark değeri bulunur:

$$\Delta_r = 100 \frac{|Th(N^{ilk}) - Th(N^{ikinci})|}{Th(N^{ilk}) + Th(N^{ikinci})} \quad (7.1)$$

Rastgele seçilen bu iki farklı aday çözüme 'savaşçı kromozomlar (*combat chromosomes*)' adı verilmiştir ve bu sebeple algoritma, 'savaşçı genetik algoritma' olarak adlandırılmıştır (Eksin ve Erol 2001). Denklem 7.1'de elde edilen bağıl fark dikkate alınarak, seçilen tampon stok konfigürasyonlarına uygulanacak işlemlere aşağıdaki kurallara göre karar verilir. ϑ , $[0, 1]$ aralığında üretilen rastgele bir sayı olmak üzere:

1. $\Delta_r > \vartheta$
 - a) $Th(N^{ilk}) > Th(N^{ikinci})$ sadece N^{ikinci} için kısmi üzerine yazma operatörü uygulanır.
 - b) $Th(N^{ikinci}) > Th(N^{ilk})$ sadece N^{ilk} için kısmi üzerine yazma operatörü uygulanır.
2. $\vartheta > \Delta_r$ iki noktalı çaprazlama operatörü uygulanır.

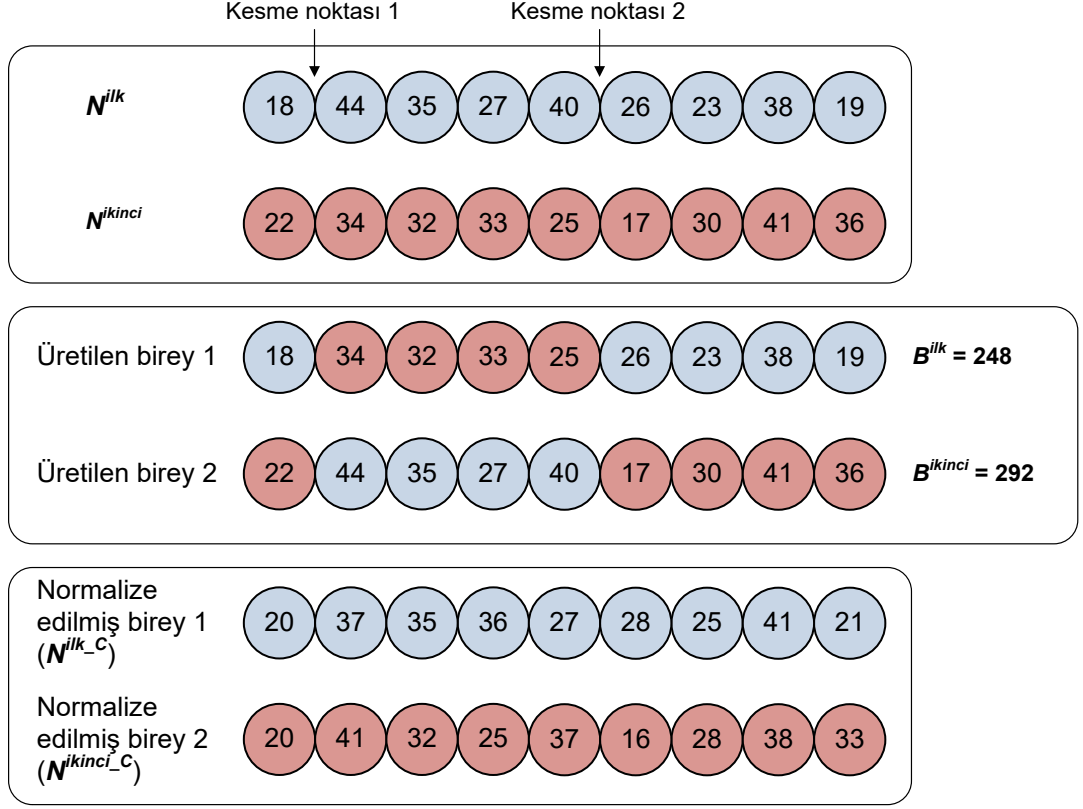
Δ_r, ϑ 'den daha büyükse, 1.a) veya 1.b) kurallarına dayanarak sırasıyla N^{ikinci} ve N^{ilk} tampon stok konfigürasyonlarına kısmi üzerine yazma operatörü (*partial overwrite operator*) uygulanır ve bir yeni tampon stok konfigürasyonu üretilir. Bu, daha az uygun olan kromozomun bazı bilgilerini kaybetmesi, yani üretim oranı değeri daha düşük olan tampon stok konfigürasyonunun bazı tampon stok değerlerinin değişmesi demektir. Bu sayede kısmi üzerine yazma operatörü uygulanan tampon stok konfigürasyonunun uygunluk değerinin, diğer tampon stok konfigürasyonundan aldığı bilgiyle artırılması amaçlanır. ϑ, Δ_r 'den daha büyükse, iki noktalı çaprazlama (*two-*

point crossover) uygulanır ve iki yeni tampon stok konfigürasyonu elde edilir. Bunun anlamı, birbirine üretim oranı değeri olarak yakın olan iki tampon stok konfigürasyonunun her ikisinin de belirli aralıktaki tampon stok değerlerini değiştirerek daha adil bir ‘savaş’ için üretim oranı değerlerini artırma şansı vermektir. TSDP’deki uygunluk değeri, hattın üretim oranıdır ve bu değer (0, 1] aralığındadır. İki üretim oranı değeri arasındaki fark çok küçük olduğundan, orijinal SGA’dan farklı olarak hesaplanan fark 100 ile çarpılarak Δ_r değeri elde edilmektedir (Denklem 7.1).

Orijinal SGA’dan farklı olarak, bu tez çalışmasında iki noktalı çaprazlama operatörü kullanılmıştır. İki noktalı çaprazlama operatörünün uygulanmasının nedeni, bilgiyi (belirli tampon stok değerlerini) doğrudan daha uygun tampon stok konfigürasyonundan alması ve bu şekilde daha iyi yakınsama sağlamasıdır. Şekil 7.1, dağıtılacak toplam tampon stok miktarının 270 olduğu 10 makineli bir hat için iki noktalı çaprazlama operatörünün uygulanışını göstermektedir. Toplam tampon stok miktarını sabit tutabilmek için, tampon stok konfigürasyonlarına çaprazlama operatörü uygulandıktan sonra elde edilen konfigürasyon Denklem 7.2 yardımıyla normalize edilmektedir:

$$N_i^{normalize edilmiş} = \left(\frac{N}{\text{Üretilen toplam tampon stok miktarı}} * B_i^{\text{üretilen}} \right), \forall i \quad (7.2)$$

$N^{ilk} = \{18, 44, 35, 27, 40, 26, 23, 38, 19\}$ ve $N^{ikinci} = \{22, 34, 32, 33, 25, 17, 30, 41, 36\}$ olan iki tampon stok konfigürasyonu için rastgele kesme noktaları ikinci ve beşinci tampon stok alanları olarak seçilmişse iki noktalı çaprazlama sonucunda Şekil 7.1’deki gibi toplam tampon stok miktarları 248 ve 292 olan geçici bireyler elde edilir. Toplam tampon stok miktarının (270) her iki tampon stok konfigürasyonu için de korunması amacıyla normalizasyon işlemi uygulanarak normalize edilmiş bireyler, $N^{ilk-C} = \{20, 37, 35, 36, 27, 28, 25, 41, 21\}$ ve $N^{ikinci-C} = \{20, 41, 32, 25, 37, 16, 28, 38, 33\}$ olarak elde edilir.

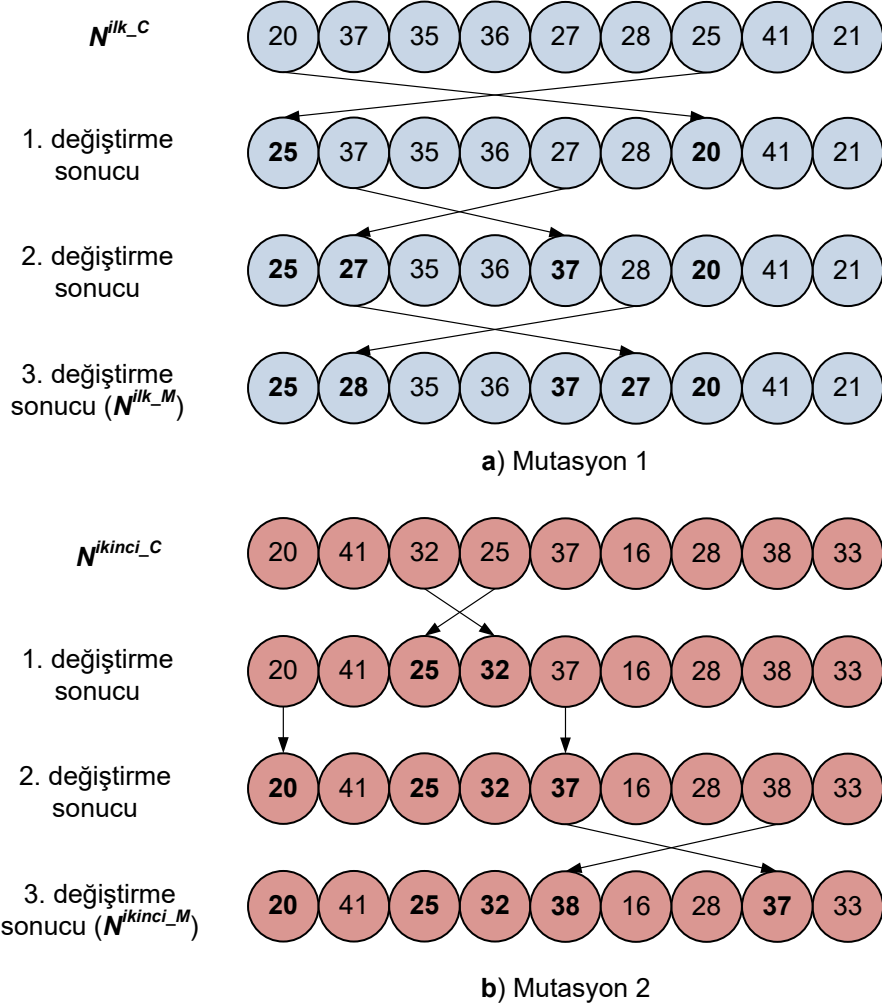


Şekil 7.1: 10 makineli hat için iki noktalı çaprazlama operatörünün uygulanması

SGA'nın bir sonraki aşamasında çaprazlama operatörü ile elde edilmiş tampon stok konfigürasyonlarına mutasyon işleminin uygulanması gerekir. Bu tez çalışmasında iki farklı mutasyon operatörü kullanılmıştır. Mutasyon işlemi sırasında kullanılan operatör rulet tekeri yöntemi ile belirlenmektedir (Denklem 6.9). Algoritma ilerledikçe, amaç fonksiyonunu iyileştiren mutasyon operatörü her yinelemede kaydedilerek aynı operatörün bir sonraki yineleme için yeniden seçim olasılığı artırılmaktadır. Çaprazlama aşamasında tampon stokların değeri değiştiğinden, her iki mutasyon operatörü de popülasyon çeşitliliğini artırmak ve tüm çözüm alanını etkin bir şekilde araştırmak için yalnızca tampon stokların konumunu değiştirmektedir.

Mutasyon 1'de Şekil 7.2a'da gösterildiği gibi rastgele seçilen iki tampon stokun konumları değiştirilmektedir. Mutasyon 2'de ise Hillier ve diğ. (1993) tarafından geliştirilen ters kâse fenomeni kavramından yola çıkarak hattın ortasında daha fazla tampon stok olacak şekilde, tampon stokların konumları değiştirilmektedir. Rastgele seçilen iki tampon stoka bu operatör şu şekilde uygulanır: Seçilen tampon stokların değeri göz önünde bulundurularak, daha büyük olan hattın başında veya

sonunda yer alıyorsa, küçük olan tampon stokla yer değiştirilir. Aksi takdirde, tampon stok konfigürasyonunda herhangi bir değişiklik yapılmaz (Şekil 7.2b). Algoritmanın başlangıcında K/P olarak belirlenen mutasyon oranı üstel azaltma stratejisi ile algoritma ilerledikçe $1/K$ 'ya düşürülmektedir (Back 1993, Cvetkovic ve Parmee 2002, Thierens 2002, Hamdi ve Zaied 2019).



Şekil 7.2: 10 makineli hat için önerilen mutasyon operatörlerinin uygulanması

Çaprazlama ve mutasyon operatörlerinin uygulanması sonucunda elde edilen tampon stok konfigürasyonlarından en iyi üretim oranı deęerini üreten konfigürasyon bir sonraki yineleme için en iyi tampon stok konfigürasyonu olarak seçilir. Aynı prosedür başlangıçta oluşturulan popülasyondan farklı iki tampon stok konfigürasyonu seçilerek tekrarlanır. Amaç fonksiyonu deęeri $10K$ ardışık yineleme boyunca deęişmezse, yani algoritma yerel optimuma ($l-opt$) takılırsa, yeni bir çözüm elde etmek için mevcut en iyi çözüme 5. bölümde ayrıntıları verilmiş olan IncDec stratejisi

uygulanır. IncDec stratejisini üç kez uyguladıktan sonra, eldeki mevcut en iyi çözüm değişmezse, bu strateji tekrar uygulanmaz. Algoritma, izin verilen maksimum yineleme sayısı olan $100K$ 'ya ulaştığında sonlandırılır. Buradaki maksimum yineleme sayıları yapılan pilot deneylerle belirlenmiştir. TSDP'yi çözmek için önerilen SGA'nın genel çerçevesi Algoritma 10'da verilmiştir.

Algorithm 10: Üretim Oranı Maksimizasyonu İçin Önerilen SGA

girdi : Makine sayısı K , toplam tampon stok sayısı N , popülasyon boyutu P , *IncDec* değeri
çıkıtı : Maksimum üretim oranına sahip en iyi tampon stok konfigürasyonu N^{best}

- 1 Yerel optimum için izin verilen maksimum yineleme sayısını ($l-opt$) $10K$ olarak ayarla
- 2 İzin verilen maksimum yineleme sayısını $100K$ olarak ayarla
- 3 Problem boyutuna göre *IncDec* değerini ayarla
- 4 $N^{geçici} \leftarrow (0, 0, \dots, 0) \leftarrow (N_1^{geçici}, N_2^{geçici}, \dots, N_{K-1}^{geçici})$ olsun
- 5 $N^{ilk} \leftarrow N^{ikinci} \leftarrow N^{en} \leftarrow N^{geçici} \leftarrow N^{veni}$
- 6 Rastgele P adet tampon stok konfigürasyonu üret
- 7 **tekrar et**
- 8 İki farklı tampon stok konfigürasyonunu (N^{ilk}, N^{ikinci}) rastgele seç
- 9 Seçilen tampon stok konfigürasyonlarının üretim oranı değerini hesapla
- 10 Δ_r değerini Denklem (7.1) kullanarak hesapla
- 11 $0 - 1$ aralığında rastgele bir sayı, $\vartheta \in [0, 1]$ üret
- 12 **if** $Th(N^{ikinci}) > Th(N^{ilk})$ **and** $\Delta_r > \vartheta$ **then**
- 13 $N^{ikinci_C} \leftarrow N^{ikinci}$
- 14 $N^{ilk_C} \leftarrow \text{çaprazlama}\{N^{ilk}, N^{ikinci}\}$
- 15 **else if** $Th(N^{ikinci}) > Th(N^{ilk})$ **and** $\vartheta > \Delta_r$ **then**
- 16 $N^{ilk_C} \leftarrow \text{çaprazlama}\{N^{ilk}, N^{ikinci}\}$
- 17 $N^{ikinci_C} \leftarrow \text{çaprazlama}\{N^{ilk}, N^{ikinci}\}$
- 18 **else if** $Th(N^{ilk}) > Th(N^{ikinci})$ **and** $\Delta_r > \vartheta$ **then**
- 19 $N^{ikinci_C} \leftarrow \text{çaprazlama}\{N^{ilk}, N^{ikinci}\}$
- 20 $N^{ilk_C} \leftarrow N^{ilk}$
- 21 **else if** $Th(N^{ilk}) > Th(N^{ikinci})$ **and** $\vartheta > \Delta_r$ **then**
- 22 $N^{ilk_C} \leftarrow \text{çaprazlama}\{N^{ilk}, N^{ikinci}\}$
- 23 $N^{ikinci_C} \leftarrow \text{çaprazlama}\{N^{ilk}, N^{ikinci}\}$
- 24 N^{ilk_C} ve N^{ikinci_C} tampon stok konfigürasyonlarını Denklem 7.2'yi kullanarak normalize et
- 25 K/P to $1/K$ olasılık ile mutasyon uygula; $N^{ilk_M} \leftarrow \text{mutasyon}\{N^{ilk_C}\}$
- 26 K/P to $1/K$ olasılık ile mutasyon uygula; $N^{ikinci_M} \leftarrow \text{mutasyon}\{N^{ikinci_C}\}$
- 27 Tampon stok konfigürasyonlarının ($Th(N^{ilk_M})$ ve $Th(N^{ikinci_M})$) üretim oranı değerini hesapla
- 28 $N^{geçici} \leftarrow \text{maksimum}\{Th(N^{ilk_M}), Th(N^{ikinci_M})\}$
- 29 **if** $Th(N^{geçici}) > Th(N^{en})$ **then**
- 30 $N^{en} \leftarrow N^{geçici}$
- 31 **if** $Th(N^{ilk_M}) > Th(N^{ilk})$ **then**
- 32 $N^{ilk} \leftarrow N^{ilk_M}$
- 33 **if** $Th(N^{ikinci_M}) > Th(N^{ikinci})$ **then**
- 34 $N^{ikinci} \leftarrow N^{ikinci_M}$
- 35 **if** $l-opt \geq 10K$ **then**
- 36 N^{en} 'e *IncDec* stratejisi uygulandığında elde edilen yeni çözüm N^{veni} olsun
- 37 **if** $Th(N^{veni}) > Th(N^{en})$ **then**
- 38 $N^{en} \leftarrow N^{veni}$
- 39 *IncDec* değerini güncelle
- 40 $yineleme\ sayısı \leftarrow yineleme\ sayısı + 1$
- 41 durdurma kriteri sağlanana kadar

7.3 Büyük Patlama – Büyük Çöküş Algoritması

Popülasyon tabanlı ve yinelemeli bir optimizasyon algoritması olan BP-BÇ, yine Erol ve Eksin (2006) tarafından evrenin evriminden esinlenilerek geliştirilmiştir. BP-BÇ algoritması evrenin büyük patlama ve büyük çöküş aşamalarını taklit eden sezgisel bir algoritmadır. İki aşamadan ilki olan büyük patlama aşamasında, aday çözümler arama uzayında düzgün dağılıma uygun bir şekilde dağıtılırlar. Büyük çöküş aşamasında ise, bu çözümler amaç fonksiyonu dikkate alınarak tek bir çözüme dönüştürülürler. Buradaki dönüştürme (çöküş), uzaya dağılmış aday çözümler arasında ya kütle merkezine ya da amaç fonksiyonunun değerine göre en uygun (minimizasyon problemlerinde uygunluk değeri olarak en küçük, maksimizasyon problemlerinde en büyük değere sahip çözüm) çözümü bulmak olarak tanımlanabilir. Yinelemeli arama boyunca büyük patlama ve büyük çöküş aşamaları art arda uygulanır. Büyük patlama aşamasının çıktılarını büyük çöküş aşaması girdi olarak kullanırken, büyük çöküş aşamasının çıktısını ise büyük patlama aşaması girdi olarak kullanır. Bu adımlar önceden belirlenmiş bir durdurma koşulu sağlanana kadar devam eder (Erol ve Eksin 2006).

Algoritmanın ilk aşaması olan büyük patlama aşamasında, P adet tampon stok konfigürasyonu makine sayısı ve toplam tampon stok sayısını dikkate alarak rastgele oluşturulur. BP-BÇ algoritmasının başlangıç popülasyon büyüklüğü problem boyutuna göre değişmekle birlikte bu tez çalışması kapasamında ele alınan problemler için bu büyüklük $2K-3K$ aralığında belirlenmiştir. Seçilen konfigürasyonların üretim oranı değerleri hesaplandıktan sonra büyük çöküş aşamasına geçilir. Büyük çöküş aşamasında, maksimum üretim oranı değerine sahip olan tampon stok konfigürasyonu Denklem 7.3 ile belirlenir:

$$N_{(m)}^{en} = maks \{Th(N_{(m)}^j)\} \quad (7.3)$$

Denklem 7.3'te, P popülasyondaki birey sayısı olmak üzere, $N_{(m)}^j$ m . yinelemede üretilmiş j . çözüm ($j = 1, 2, \dots, P$), $Th(N_{(m)}^j)$, ilgili tampon stok konfigürasyonunun uygunluk fonksiyonu değerini (üretim oranı) ve $N_{(m)}^{en}$, büyük patlama aşamasındaki

maksimum üretim oranına sahip en iyi tampon stok konfigürasyonunu ifade etmektedir.

Erol ve Eksin (2006) tarafından belirtildiği gibi, bir sonraki yinelemenin büyük patlama aşamasında kullanılacak en iyi tampon stok konfigürasyonu etrafında yeni konfigürasyonların oluşturulması çeşitli şekillerde yapılabilir. Bu tez çalışmasında büyük patlama aşamasında aşağıdaki 3 kurala göre tampon stok konfigürasyonları oluşturulmuştur:

$$N_{(m+1)}^{geçici(j)} = norm \left(N_{(m)}^{en}, r * \frac{N_{(m)}^{en}}{m} \right) \quad (7.4)$$

Kural 1:

$$r * \frac{N_{(m)}^{en}}{m} \text{ tamsayı}$$

$$N_{(m+1)}^{geçici(j)} = norm (N_{(m)}^{en}, IncDec) \quad (7.5)$$

Kural 2:

$$N_{(m+1)}^{geçici(j)} = U \left(\frac{N_{(m)}^{en}}{2}, \left(2 \frac{N}{K-1} \right) \right) \quad (7.6)$$

Kural 3:

$$\frac{2N}{K-1} \text{ ve } \frac{N_{(m)}^{best}}{2} \text{ tamsayılar}$$

Denklem 7.4 – 7.6’da, *norm* normal dağılım operatörünü, r (0,1) arasında rastgele üretilmiş bir sayıyı, *IncDec* standart sapma değeri olarak artırma-azaltma miktarını, m mevcut yineleme sayısını, U ise düzgün dağılımı ifade etmektedir. Kural 1 ve 2 Erol ve Eksin’in (2006) çalışmasından, Kural 3 ise Alataş’ın (2011) çalışmasından uyarlanmıştır. Bu kuralların ayrıntıları hakkında daha fazla bilgi için Erol ve Eksin (2006) ve Alataş (2011)’in çalışmaları incelenebilir.

Kural 1 ve 2’de tampon stoklar normal dağılıma uygun bir şekilde en iyi tampon stok konfigürasyonunun etrafında yeni bireyler üretilmektedir. Kural 1’deki r sayısı, ortalamanın ($N_{(m)}^{en}$) daima standart sapmadan ($r * \frac{N_{(m)}^{en}}{m}$) büyük olmasını sağlar. Algoritmanın başlangıcında m değeri küçük ve standart sapma değeri büyük olduğu için bu kural aday çözümlerin başlangıçtaki çeşitliliğini artırmaktadır. Yinelemeler ilerledikçe m değeri artmakta ve buna bağlı olarak standart sapma değeri derinlemesine arama yapmak için kademeli olarak azalmaktadır.

Kural 2’de *IncDec* değeri standart sapma olarak kullanılmaktadır. *IncDec* değeri mevcut yineleme sayısına göre belirlendiği için m bu denklemde kullanılmamıştır. Küçük boyutlu problemlerin dikkate alındığı durumlarda, aday çözümleri bu formülasyon ile rastgele üretmenin optimum veya en iyi çözüme yakın sonuçlar üretebileceği görülmüştür. Bu gibi durumlarda, bu kural birkaç yinelemede hızlı bir şekilde optimum çözüme ulaşmaktadır.

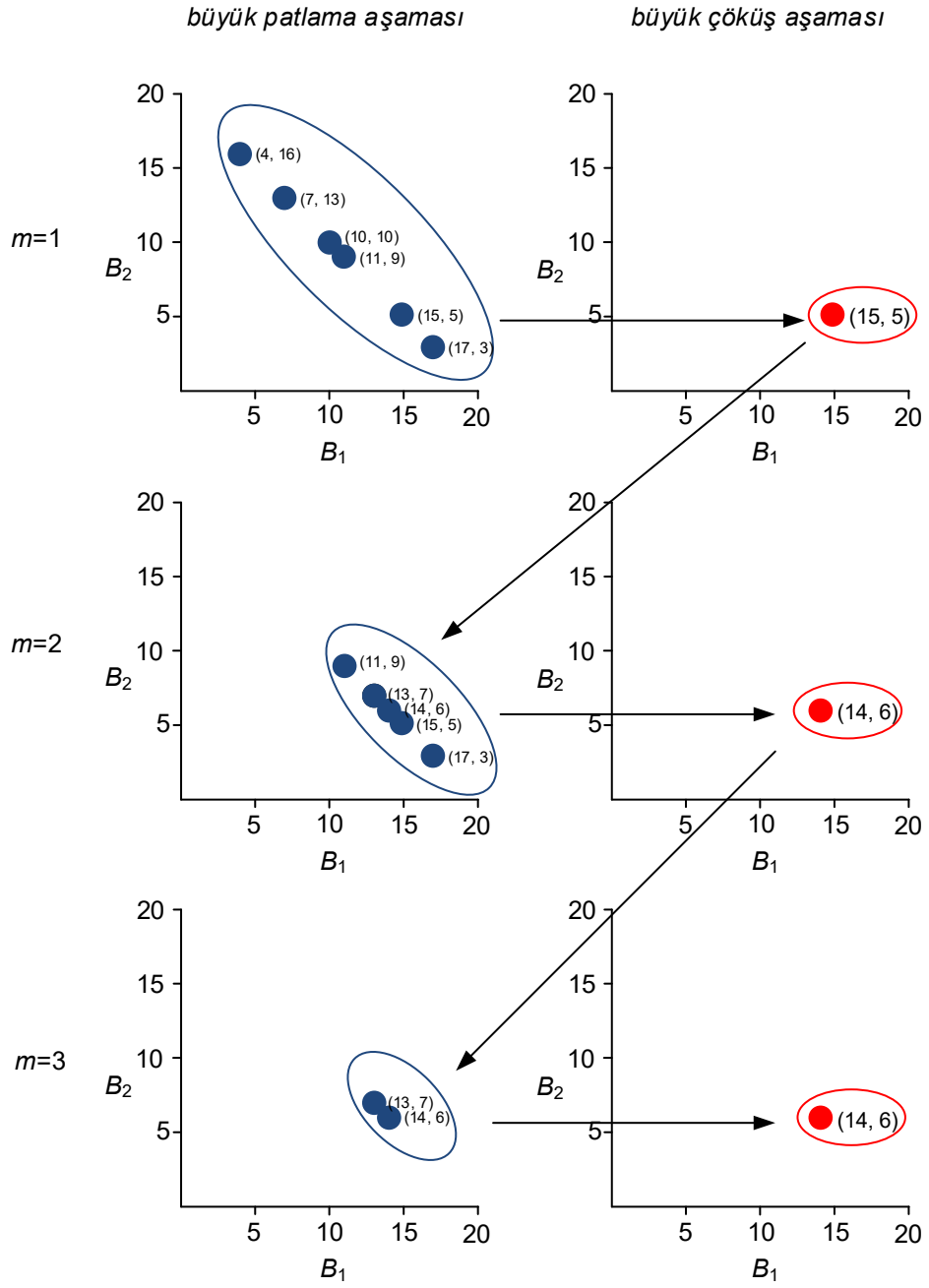
Kural 1 ve 2’den farklı olarak Kural 3’te, aday çözümler düzgün dağılım kullanılarak üretilmektedir. Alataş (2011)’in belirttiği gibi, düzgün dağıtılmış aday çözümü içeren iyi dağıtılmış bir popülasyonu hızlandırabileceğinden, bu kural BP-BÇ algoritmasının büyük patlama aşamasında alternatif üçüncü kural olarak dikkate alınmıştır. Bir üretim hattındaki makinelerin güvenilirlik parametreleri aynıysa ya da benzerse, düzgün dağıtılmış tampon stok konfigürasyonlarının daha iyi üretim oranı değerleri ürettiği bilinmektedir. Bu bilgi, bu kuralı uygulamak için başka bir nedendir.

Algoritmanın büyük patlama aşamasında uygulanacak kural rulet tekeri yöntemi kullanılarak rastgele belirlenmektedir (Denklem 6.9). Seçilen kural mevcut (en iyi) tampon stok konfigürasyonundaki tüm tampon stoklara uygulanmaktadır. Büyük patlama aşamasında yeni tampon stok konfigürasyonu oluştururken, tampon stok değerleri tamsayı değerlere yuvarlanmaktadır. En iyi üretim oranı değerini veren tampon stok konfigürasyonu, bir sonraki yineleme için mevcut çözüm olarak seçilir. En iyi çözüm belirli sayıda yineleme (K) için değişmezse, yerel optimumdan kaçmak amacıyla SGA’da olduğu gibi *IncDec* stratejisi (Algoritma 5) uygulanır. *IncDec* stratejisi üç kez uygulandıktan sonra en iyi çözüm değişmezse, *IncDec* stratejisi bir kez daha kullanılmaz. Algoritma $10K$ yinelemeden sonra sonlandırılır. Sonlandırma kriteri için belirtilen yineleme sayısı yapılan pilot deneylerle belirlenmiştir. TSDP’nin çözümü için önerilen BP-BÇ algoritmasının genel çerçevesi Algoritma 11’de verilmiştir. Şekil 7.3’te ise 3 makineli bir hatta dağıtılacak 20 adet toplam tampon stokun olduğu bir örnek için 3 yineleme boyunca BP-BÇ algoritmasının yakınsama davranışı gösterilmiştir (veriler için bkz. Bölüm 7.4.2).

Algoritma 11: Üretim Oranı Maksimizasyonu İçin Önerilen BP-BÇ Algoritması

girdi : Makine sayısı K , toplam tampon stok miktarı N , popülasyon boyutu P , $IncDec$ değeri
çıkıtı : Maksimum üretim oranına sahip en iyi tampon stok konfigürasyonu N^{en}

- 1 Yerel optimum için izin verilen maksimum yineleme sayısını ($l-opt$) K olarak ayarla
- 2 İzin verilen maksimum yineleme sayısını $10K$ olarak ayarla
- 3 Problem boyutuna göre $IncDec$ değerini ayarla
- 4 $N^{geçici} \leftarrow (0, 0, \dots, 0) \leftarrow (N_1^{geçici}, N_2^{geçici}, \dots, N_{K-1}^{geçici})$ olsun
- 5 $N^{en} \leftarrow N^{eni} \leftarrow N^{geçici}$
- 6 Rastgele P adet tampon stok konfigürasyonu üret
- 7 **tekrar et**
 - 8 Her tampon stok konfigürasyonunun üretim oranı değerini hesapla
 - 9 Maksimum üretim oranına sahip tampon stok konfigürasyonunu (N^{en}) Denklem (7.3)'ü ile bul
 - 10 P adet yeni aday çözümü Denklem (7.4-7.6)'yı kullanarak üret
 - 11 Aday çözümleri normalize et
 - 12 **if** $l-opt \geq K$ **then**
 - 13 N^{en} 'e $IncDec$ stratejisi uygulandığında elde edilen çözüm N^{eni} olsun
 - 14 **if** $Th(N^{eni}) > Th(N^{en})$ **then**
 - 15 $N^{en} \leftarrow N^{eni}$
 - 16 $IncDec$ değerini güncelle
 - 17 $yineleme\ sayısı \leftarrow yineleme\ sayısı + 1$
 - 18 durdurma kriteri sağlanana kadar



Şekil 7.3: BP-BÇ algoritması: 3 makineli hat örneği

7.4 Deneysel Çalışmalar

Bu bölümde, önerilen çözüm algoritmalarının performansları literatürden alınan karşılaştırma problemleri üzerinde test edilmiştir. Her iki algoritma da MATLAB (R2016a)'da kodlanmıştır. Tüm deneyler, 8 GB RAM ve 3.40 GHz Intel (R) i7 işlemciye sahip bir bilgisayarda gerçekleştirilmiştir. Bu bölümde sunulan

deneysel çalışma iki aşamada yürütülmüştür. İlk aşamada parametre optimizasyonu yapılarak her algoritma için algoritmalarda kullanılan parametrelerin optimum değerleri belirlenmiştir. Daha sonra, önerilen algoritmaların performanslarını değerlendirmek için karşılaştırma problemleri kullanılmıştır. Karşılaştırmada kullanılan literatürdeki yöntemler için sadece ilgili makalelerde raporlanan sonuçlar açısından değerlendirme yapılmıştır.

7.4.1 Parametre Optimizasyonu

Meta-sezgisel algoritmaların performansının algortmada kullanılan parametre değerlerine bağlı olduğu bilinmektedir (Joshi ve Bansal 2020). Optimum parametre değerleriyle bir arama algoritmasını çalıştırmak algoritmanın performansını artırmaktadır (Mobin ve diğ. 2018). Bu nedenle deneysel çalışmanın ilk aşamasında optimum parametre değerlerini bulmak için pilot bir deney gerçekleştirilmiştir. Önerilen algoritmaların performansına etkisi olan en önemli parametreler popülasyon büyüklüğü ve IncDec değeridir. Problem büyüklükleri dikkate alınarak her bir parametre için üç farklı değer belirlenmiştir (Tablo 7.1). Önerilen algoritmalar, her karşılaştırma problemi için 10 defa çalıştırılmıştır. Toplam 4500 çalıştırma sonucunda, her bir karşılaştırma problemi için parametrelerin optimum değerleri Tablo 7.1'deki gibi elde edilmiştir.

Tablo 7.1: SGA ve BP-BÇ algoritmasının optimum parametre değerleri

Problem (K, N)	Popülasyon büyüklüğü (P)	IncDec Değeri
(3, 20)	$K, 2K^{*\S}, 3K$	$1^{*\S}, 2, 3$
(5, 31)	$K, 2K^{*\S}, 3K$	$1^{*\S}, 2, 3$
(10, 270)	$K, 2K^*, 3K^{\S}$	$1, 2^{*\S}, 3$
(20, 100)	$K, 2K^*, 3K^{\S}$	$1^{*\S}, 2, 3$
(40, 200)	$2K, 3K^*, 4K^{\S}$	$1^{*\S}, 2, 3$
(20, 400)	$2K^*, 3K, 4K^{\S}$	$2^*, 3^{\S}, 4$
(40, 400)	$2K, 3K^{*\S}, 4K$	$2^{*\S}, 3, 4$
(40, 800)	$2K, 3K^*, 4K^{\S}$	$3^{*\S}, 4, 5$
(40, 1600)	$3K^*, 4K, 5K^{\S}$	$3, 4^{*\S}, 5$

\S SGA için en iyi parametre değerleri

* BB-BC için en iyi parametre değerleri

7.4.2 Karşılaştırma Problemleri İçin Deneyler

Bu bölümde, karşılaştırma problemlerinin ayrıntıları sunulmuş ve deneysel hesaplama sonuçları tartışılmıştır.

3 makineli hat: Bu bölümde ele alınan ilk karşılaştırma problemi Gershwin ve Schor (2000) tarafından önerilmiştir. Probleme dikkate alınan üretim hattı 3 makineden ve toplam 20 tampon stoktan oluşmaktadır. Tablo 7.2’de 3 makineli üretim hattına ait güvenilirlik parametreleri verilmiştir.

Tablo 7.2: 3 makineli hattın güvenilirlik parametreleri (Gershwin ve Schor 2000)

Makine (i)	1	2	3
β_i	0,037	0,015	0,02
r_i	0,35	0,15	0,4

Problemin optimum çözümü 0,86799 üretim oranı ile {14, 6} tampon stok konfigürasyonu ile elde edilmektedir. Probleme ait karşılaştırmalı sonuçlar Tablo 7.3’te verilmiştir. Önerilen algoritmaların her ikisi de optimum çözüme 10 tekrarın tamamında ulaşmıştır. Önerilen SGA optimum çözüme 1,4 saniyede ulaşırken BP-BÇ algoritması 0,6 saniyede aynı çözümü elde etmiştir. Ancak SGA, BP-BÇ algoritması ile karşılaştırıldığında daha az değerlendirme sayısı ile optimum çözüme ulaşmıştır.

Tablo 7.3: 3 makineli hattın karşılaştırmalı sonuçları

Yöntem	Ort. /En İyi Üretim Oranı	Ort. Değ. Sayısı	En İyi Çözüm Say.	Ort. CPU Süresi
Gershwin ve Schor (2000)	0,86799	-	-	-
SGA	0,86799	8,8	10	1,4
BP-BÇ	0,86799	10,8	10	0,6

5 makineli hat: Güvenilirlik parametreleri Tablo 7.4’te verilen bu problem Ho ve diğ. (1979) tarafından önerilmiştir. Daha sonra Gershwin ve Schor (2000) tarafından da kullanılan problemin çözümü için her iki çalışmada da gradyan tabanlı arama sezgiselleri kullanılmıştır. Probleme her tampon stok alanı için alt sınır 4 ve toplam tampon stok miktarı 31 olarak belirlenmiştir. Sonuçlar Tablo 7.5’te sunulmuştur.

Tablo 7.4: 5 Makineli hattın güvenilirlik parametreleri (Ho ve diğ. 1979)

Makine (<i>i</i>)	1	2	3	4	5
$1/r_i$	11	19	12	7	7
$1/\beta_i$	20	167	22	22	26

Problemin optimum sonucu {7, 10, 10, 4} tampon stok konfigürasyonu ile 0,4943'tür. Tablo 7.5'te görüldüğü gibi optimum sonuca SGA ile 2,4 saniyede, BP-BÇ algoritması ile 1,4 saniyede ulaşılmıştır. Optimum çözüme ulaşmak için gereken ortalama değerlendirme sayıları ise SGA için 28,4, BP-BÇ algoritması içinse 25'tir.

Tablo 7.5: 5 makineli hattın karşılaştırmalı sonuçları

Yöntem	Ort. / En İyi Üretim Oranı	Ort. Değ. Sayısı	En İyi Çözüm Say.	Ort. CPU Süresi
Ho ve diğ. (1979)	0,4914	-	-	-
Gershwin ve Schor (2000)	0,4943	-	-	-
SGA	0,4943	28,4	10	2,4
BP-BÇ	0,4943	25,0	10	1,4

Bu küçük boyutlu örnekler ile önerilen algoritmaların optimum çözümü bulma kapasiteleri gösterilmiştir. Ancak, bu sonuçları genelleştirmek için önerilen algoritmalar büyük boyutlu problemlerin çözümünde kullanılmıştır.

10 makineli hat: İlk defa Nahas ve diğ. (2006) tarafından önerilen bu probleme ait güvenilirlik parametreleri Tablo 7.6'da verilmiştir. Kullanılacak toplam tampon stok sayısı 270'tir. Problem daha sonra Demir ve diğ. (2011) tarafından da ele alınmıştır. Bu problemin çözümü için Nahas ve diğ. (2006) tarafından İTA, Demir ve diğ. (2011) tarafından DTL-TS isimli bir TA algoritması önerilmiştir.

Tablo 7.6: 10 makineli hattın güvenilirlik parametreleri (Nahas ve diğ. 2006)

Makine (<i>i</i>)	1	2	3	4	5	6	7	8	9	10
$1/r_i$	7	7	5	10	9	14	5	8	10	10
$1/\beta_i$	20	33	22	22	25	40	23	30	45	20

Tablo 7.7'de karşılaştırmalı sonuçlar sunulmuştur. Tablodan da görülebileceği üzere, SGA ve BP-BÇ algoritması {14, 20, 30, 53, 45, 26, 23, 25, 34} tampon stok konfigürasyonu ile 0,64346 üretim oranı değerine ulaşmıştır. Önerilen algoritmaların her ikisi de en iyi çözüme 10 tekrarın tamamında ulaşmıştır. SGA'da en iyi çözüme ulaşmak için gereken ortalama yineleme sayısı 350 iken, BP-BÇ algoritması için bu

sayı 19'dur. Bu örnek önerilen BP-BÇ algoritmasının diğer algoritmalara kıyasla daha az sayıda yineleme ile problemi çözdüğünü göstermiştir. Ayrıca, önerilen BP-BÇ algoritmasının hem çözüm süresi hem de yineleme sayısı bakımından önerilen SGA'dan daha iyi olduğu görülmüştür.

Tablo 7.7: 10 makineli hattın karşılaştırmalı sonuçları

Yöntem	En İyi Üretim Oranı	Ort. Yineleme Sayısı	Ort. CPU Süresi
Nahas ve diğ. (2006)	0,64135	14000	1120
Demir ve diğ. (2011)	0,64135	78	-
SGA	0,64346	350	75,6
BP-BÇ	0,64346	19	35,4

Büyük boyutlu problemler-1: Önerilen SGA ve BP-BÇ algoritmasının performansları Demir ve diğ. (2011) tarafından önerilen 20 ve 40 makineli problem setleri üzerinde test edilmiştir. Toplam tampon stok miktarı 20 makineli hatlar için 100, 40 makineli hatlar içinse 200 olarak belirlenmiştir. Problemin çözümünde yazarlar, klasik TA algoritmasını DTL-TS adını verdikleri ve yazarlar tarafından geliştirilen başka bir TA algoritması ile karşılaştırmışlardır. Bu problem setleri, Tablo 7.8 ve 7.9'da gösterildiği gibi dokuz alt problemi içermektedir. Her problem ve alt problem seti için 10 tekrar yapılmış ve ortalama üretim oranı değerleri hesaplanmıştır.

Tablo 7.8: 20 makineli hatların karşılaştırmalı sonuçları

Problem Seti	Makine Parametreleri		Üretim Oranı			DTL-TS'ye Göre İyileşme (%)	
	β_i	r_i	DTL-TS	SGA	BP-BÇ	SGA	BP-BÇ
1	0,1	0,1	0,234191	0,237285	0,239860	1,32	2,42
2	0,2	0,2	0,297025	0,304254	0,306625	2,43	3,23
3	0,3	0,3	0,334450	0,344769	0,346603	3,09	3,63
4	0,4	0,4	0,359637	0,376133	0,378570	4,59	5,26
5	0,5	0,5	0,377895	0,401535	0,402820	6,26	6,60
6	0,6	0,6	0,391846	0,423280	0,424286	8,02	8,28
7	0,7	0,7	0,402760	0,435383	0,437793	8,10	8,70
8	0,8	0,8	0,411638	0,457630	0,459787	11,17	11,70
9	0,9	0,9	0,419018	0,477286	0,478804	13,91	14,27
<i>Ortalama</i>						6,54	7,12

Tablo 7.9: 40 makineli hatların karşılaştırmalı sonuçları

Problem Seti	Makine Parametreleri		Üretim Oranı			DTL-TS'ye Göre İyileşme (%)	
	β_i	r_i	DTL-TS	SGA	BP-BÇ	SGA	BP-BÇ
1	0,1	0,1	0,221273	0,222362	0,224404	0,49	1,41
2	0,2	0,2	0,282169	0,289244	0,293719	2,51	4,09
3	0,3	0,3	0,325256	0,333629	0,340238	2,57	4,61
4	0,4	0,4	0,351494	0,367044	0,371684	4,42	5,74
5	0,5	0,5	0,370666	0,396157	0,399711	6,88	7,84
6	0,6	0,6	0,385328	0,420090	0,422613	9,02	9,68
7	0,7	0,7	0,397255	0,431178	0,433123	8,54	9,03
8	0,8	0,8	0,406559	0,453735	0,455542	11,60	12,05
9	0,9	0,9	0,413990	0,470600	0,471823	13,67	13,97
<i>Ortalama</i>						6,63	7,60

Önceki örnekte olduğu gibi, önerilen algoritmalar DTL-TS algoritmasından daha yüksek üretim oranı değerleri elde etmiştir. Ayrıca, BP-BÇ algoritması ele alınan tüm problem setleri için hem DTL-TS hem de SGA'dan çözüm kalitesi açısından daha iyi sonuçlar üretmiştir. Önerilen SGA ile elde edilen amaç fonksiyonu değerleri DTL-TS algoritmasına göre 20 makineli hat için ortalama %6,54, 40 makineli hat içinse ortalama %6,63 daha iyidir. Önerilen BP-BÇ algoritması için amaç fonksiyonundaki iyileşme oranları sırasıyla %7,12 ve %7,60'tır. SGA için optimum çözüme ulaşmak için gereken çözüm süresi 20 makineli hatlar için 92,3-103,4 sn. arasında, 40 makineli hatlar içinse 180,9-191,8 sn. arasındadır. Öte yandan, BP-BÇ algoritmasında bu süreler 20 makineli hatlar için 83,9-105,6 sn. arasındayken, 40 makineli hat için 544,7-585,2 sn. arasındadır.

Bu örnek, önerilen algoritmaların TSDP'yi makul bir hesaplama süresi içinde çözme kapasitesini göstermiştir. Ayrıca, önerilen algoritmaların performanslarının aynı güvenilirlik parametrelerine sahip makineleri içeren büyük boyutlu hatlar için daha iyi olduğu görülmüştür. Önerilen BP-BÇ algoritması çözüm kalitesi açısından önerilen SGA ile karşılaştırıldığında ise BP-BÇ algoritmasının daha iyi olduğu da bu deney sonuçlarından açıkça görülmektedir.

Büyük boyutlu problemler-2: Büyük boyutlu hatları dikkate alan bu problemde, önerilen SGA ve BP-BÇ algoritmasının performansları farklı güvenilirlik parametrelerine sahip 20 ve 40 makineli hatlar için test edilmiştir. Problem, 20

makineli hatlar için toplam tampon stok miktarı 400 kısıtı altında, 40 makineli hatlar içinse 400, 800 ve 1600 olarak belirlenmiş toplam tampon stok miktarları için çözülmüştür. Bu problem seti Li ve diğ. (2015) tarafından önerilmiştir. Li ve diğ. (2015) problemin çözümü için FA olarak isimlendirdikleri sezgisel bir algoritma önermişlerdir. 20 makineli hattın güvenilirlik parametreleri Tablo 7.10'da verilmiştir. 40 makineli hat, 20 makineli hattın tekrar edilmesiyle oluşturulmuştur.

Tablo 7.10: 20 makineli hattın güvenilirlik parametreleri (Li ve diğ. 2015)

Makine (<i>i</i>)	1	2	3	4	5	6	7	8	9	10
β_i	0,011	0,005	0,022	0,025	0,052	0,021	0,017	0,013	0,011	0,016
r_i	0,039	0,016	0,166	0,085	0,186	0,105	0,116	0,136	0,043	0,139
Makine (<i>i</i>)	11	12	13	14	15	16	17	18	19	20
β_i	0,019	0,014	0,013	0,025	0,014	0,001	0,006	0,013	0,059	0,038
r_i	0,074	0,099	0,111	0,178	0,094	0,015	0,061	0,046	0,184	0,134

Önerilen SGA ve BP-BÇ algoritması ile elde edilen sonuçlarla FA algoritmasının sonuçları karşılaştırmalı olarak Tablo 7.11'de sunulmuştur. Tablo 7.11'de görüldüğü gibi, önerilen BP-BÇ algoritması FA'ya göre her test problemi için üretim oranı açısından ortalama olarak sırasıyla %1,51, %1,09, %3,00 ve %3,29 daha iyi sonuçlar üretmiştir. Öte yandan, önerilen SGA her test problemi için FA'ya kıyasla sırasıyla %1,23, %0,58, %2,67 ve %1,98 daha iyi sonuçlar üretmiştir. Tablo 7.11, önerilen BP-BÇ algoritmasının çözüm kalitesi açısından problemin çözümünde kullanılan algoritmalar arasında en iyi çözümü ürettiğini göstermektedir.

Önerilen BP-BÇ algoritmasının performansı çözüm süresi açısından değerlendirildiğinde, en iyi çözümü elde etmek için gereken sürenin 20 makineli hat için ortalama 161,2 sn. olduğu ve 40 makineli hat için ise ortalama 886,2-1127,5 sn. arasında olduğu görülmektedir. Önerilen SGA'nın çözüm süresi ise 20 makineli hat için ortalama 167,3 sn. iken 40 makineli hat için ise ortalama 302,4-418,9 sn. aralığındadır (Tablo 7.11). Öte yandan, Li ve diğ. (2015) tarafından FA'nın çözüm süresinin birkaç saniye olduğu belirtilmiştir. Çözüm süresindeki bu farklılığın bir nedeni, kullanılan ayrıştırma yöntemindeki farklılıktır. 5. bölümdeki deneysel çalışmalarda da belirtildiği üzere Li ve diğ. (2015), üretim oranı değerlendirmelerinin sayısını önemli ölçüde azaltan üç makineli-iki tampon stok alanlı bir ayrıştırma yöntemi önermişler ve FA algoritması ile beraber kullanmışlardır.

Önerilen BP-BÇ algoritması, en iyi çözümü elde etmek için daha fazla zaman harcarsa bile, önerilen SGA'ya kıyasla daha az sayıda tampon stok konfigürasyonu değerlendirmesi yaptığı görülmektedir. Bunun nedeni, SGA'da uygulanan IncDec stratejisi sayısının BP-BÇ algoritmasında uygulananından daha fazla olmasıdır. Tablo 7.12, her problem seti için uygulanan ortalama IncDec stratejisi sayısını ve ayrıca IncDec stratejisi uygulandıktan sonra üretim oranı değerindeki ortalama iyileşmeyi göstermektedir. Tablo 7.12'den görülebileceği gibi, SGA, özellikle büyük boyutlu problem setleri için, BP-BÇ algoritmasından daha fazla yerel optimumda takılmıştır. Ayrıca, IncDec stratejisinin uygulanması, SGA için üretim oranı değerinde önemli iyileşme sağlamıştır.

Tablo 7.11: Büyük boyutlu hatların karşılaştırmalı sonuçları

Problem Seti	(K, N)	FA (Li ve diğ. 2015)		SGA			BP-BÇ			
		En İyi Üretim Oranı	Ort. Üretim Oranı	Ort. Değ. Sayısı	Ort. İyileş. (%) *	Ort. CPU Süresi	Ort. Üretim Oranı	Ort. Değ. Sayısı	Ort. İyileş. (%) **	Ort. CPU Süresi
1	(20, 400)	0,662264	0,670416	3125,5	1,23	167,3	0,672237	2438,5	1,51	161,2
2	(40, 400)	0,571864	0,575174	5881,9	0,58	302,4	0,578075	4113,3	1,09	886,2
3	(40, 800)	0,651023	0,668432	6140,2	2,67	331,8	0,670524	4580,7	3,00	945,6
4	(40, 1600)	0,707086	0,721056	6565,8	1,98	418,9	0,730370	5043,6	3,29	1127,5

* FA'ya kıyasla üretim oranında ortalama iyileşme (%) = $100 \times [Th(SGA) - Th(FA)] / Th(FA)$

** FA'ya kıyasla üretim oranında ortalama iyileşme (%) = $100 \times [Th(BP-BÇ) - Th(FA)] / Th(FA)$

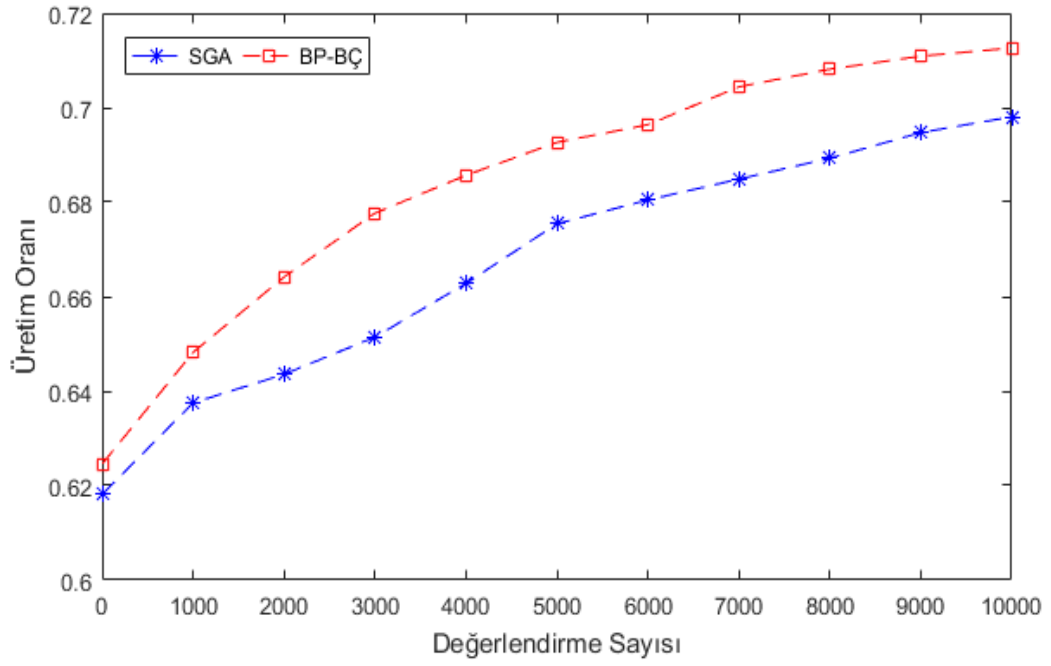
Tablo 7.12: IncDec stratejisinin önerilen algoritmalar üzerindeki etkisi

Problem Seti (K, N)	SGA		BP-BÇ	
	Ort. Uygulama Sayısı	Ort. İyileşme (%)	Ort. Uygulama Sayısı	Ort. İyileşme (%)
(3, 20)	0	0	0	0
(5, 31)	0	0	0	0
(10, 270)	1	1,53	0	0
(20, 100)	1	3,46	0	0
(40, 200)	0	0	0	0
(20, 400)	2	2,71	1,8	2,64
(40, 400)	2	2,85	1,5	1,29
(40, 800)	2	2,58	1,6	2,53
(40, 1600)	2,2	2,41	1,7	2,24

Bu örnekten, önerilen BP-BÇ algoritmasının çözüm kalitesi açısından diğer algoritmalarından daha iyi olduğu sonucuna varılabilir. Ayrıca, TSDP bir tasarım problemi olduğu göz önünde bulundurulduğunda önceki bölümlerde de belirtildiği gibi makul bir sürede iyi çözümler elde etmek önemlidir. Dolayısıyla önerilen BP-BÇ algoritmasının TSDP için iyi kalitede çözüm sağlamada etkili bir algoritma olduğu söylenebilir. Ayrıca, en iyi çözümü elde etmek için gereken CPU süresinin hem SGA hem de BP-BÇ algoritması için makul olduğunu söylemek mümkündür.

Önerilen her iki algoritma çözüm süresi açısından karşılaştırıldığında ise önerilen SGA'nın büyük boyutlu problem setleri için BP-BÇ algoritmasından daha iyi olduğu görülmektedir. Bunun nedeni ise BP-BÇ algoritmasının her yinelemede SGA'dan daha fazla tampon stok konfigürasyonu değerlendirmesidir. Diğer yandan önerilen algoritmalarla bilinen en iyi çözüme göre elde edilen ortalama iyileşme önceki büyük boyutlu hatlara göre daha küçüktür. Bu örnekten, önerilen SGA ve BP-BÇ algoritmasının performanslarının problem boyutu arttıkça azaldığı sonucuna varılabilir. Bu dezavantajın üstesinden gelmek ve arama alanını etkili bir şekilde çeşitlendirmek için, her iki algoritmaya da farklı arama stratejileri dâhil edilebilir.

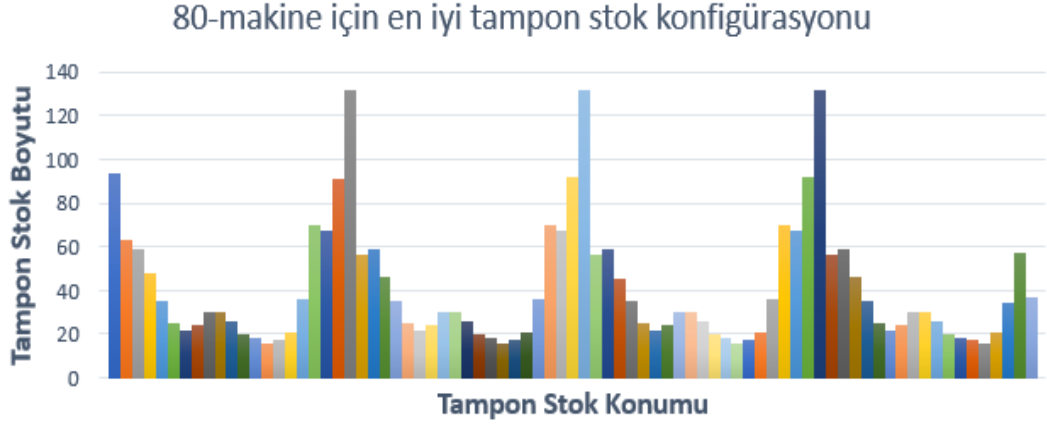
Her yinelemede değerlendirilen tampon stok konfigürasyonlarının sayısı önerilen algoritmalar için farklı olduğundan, algoritmaların performansları maksimum 10000 değerlendirmede elde edilen üretim oranı değerleri 80 makine ve 3200 tampon stok içeren bir üretim hattı için karşılaştırılmıştır (Şekil 7.4). Şekil 7.4'te görüldüğü gibi, önerilen BP-BÇ algoritması, aynı sayıda değerlendirme için önerilen SGA'dan daha yüksek üretim oranı değerine ulaşmıştır.



Şekil 7.4: 10000 değerlendirme için SGA ve BP-BÇ algoritmasının karşılaştırması

Genel olarak, her iki algoritma popülasyon tabanlı ve algoritmaların temel mantığı benzer olsa da BP-BÇ algoritması bu bölümde sunulan deneysel çalışmalardan da anlaşılacağı üzere çözüm kalitesi açısından SGA'dan daha iyidir. Bunun nedenlerinden biri, BP-BÇ algoritmasının her yinelemede en iyi tampon stok konfigürasyonunun kalıtımını kullanarak yeni çözümler üretmesidir. Diğer bir neden ise, algoritmanın başında çeşitlendirme fazla iken, algoritma ilerledikçe BP-BÇ algoritmasının yoğunlaştırma özelliğinin ön plana çıkmasıdır. BP-BÇ algoritmasının bu özellikleri, SGA ile karşılaştırıldığında yerel optimuma takılmama şansını arttırmaktadır.

Önerilen BP-BÇ algoritması kullanılarak 10000. değerlendirmede elde edilen en iyi tampon stok konfigürasyonu Şekil 7.5'te verilmiştir. Darboğaz makineler hattın etkinliğini önemli ölçüde etkiler (Staley ve Kim 2012). Bu nedenle, tampon stoklar Şekil 7.5'te görülebileceği gibi darboğaz makinelerin etrafına daha fazla atanmıştır. Şekil 7.5'te, 21., 41. ve 61. tampon stok alanları en yüksek tampon stok değerlerine sahiptir ve tampon stok dağılımı simetriktir.



Şekil 7.5: 80 makineli hat için önerilen BP-BÇ algoritmasıyla elde edilen en iyi tampon stok konfigürasyonu

Bu bölümde sunulan deneysel sonuçlar, önerilen BP-BÇ algoritmasının literatürde TSDP için önerilen diğer algoritmalar ve SGA ile aynı veya daha iyi sonuçlar verdiğini göstermektedir. Elde edilen üretim oranı değerleri arasındaki farklar küçük görüldüğünden, üretim oranı değerlerindeki farklılıkların istatistiksel olarak anlamlı olup olmadığını belirlemek için Friedman testi kullanılmıştır. Aşağıdaki H_0 hipotezi $\alpha=0,05$ anlamlılık düzeyinde test edilmiştir:

H_0 : Bütün algoritmalar ortalama olarak aynı üretim oranı değerini üretir.

Tablo 7.13: Friedman testi

Sıralar		Test İstatistikleri	
Yöntemler	Ortalama Sıra	N (Problem seti)	25
Literatürdeki en iyi	1,08	Ki-kare	45,516
SGA	2,02	Serbestlik derecesi	2
BP-BÇ	2,90	Anlamlılık (p)	0,000

Friedman testinin sonucu Tablo 7.13'te sunulmuştur. Tablo 7.13'te 0,000'lık p -değeri, üretim oranı değerleri arasındaki farkların %95 güven düzeyinde istatistiksel olarak anlamlı olduğunu göstermektedir. Bu bölümde sunulan tüm deneyler göz önüne alındığında, özetle önerilen BP-BÇ algoritmasının, güvenilir olmayan dengeli üretim hatlarında TSDP'yi çözme konusunda büyük bir kabiliyete sahip olduğu söylenebilir.

7.5 Bulgular ve Tartışma

Bu bölümde TSDP'yi üretim oranı maksimizasyonu amacı ile çözmek için önerilen SGA ve BP-BÇ algoritmasının, dikkate alınan tüm karşılaştırma problemleri için literatürdeki diğer yöntemlerle aynı veya daha iyi sonuçlar verdiği görülmüştür. Ayrıca önerilen algoritmaların hesaplama karmaşıklığı, K makine sayısı olmak üzere, ortalama değerlendirme sayısı açısından $O(K^3)$ olarak hesaplanmıştır.

Bu çalışmada, önerilen SGA ve BP-BÇ algoritmasının etkinliğini kanıtlamak için hipotetik üretim hatları dikkate alınmıştır. Bununla birlikte, önerilen algoritmaların büyük boyutlu problemleri çözümedeki başarısı dikkate alındığında algoritmaların gerçek hayattaki üretim problemlerini çözmek için büyük bir potansiyele sahip olduğu görülmektedir. Bu bağlamda, önerilen algoritmaların en büyük dezavantajı, en iyi çözümü elde etmek için gereken CPU süresidir. Önerilen BP-BÇ algoritmasının her yinelemede birden fazla çözümü değerlendirmesi algoritmanın çözüm süresini arttırmaktadır. Bu dezavantajın üstesinden gelmek için önerilen algoritma, diğer arama algoritmaları ile melezleştirilebilir.

7.6 Sonuç

Bu bölümde TSDP'nin çözümü için popülasyon tabanlı iki evrimsel algoritma önerilmiş ve önerilen algoritmaların performansı literatürdeki mevcut karşılaştırma problemleri üzerinde test edilmiştir. Deneysel sonuçlar, önerilen BP-BÇ algoritmasının önerilen SGA ve literatürdeki mevcut karşılaştırma problemleri için önerilen diğer algoritmalara kıyasla daha iyi kalitede çözümler ürettiğini göstermiştir. Problem boyutu arttıkça önerilen algoritmaların performanslarının düştüğü görülmüştür. Bu dezavantajın üstesinden gelmek için, önerilen algoritmalara farklı arama stratejileri entegre edilebileceği veya önerilen algoritmaların performansının diğer sezgisel arama yöntemleriyle melez bir şekilde kullanarak artırılacağı düşünülmektedir.

8. SONUÇ VE ÖNERİLER

8.1 Tezin Özeti

Tampon stok dağılımı problemi üretim hattı tasarımcılarının karşılaştığı, hattın performansını artırmak için tampon stokların üretim hattındaki makineler arasına nasıl dağıtılacağını içeren başlıca optimizasyon problemlerinden biridir. Karmaşıklığı ve önemi dolayısıyla hem akademi hem de imalat endüstrisinde bu alanla ilgili çok fazla çalışma yapılmıştır.

Seri üretim hatları imalat endüstrisinde en çok karşılaşılan ve genellikle elektronik ürün, otomobil ve gıda ürünleri gibi yüksek hacimli tek tip ürün üreten hatlardır. Seri üretim hatlarının performansları makine bozulmaları ve stokastik işlem süreleri gibi parametrelerden etkilenmektedir. Tampon stok kullanımıyla bu hatlarda makinelerin birbirinden bağımsız olarak çalışması ve makine duruşlarından kaynaklı meydana gelen âtil sürenin azaltılması amaçlanmaktadır.

TSDP alanındaki çalışmalar çoğunlukla ele alınan üretim hattı topolojisine göre bir optimizasyon algoritması geliştirmeye odaklanmıştır. TSDP gibi bir kombinatoriyal optimizasyon problemini çözmek için çok sayıda algoritma önerilmiş olsa da en uygun olan algoritmayı bulma sorusu tatmin edici bir şekilde cevaplanmamıştır. TSDP literatüründe yapılan çalışmalara bakıldığında, genel olarak iki temel amaç fonksiyonunun ele alındığı görülmektedir: verilen sınırlı bir tampon stok miktarı için üretim oranı maksimizasyonu ve belirli bir üretim oranının altına düşmeyecek şekilde toplam tampon stok miktarı minimizasyonu. Problemi bu amaç fonksiyonları ile etkin bir şekilde çözecek, çeşitlilik ve yakınsama açısından güçlü optimizasyon algoritmalarının araştırılması halen literatürde önemli bir araştırma alanıdır.

Bu tez çalışmasında:

- Makinelerin bozulmaya maruz kaldığı güvenilir olmayan dengeli seri üretim hatlarına odaklanılmıştır.

- DKA, ABKA, SGA ve BP-BÇ algoritmaları literatürde ilk kez TSDP'yi üretim oranı maksimizasyonu amacı ile çözmek için önerilmiştir. DKA algoritması ayrıca, toplam tampon stok miktarı minimizasyonu problemini çözmek için literatürdeki bir ikili arama yöntemiyle birlikte kullanılmıştır.
- DKA algoritmasındaki komşuluk yapıları klasik DKA algoritmasından farklı olarak, arama etkinliğini arttırmak için adaptif bir şekilde kullanılmıştır.
- ABKA algoritmasını TSDP'nin çözümünde kullanabilmek için iki yeni çıkarma-ekleme operatörü önerilmiştir.
- Önerilen komşuluk tabanlı arama algoritmalarının arama eforunu azaltmak için TDSP literatürüne özgü üç farklı başlangıç çözümü oluşturma sezgiseli önerilmiştir.
- SGA ve BP-BÇ algoritması sürekli karşılaştırma fonksiyonları yerine ilk defa bir üretim hattı tasarım problemine uyarlanmıştır.
- Önerilen SGA'da iki farklı mutasyon operatörü adaptif bir şekilde kullanılırken, orijinal SGA'dan farklı olarak, iki noktalı çaprazlama operatörü kullanılmıştır.
- BP-BÇ algoritması TSDP'nin çözümü için uyarlanırken üç farklı popülasyon üretme kuralı adaptif bir şekilde kullanılmıştır.
- Popülasyon tabanlı olan SGA ve BP-BÇ algoritması, TSDP literatüründe çoğunlukla simülasyonla beraber kullanılan popülasyon tabanlı algoritmaların aksine analitik bir değerlendirme algoritmasıyla birlikte kullanılmıştır.
- Ayrıca çözüm uzayında aramanın yoğunlaştırılması için bir yerel arama sezgiseli geliştirilmiş ve tez kapsamında önerilen tüm algoritmaların ihtiyaç duyulan aşamalarında kullanılmıştır.

Yukarıda belirtilen çalışmalar üç aşamada gerçekleştirilmiştir. İlk aşamada, deterministik işlem süreli güvenilir olmayan seri üretim hatlarında hem üretim oranı maksimizasyonu hem de toplam tampon stok miktarı minimizasyonu problemlerinin çözümü için bir DKA algoritması önerilmiştir. İkinci aşamada ise yine aynı özelliklere sahip hatlarda üretim oranı maksimizasyonu problemini çözmek için bir ABKA algoritması önerilmiştir. Üçüncü aşamada ise ilk iki aşamadan farklı olarak problemin çözümü için popülasyon tabanlı arama algoritmaları önerilmiştir.

8.2 Bulgular

Önerilen tüm algoritmaların performansı literatürdeki karşılaştırma problemleri üzerinde test edilmiştir. Tablo 8.1’de DKA-MEKF, ABKA-DKF, SGA ve BP-BÇ algoritmalarının literatürde üretim oranı açısından karşılaştırıldıkları en iyi yönteme göre iyileşme oranları verilmiştir. Tablodan da görülebileceği gibi problem boyutu arttıkça komşuluk tabanlı arama algoritmalarının problemi çözmedeki etkinliği popülasyon tabanlı algoritmalara oranla daha fazladır. Ayrıca önerilen algoritmaların büyük boyutlu problemler üzerindeki performansı değerlendirildiğinde önerilen DKA-MEKF algoritmasının diğer algoritmalarından daha etkin olduğu sonucuna varmak mümkündür. Komşuluk tabanlı algoritmalar kendi içinde kıyaslandığında önerilen DKA-MEKF algoritmasının, popülasyon tabanlı algoritmalar kendi içinde kıyaslandığında ise önerilen BB-BÇ algoritmasının öne çıktığı görülmektedir.

Tablo 8.1: DKA-MEKF, ABKA-DKF, SGA ve BP-BÇ algoritmalarının karşılaştırması

Problem seti	DKA-MEKF	ABKA-DKF	SGA	BP-BÇ
(3, 20)	-	-	0,00	0,00
(5, 31)	0,00	0,00	0,00	0,00
(9, 160)	0,04	0,04	-	-
(10, 270)	-	0,33	0,33	0,33
(20, 100)	-	-	6,54	7,12
(40, 200)	-	-	6,63	7,60
(20, 400)	2,39	1,85	1,23	1,51
(40, 400)	1,52	1,59	0,58	1,09
(40, 800)	4,09	2,77	2,67	3,00
(40, 1600)	3,97	3,66	1,98	3,29

Algoritmalarda kullanılan çeşitlendirme stratejilerinin farklı yerel optimum noktalarda arama yapmasının çözüm kalitesi üzerinde etkisi olduğu gözlenmiştir. Ayrıca mevcut en iyi çözümün kalıtımını kullanarak yeni çözümler üretmenin algoritmaların yakınsama hızı üzerinde etkili olduğu deneysel çalışmalarla ortaya konmuştur. Önerilen yerel arama (IncDec) stratejisi de çözüm kalitesini artırmıştır. Ancak IncDec stratejisi ile bütün komşu çözümler değerlendirildiği için, bu stratejinin uygulanması çözüm süresini de arttırmıştır.

8.3 Katkılar

Bu doktora tez çalışmasının orijinal katkıları aşağıdaki gibi özetlenebilir:

- Literatürde ilk defa DKA, ABKA, SGA ve BP-BÇ algoritmaları TSDP'yi çözmek için önerilmiştir. Algoritmaların performansları literatürde mevcut karşılaştırma problemleri üzerinde test edilerek etkinlikleri ortaya konmuştur.
- DKA algoritmasında önerilen komşuluk yapıları karıştırma aşamasında rulet tekeri yöntemiyle belirlenmek suretiyle çözüm uzayı çeşitlendirilmiştir.
- ABKA algoritmasını TSDP'ye uyarlayabilmek için iki yeni çıkarma-ekleme operatör çifti önerilmiş ve adaptif bir şekilde kullanılmıştır. Bu operatör çiftlerinden makinelerin çalışma/bozulma durumlarını dikkate alan durum-tabanlı çıkarma-ekleme operatörünün problemin çözümünde oldukça etkin olduğu gösterilmiştir.
- DKA'nın yerel arama aşamasında, ABKA algoritmasında ise yerel optimumdan kaçmak amacıyla bir yerel arama sezgiseli önerilmiş ve arama boyunca etkin bir şekilde kullanılmıştır.
- Komşuluk tabanlı arama algoritmalarında rastgele çözüm üretme mekanizmasından farklı olarak algoritmaların harcadığı arama eforunu azaltmak üzere üç farklı başlangıç çözümü oluşturma sezgiseli önerilmiştir.
- SGA ve BP-BÇ algoritması sürekli karşılaştırma fonksiyonları ve sürekli optimizasyon problemlerinden farklı olarak ilk defa bir üretim hattı tasarım problemine uyarlanmıştır. SGA'da iki farklı mutasyon operatörü rulet tekeri yöntemiyle algoritma boyunca adaptif bir şekilde kullanılmıştır. Orijinal SGA'dan farklı olarak, bu tez çalışmasında iki noktalı çaprazlama operatörü kullanılmıştır. BP-BÇ algoritmasında ise üç farklı popülasyon üretme kuralı yine rulet tekeri yöntemiyle algoritma boyunca adaptif bir şekilde kullanılmıştır.
- Son olarak, popülasyon tabanlı olan SGA ve BP-BÇ algoritması, TSDP literatüründe çoğunlukla simülasyonla beraber kullanılan popülasyon tabanlı algoritmaların aksine analitik bir değerlendirme algoritmasıyla birlikte kullanılmıştır.

Güvenilir olmayan varsayımsal seri üretim hatları için etkinlikleri kanıtlanmış olan bu algoritmaların benzer özelliklere sahip gerçek hayat problemlerine kolaylıkla uygulanabileceği göz önünde bulundurulduğunda, bu tez çalışmasının işletme

yöneticileri için uzun dönem stratejik kararlarından biri olan üretim hattı tasarımı aşamasında faydalanabilecekleri bir rehber olduğu düşünülmektedir.

8.4 Gelecekte Yapılabilecek Çalışma Önerileri

Bu tez çalışmasının kapsamı dikkate alınarak gelecekte yapılabilecek çalışmalar şu şekilde özetlenebilir:

- Bu çalışmada sunulan deneysel sonuçlar sadece dengeli üretim hatlarını dikkate almıştır. Önerilen algoritmaların performansı, bu çalışmanın bir uzantısı olarak dengeli olmayan üretim hatları için test edilebilir.

- Önerilen algoritmaların performansları seri-paralel ve montaj/de-montaj üretim hatları gibi farklı topolojilere sahip üretim sistemleri için de test edilebilir. Bu amaçla önerilen algoritmaları farklı değerlendirme yöntemleriyle kullanmak mümkündür.

- Önerilen algoritmalar birbirleriyle veya literatürdeki diğer meta-sezgisel algoritmalarla arama eforunu azaltacak ve daha fazla çeşitlilik sağlayacak şekilde melezleştirilebilirler.

- Süreç içi stok maliyeti imalat sistemlerinde pek de istenmeyen bir maliyet kalemidir. Önerilen algoritmalar süreç içi stok minimizasyonu problemi gibi farklı amaç fonksiyonları için çeşitli kısıtlar altında kullanılabilir.

9. KAYNAKLAR

Ahuja, R. K., Ergun, Ö., Orlin, J. B. and Punnen, A. P., “A survey of very large scale neighborhood search techniques”, *Discret. Appl. Math.*, 123, 75-102, (2002).

Akpınar, Ş., “Hybrid large neighbourhood search algorithm for capacitated vehicle routing problem”, *Expert Syst. Appl.*, 61, 28-38, (2016).

Alataş, B., “Uniform big bang-chaotic big crunch optimization”, *Commun. Nonlinear Sci. Numer. Simul.*, 16 (9), 3696-3703, (2011).

Alfieri, A. and Matta, A., “Mathematical programming formulations for approximate simulation of multistage production systems”, *Eur. J. Oper. Res.*, 219 (3), 773-783, (2012).

Alfieri, A., Matta, A. and Pastore, E., “A column generation algorithm for the buffer allocation problem approximated by the time buffer concept”, *IFAC-PapersOnLine*, 49 (12), 739-744, (2016).

Alfieri, A., Matta, A. and Pastore, E., “The time buffer approximated buffer allocation problem: A row-column generation approach”, *Comput. Oper. Res.*, 115, <https://doi.org/10.1016/j.cor.2019.104835>, (2020).

Alinaghian, M. and Shokouhi, N., “Multi-depot multi-compartment vehicle routing problem, solved by a hybrid adaptive large neighborhood search”, *Omega*, 76, 85-99, (2018).

Aloise, D.J. Aloise, D. Rocha, C.T. Ribeiro, C.C. Ribeiro Filho, J.C. and Moura, L.S., “Scheduling workover rigs for onshore oil production”, *Discret. Appl. Math.*, 154 (5), 695-702, (2006).

Altıparmak, F., Dengiz, B. and Bulgak, A. A., “Optimization of buffer sizes in assembly systems using intelligent techniques” In Winter Simulation Conference Proceedings, San Diego, CA, USA, 1157-1162, (2002).

Altıparmak, F., Dengiz, B. and Bulgak, A. A., “Buffer allocation and performance modeling in asynchronous assembly system operations: An artificial neural network metamodeling approach”, *Appl. Soft Comput.*, 7, 946-956, (2007).

Amiri, M. and Mohtashami, A., "Buffer allocation in unreliable production lines based on design of experiments, simulation, and genetic algorithm", *Int. J. Adv. Manuf. Technol.*, 62, 371-383, (2012).

Amous, M., Toumi, S., Jarboui, B. and Eddaly, M., "A variable neighborhood search algorithm for the capacitated vehicle routing problem", *Electron. Notes Discret. Math.*, 58, 231-238, (2017).

Avcı, M. G. and Avcı, M., "An adaptive large neighborhood search approach for multiple traveling repairman problem with profits", *Comput. Oper. Res.*, 111, 367-385, (2019).

Azi, N., Gendreau, M. and Potvin, J. Y., "An adaptive large neighborhood search for a vehicle routing problem with multiple trips" Technical Report. 2010-08, CIRRELT, Montreal. 2010.

Azi, N., Gendreau, M. and Potvin, J., "A dynamic vehicle routing problem with multiple delivery routes", *Ann. Oper. Res.*, 199, 103-112, (2012).

Azi, N., Gendreau, M. and Potvin, J. Y., "An adaptive large neighborhood search for a vehicle routing problem with multiple routes", *Comput. Oper. Res.*, 41, 167-173, (2014).

Back, T., "Optimal mutation rates in genetic search", *Fifth International Conference on Genetic Algorithms*, United States, Morgan Kaufmann Publishers Inc., 5, 2-8, (1993).

Barten, K., "A queueing simulator for determining optimum inventory levels in a sequential process", *J. Ind. Eng.*, 13 (4), 245-252, (1962).

Battini, D., Persona, A. and Regattieri, A., "Buffer size design linked to reliability performance: A simulative study", *Comput. Ind. Eng.*, 56, 1633-1641, (2009).

Belacel, N., Cuperlovic-Culf, M., Laflamme, M. and Ouellette, R., "Fuzzy J-means and VNS methods for clustering genes from microarray data", *Bioinform.*, 20 (11), 1690-1701, (2004).

Bihan, H. L. and Dallery, Y., "A robust decomposition method for the analysis of production lines with unreliable machines and finite buffers", *Ann. Oper. Res.*, 93 (1), 265-297, (2000).

Biradar, S., Hote, Y. V. and Saxena, S., "Reduced-order modeling of linear time invariant systems using big bang big crunch optimization and time

moment matching method”, *Appl. Math. Modell.*, 40 (15-16), 7225-7244, (2016).

Borchani, R., Elloumi, A. and Masmoudi, M., “Variable neighborhood descent search based algorithms for course timetabling problem: Application to a Tunisian University”, *Electron. Notes Discret. Math.*, 58, 119-126, (2017).

Braysy, O., “A reactive variable neighborhood search for the vehicle-routing problem with time windows”, *INFORMS J Comput.*, 15 (4), 347-368, (2003).

Brimberg, J., Hansen, P., Mladenovic, N. and Taillard, É., “Improvements and comparison of heuristics for solving the multisource Weber problem”, *Oper. Res.*, 48 (3), 444-460, (2000).

Brimberg, J., Hansen, P., Lih, K. W., Mladenovic N. and Breton, M., “An oil pipeline design problem”, *Oper. Res.*, 51 (2), 228-239, (2003).

Bulgak, A. A., “Analysis and design of split and merge unpaced assembly systems by metamodelling and stochastic search”, *Int. J. Prod. Res.*, 44 (18-19), 4067-4080, (2006).

Burke, E. K., Cowling, P. I. and Keuthen, R., “Effective local and guided variable neighborhood search methods for the asymmetric traveling salesman problem”, In *Evo Workshop, LNCS*, 2037. Springer, 203-312, (2001).

Burke, E. K., Curtis, T., Post, G., Qu, R. and Veltman, B., “A hybrid heuristic ordering and variable neighborhood search for the nurse rostering problem”, *Eur. J. Oper. Res.*, 188, 330-341, (2008).

Burke, E. K., Li, J. and Qu, R., “A hybrid model of integer programming and variable neighbourhood search for highly-constrained nurse rostering problems”, *Eur. J. Oper. Res.*, 203 (2), 484-493, (2010).

Burman, M.H., “New results in flow line analysis”, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge MA, (1995).

Buzacott, J. A., “Automatic transfer lines with buffer stocks”, *Int. J. Prod. Res.*, 5, 182-200, (1967).

Buzacott, J. A., “Prediction of the efficiency of production systems without internal storage”, *Int. J. Prod. Res.*, 6, 173-188, (1968).

Buzacott, J. A. and Hanifin, L.E., “Models of automatic transfer lines with inventory banks: a review and comarison”, *AIIE Trans.*, 10, 197-207, (1978).

Buzacott, J. A. and Shanthikumar, J. G., Stochastic models of manufacturing systems, New Jersey: Prentice-Hall. (1993).

Camp, C. V., and Huq, F., “CO₂ and cost optimization of reinforced concrete frames using a big bang-big crunch algorithm”, *Eng. Struct.*, 48, 363-372, (2013).

Caramanis, M., “Production system design: A discrete event dynamic system and generalized Benders’ decomposition approach”, *Int. J. Prod. Res.*, 25 (8), 1223-1234, (1987).

Chaharsooghi, S. K. and Nahavandi, N., “Buffer allocation problem, a heuristic approach”, *Sci. Iran.*, 10 (4), 401-409, (2003).

Chan, F. T. S. and Ng, E. Y. H., “Comparative evaluations of buffer allocation strategies in a serial production line”, *Int. J. Adv. Manuf. Technol.*, 19 (11), 789-800, (2002).

Chen, S., Chen, R., Wang, G., Gao, J. and Kumar, A., “An adaptive large neighborhood search heuristic for dynamic vehicle routing problems”, *Comput. Electr. Eng.* 67, 596-607, (2018).

Chow, W. M., “Buffer capacity analysis for sequential production lines with variable process times”, *Int. J. Prod. Res.*, 25 (8), 1183-1196, (1987).

Chuang, L. Y., Yang, C. S., Li, J. C. and Yang, C. H., “Combat GA-based gene selection for classification of microarray data”, *Biomed. Eng.: Appl. Basis Commun.*, 20 (6), 345-352, (2008).

Chuang, L. Y., Yang, C. H., Li, J. C. and Yang, C. H., “A hybrid BPSO-CGA approach for gene selection and classification of microarray data”, *J Comput. Biol.*, 19 (1), 68-82, (2012).

Conway, R., Maxwell, W., McClain, J. O. and Thomas, J. L., “The role of work-in-process inventory in serial production lines” *Oper. Res.*, 36, 229-241, (1988).

Costa, A., Alfieri, A., Matta, A. and Fichera, S., “A parallel tabu search for solving the primal buffer allocation problem in serial production systems” *Comput. Oper. Res.*, 64, 97-112, (2015).

Costa, M., Monclar, F. and Zrikem, M., “Variable neighborhood decomposition search for the optimization of power plant cable layout”, *J. Intell. Manuf.*, 13, 353-365, (2002).

Cruz, F. R. B., Kendall, G., While, L., Duarte, A. R. and Brito, N. L. C., “Throughput maximization of queueing networks with simultaneous minimization of service rates and buffers”, *Math. Probl. Eng.*, <https://doi.org/10.1155/2012/692593>, (2012).

Cvetkovic, D. and Parmee, I. C., “Preferences and their application in evolutionary multiobjective optimization”, *IEEE Trans. Evol. Comput.*, 6 (1), 42-57, (2002).

Dallery, Y., David, R. and Xie, X. L., “An efficient algorithm for analysis of transfer lines with unreliable machines and finite buffers”, *IIE Trans.*, 20 (3), 281-283, (1988).

Dallery, Y., David, R. and Xie, X. L., “Approximate analysis of transfer lines with unreliable machines and finite buffers”, *IEEE Trans. Automat. Contr.*, 34 (9), 943-953, (1989).

Dallery, Y. and Gershwin, S.B., “Manufacturing flow line systems: a review of models and analytical results”, *Queueing Syst.*, 12 (1), 3-94, (1992).

Dallery, Y. and Frein, Y., “On decomposition methods for tandem queueing networks with blocking”, *Oper. Res.*, 41 (2), 386-399, (1993).

Dallery, Y. and Bihan, H. L., “An improved decomposition method for the analysis of production lines with unreliable machines and finite buffers”, *Int. J. Prod. Res.*, 37 (5), 1093-1117, (1999).

De Koster, M. B. M., “Estimation of line efficiency by aggregation”, *Int. J. Prod. Res.*, 25, 615-626, (1987).

Demir, E., Bektaş, T. and Laporte, G., “An adaptive large neighborhood search heuristic for the pollution-routing problem”, *Eur. J. Oper. Res.*, 223 (2), 346-359, (2012).

Demir, E., Bektaş, T. and Laporte, G., “The bi-objective pollution-routing problem”, *Eur. J. Oper. Res.*, 232, 464-478, (2014).

Demir, L., Tunali, S. and Løkketangen, A., “A tabu search approach for buffer allocation in production lines with unreliable machines”, *Eng. Optimiz.*, 43 (2), 213-231, (2011).

Demir, L., Tunali, S. and Eliiyi, D. T., “An adaptive tabu search approach for buffer allocation problem in unreliable non-homogenous production lines”, *Comput. Oper. Res.*, 39 (7), 1477-1486, (2012).

Demir, L., Tunali, S. and Eliiyi, D. T., “The state of the art on buffer allocation problem: A comprehensive survey”, *J. Intell. Manuf.*, 25 (3), 371-392, (2014).

Derbel, H., Jarboui, B. and Bhiri, R., “A skewed general variable neighborhood search algorithm with fixed threshold for the heterogeneous fleet vehicle routing problem”, *Ann. Oper. Res.*, 272, 243-272, (2019).

Desai, S. R. and Prasad, R., “A novel order diminution of LTI systems using big bang big crunch optimization and routh approximation”, *Appl. Math. Model.*, 37 (16-17), 8016-8028, (2013).

Diamantidis, A. C. and Papadopoulos, C. T., “A dynamic programming algorithm for the buffer allocation problem in homogeneous asymptotically reliable serial production lines”, *Math. Probl. Eng.*, 3, 209-223, (2004).

Diamantidis, A., Lee, J-H., Papadopoulos, C. T., Li, J. and Heavey, C., “Performance evaluation of flow lines with non identical and unreliable parallel machines and finite buffers”, *Int. J. Prod. Res.*, 58 (13), 3881-3904, (2020).

Dolgui, A., Ereemeev, A., Kolokolov, A. and Sigaev, V., “A Genetic algorithm for the allocation of buffer storage capacities in a production line with unreliable machines”, *J. Math. Model. Algorithms*, 1 (2), 89-104, (2002).

Dolgui, A., Ereemeev, A. V. and Sigaev, V. S., “HBBA: Hybrid algorithm for buffer allocation in tandem production lines”, *J. Intell. Manuf.*, 18 (3), 411-420, (2007).

Dolgui, A., Ereemeev, A., Kovalyov, M. Y. and Sigaev, V., “Complexity of buffer capacity allocation problems for production lines with unreliable machines.” *J. Math. Model. Algorithms Oper. Res.*, 12 (2), 155-165, (2013).

Dolgui, A. B., Ereemeev, A. V. and Sigaev, V.S., “Analysis of a multicriterial buffer capacity optimization problem for a production line”, *Autom. Remote Control.*, 78 (7), 1276-1289, (2017).

Eksin, I. and Erol, O.K., “Evolutionary algorithm with modifications in the reproduction phase”, *IEE Proceedings-Software*, 148 (2), 75-80, (2001).

Elshaikh, A., Salhi, S. and Nagy, G., “The continuous p-centre problem: An investigation into variable neighborhood search with memory”, *Eur. J. Oper. Res.*, 241 (3), 606-621, (2015).

Erol, O. K. and Eksin, I., “A new optimization method: Big Bang-Big Crunch”, *Adv. Eng. Softw.*, 37 (2), 106-111, (2006).

Fernandez Caban, P. L. and Masters, F. J., “Hybridizing particle swarm and big bang-big crunch optimization methods to explore then exploit the design domain of large planar frame structures”, *Comput. Struct.*, 202, 1-14, (2018).

Ferreira, H. S., Bogue, E. T., Noronha, T. F., Belhaiza, S. and Prins, C., “Variable Neighborhood Search for Vehicle Routing Problem with Multiple Time Windows.” *Electron. Notes Discret. Math.*, 66, 207-214, (2018).

Fleszar, K., and Hindi, K. S., “New heuristics for one-dimensional bin-packing”, *Comput. Oper. Res.*, 29 (7), 821-839, (2002).

Fleszar, K. and Hindi, K. S., “An effective VNS for the capacitated p-median problem”, *Eur. J. Oper. Res.*, 191 (3), 612-622, (2008).

Fleszar, K., Osman, I. H. and Hindi, K. S., “A variable neighborhood search algorithm for the open vehicle routing problem”, *Eur. J. Oper. Res.*, 195 (3), 803-809, (2009).

Fonseca, G. H. G. and Santos, H. G., “Variable Neighborhood Search based algorithms for high school timetabling”, *Comput. Oper. Res.*, 52 (B), 203-208, (2014).

Franceschetti, A., Demir, E., Honhon, D., Van Woensel, T., Laporte, G. and Stobbe, M., “A metaheuristic for the time-dependent pollution-routing problem”, *Eur. J. Oper. Res.*, 259, 972-991, (2017).

Fuxman, L., “Optimal buffer allocation in asynchronous cyclic mixed model assembly lines”, *Prod. Oper. Manag.*, 7 (3), 294-311, (1998).

Garcia Lopez, F., Melian-Batista, B., Moreno-Perez, J. A. and Moreno-Vega, J. M., “The parallel variable neighborhood search for the p-median problem”, *J Heuristics*, 8 (3), 375-388, (2002).

Gen, M. and Lin, L., “Multiobjective evolutionary algorithm for manufacturing scheduling problems: State-of-the-art survey”, *J. Intel. Manuf.*, 25 (5), 849-866, (2014).

Genç, H. M., Eksin, I. and Erol, O. K., “Big bang-big crunch optimization algorithm with local directional moves”, *Turk. J. Electr. Eng. Co.*, 21 (5), 1359-1375, (2013).

Gershwin, S. B., “An efficient decomposition method for the approximate evaluation of production lines with finite storage space and blocking”, *Oper. Res.*, 35 (2), 291-305, (1987).

Gershwin, S.B., *Manufacturing Systems Engineering*, New Jersey, Prentice Hall, (1994).

Gershwin, S. B., “Design and operation of manufacturing systems – The Control-Point Policy”, *IIE Trans.* 32 (10), 891-906, (2000).

Gershwin, S. B., “The future of manufacturing systems engineering”, *Int. J. Prod. Res.*, 56 (1-2), 224-237, (2018).

Gershwin S. B. and Berman, O., “Analysis of transfer lines consisting of two unreliable machines with random processing times and finite storage buffers”, *AIIE Trans.*, 13, 2-11, (1981).

Gershwin, S. B. and Schick, I. C., “Modeling and analysis of three-stage transfer lines with unreliable machines and finite buffers”, *Oper. Res.*, 31 (2), 354-380, (1983).

Gershwin, S. B. and Goldis, Y., “Efficient algorithms for transfer line design”, *MIT Laboratory for Manufacturing and Productivity Report*, LMP-95-005, (1995).

Gershwin, S. B., and Schor, J., “Efficient algorithms for buffer space allocation”, *Ann. Oper. Res.*, 93 (1-4), 117-144, (2000).

Ghilas, V., Demir, E. and Van Woensel, T., “An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows and scheduled lines”, *Comput. Oper. Res.*, 72, 12-30, (2016).

Glasse, C.R. and Hong, Y., “Analysis of behaviour of an unreliable n-stage transfer line with (n-1) inter-storage buffers”, *Int. J. Prod. Res.*, 31, 519-530, (1993).

Gogna, A. and Tayal, A., “Metaheuristics: review and application”, *J. Exp. Theor. Artif. Intell.*, 25 (4), 503-526, (2013).

Güner Gören, H., Tunali, S. and Jans, R., “A review of applications of genetic algorithms in lot sizing”, *J. Intel. Manuf.*, 21 (4), 575-590, (2010).

Hamdi, M. and Zaied, M., “Resource allocation based on hybrid genetic algorithm and particle swarm optimization for D2D multicast communications”, *Appl. Soft Comput. J.*, 83, 105605, (2019).

Hanifin, L. E., “Increased Transfer Line Productivity Utilizing Systems Simulation,” Ph.D. Thesis, *University of Detroit*, Detroit, (1975).

Hansen, P. and Mladenovic, N., “An introduction to variable neighborhood search”, (eds: Voss, S., Martello, S., Osman, I. H., Roucairol, C.), *Metaheuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer, Dordrecht, 433-458, (1999).

Hansen, P. and Mladenovic, N., “Developments of Variable Neighborhood Search. In: *Essays and Surveys in Metaheuristics*”, *Operations Research/Computer Science Interfaces Series*, Springer, Boston, MA, https://doi.org/10.1007/978-1-4615-1507-4_19, 415-439, (2001).

Hansen, P., Mladenovic, N. and Pérez-Brito, D., “Variable neighborhood decomposition search”, *J. Heuristics.*, 7 (4), 335-350, (2001).

Hansen, P., Mladenovic, N. and Moreno Perez, J. A., “Variable neighborhood search: methods and applications”, *4OR*, 6, 319-360, (2008).

Hansen, P., Mladenovic, N., Todosijevic, R. and Hanafi, S., “Variable neighborhood search: basics and variants”, *EURO J. Comput. Optim.*, 5, 423-454, (2017).

Hasançebi, O. and Azad, S. K., “An exponential big bang-big crunch algorithm for discrete design optimization of steel frames”, *Comput. Struct.*, 110-111, 167-179, (2012).

Helber, S., “Cash-flow-oriented buffer allocation in stochastic flow lines”, *Int. J. Prod. Res.*, 39 (14), 3061-3083, (2001).

Hemmelmayr, V., Schmid, V. and Blum, C., “Variable neighborhood search for the variable sized bin packing problem”, *Comput. Oper. Res.*, 39 (5), 1097-1108, (2012).

Hemmelmayr, V., Smilowitz, K. and Torre, L., “A periodic location routing problem for collaborative recycling”, *IIE Trans.*, 49 (4), 414-428, (2017).

Hifi, M., Saleh, S. and Wu, L., “A Fast Large Neighborhood Search for Disjunctively Constrained Knapsack Problems”, (eds: Fouilhoux P., Gouveia L., Mahjoub A., Paschos V.), *Combinatorial Optimization. ISCO 2014. Lecture Notes in Computer Science*, 8596. Springer, Cham., 396-407, https://doi.org/10.1007/978-3-319-09174-7_34, (2014).

Hillier, M. S., “Characterizing the optimal allocation of storage space in production line systems with variable processing times”, *IIE Trans.*, 32 (1), 1-8, (2000).

Hillier, M. S., “Designing unpaced production lines to optimize throughput and work-in-process inventory”, *IIE Trans.*, 45 (5), 516-527, (2013).

Hillier, F.S. and Boling, R.W., “The effect of some design factors on the efficiency of production lines with variable operation times”, *J. Ind. Eng.*, 17, 651-658, (1966).

Hillier, F.S. and Boling, R.W., “Toward characterizing the optimal allocation of work in production lines with variable operation times”, Amsterdam, *Adv. Oper. Res.*, 109-119, (1977).

Hillier, F.S. and Boling, R.W., “On the optimal allocation of work in symmetrically unbalanced production line systems with variable operation times”, *Manag. Sci.*, 25, 721-728, (1979).

Hillier, F. S., So, K. C., and Boling, R. W., “Toward characterizing the optimal allocation of storage space in production line systems with variable processing times”, *Manage. Sci.*, 39, 126-133, (1993).

Ho, Y. C., Eyler, M. A. and Chien, T. T., “A gradient technique for general buffer storage design in a production line”, *Int. J. Prod. Res.*, 17 (6), 557-580, (1979).

Hof, J. and Schneider, M., “An adaptive large neighborhood search with path relinking for a class of vehicle-routing problems with simultaneous pickup and delivery”, *Networks*, 74 (3), 207-250, <https://doi.org/10.1002/net.21879>, (2019).

Holland, J. H., *Adaptation in natural and artificial systems*, Ann Arbor, MIT Press, Michigan, (1975).

Irawan, C. A., Imran, A. and Luis, M., “Solving the bi-objective capacitated p-median problem with multilevel capacities using compromise programming and VNS”, *Intl. Trans. in Op. Res.*, 27, 361-380, (2020).

Iris, Ç., Pacino, D. and Ropke, S., “Improved formulations and an adaptive large neighborhood search heuristic for the integrated berth allocation and quay crane assignment problem”, *Transp. Res. Part E Logist. Transp. Rev.*, 105, 123-147, (2017).

Irnich, S., Funke, B. and Grünert, T., “Sequential search and its application to vehicle-routing problems”, *Comput. Oper. Res.*, 33 (8), 2405-2429, (2006).

Joshi, S. K. and Bansal, J. C., “Parameter tuning for meta-heuristics”, *Knowl. Based Syst.*, 189, 105094, (2020).

Jourdan, L., Basseur, M. and Talbi, E. G., “Hybridizing exact methods and metaheuristics: A taxonomy”, *Eur. J. Oper. Res.*, 199, 620-629, (2009).

Karakostas, P., Sifaleras, A. and Georgiadis, M.C., “Variable neighborhood search-based solution methods for the pollution location inventory-routing problem”, *Optim. Lett.*, <https://doi.org/10.1007/s11590-020-01630-y>, (2020).

Kaveh, A. and Talatahari, S., “Optimal design of Schwedler and ribbed domes via hybrid Big Bang-Big Crunch algorithm”, *J. Constr. Steel. Res.*, 66 (3), 412-419, (2010).

Kaveh, A. and Talatahari, S., “Size optimization of space trusses using Big Bang-Big Crunch algorithm”, *Comput. Struct.*, 87 (17-18), 1129-1140, (2009).

Kerbache, L. and Smith, J. M., “The generalized expansion method for open finite queueing networks”, *Eur. J. Oper. Res.*, 32, 448-461, (1987).

Kiesmüller, G. P. and Sachs, F. E., “Spare parts or buffer? How to design a transfer line with unreliable machines”. *Eur. J. Oper. Res.*, 284 (1), 121-134, (2020).

Kim, J. and Gershwin, S. B., “Analysis of long flow lines with quality and operational failures”, *IIE Trans.*, 40 (3), 284-296, (2008).

Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P., “Optimization by simulated annealing”, *Science*, 220 (4598), 671-680, (1983).

Kirkpatrick, S. and Toulouse, G. (1985) “Configuration space analysis of traveling salesman problems”, *J Phys.*, 46, 1277-1292, (1985).

Kocatürk, F. and Özpeynirci, Ö., “Variable neighborhood search for the pharmacy duty scheduling problem”, *Comput. Oper. Res.*, 51, 218-226, (2014).

Koenigsberg, E., “Production lines and internal storage-A review”. *Manage. Sci.*, 5 (4), 410-433, (1959).

Kolb, O. and Göttlich, S., “A continuous buffer allocation model using stochastic processes”, *Eur. J. Oper. Res.*, 242 (3), 865-874, (2015).

Köse, S. Y. and Kılınçcı, O., “Hybrid approach for buffer allocation in open serial production lines”, *Comput. Oper. Res.*, 60, 67-78, (2015).

Köse, S. Y. and Kılınçcı, O., “A multi-objective hybrid evolutionary approach for buffer allocation in open serial production lines”, *J. Intel. Manuf.*, 31, 33-51, (2020).

- Köse, S. Y., Demir, L., Tunalı, S. and Eliiyi, D. T., “Capacity improvement using simulation optimization approaches: A case study in the thermotechnology industry”, *Eng. Optimiz.*, 47 (2), 149-164, (2015).
- Kusiak, A., “Smart manufacturing”, *Int. J. Prod. Res.*, 56 (1-2), 508-517, (2018).
- Kytöjoki, J., Nuortio, T., Bräysy, O. and Gendreau, M., “An efficient variable neighborhood search heuristic for very large scale vehicle routing problems”, *Comput. Oper. Res.*, 34 (9), 2743-2757, (2007).
- Lapierre, S., Ruiz, A. and Soriano, P., “Designing Distribution Networks: Formulations and Solution Heuristic”, *Transp. Sci.*, 38 (2), 174-187, (2004).
- Li, J. and Meerkov, S. M. (2009). *Production Systems Engineering*. Springer.
- Li, J., Blumenfeld, D. E., Huang, H. and Alden, J. M., “Throughput analysis of production systems: recent advances and future topics”, *Int. J. Prod. Res.*, 47 (14), 3823-3851, (2009).
- Li, L., Qian, Y. L., Yang, Y. M. and Du, K., “A fast algorithm for buffer allocation problem”, *Int. J. Prod. Res.*, 54 (11), 3243-3255, (2015).
- Li, L., Qian, Y., Du, K. and Yang, Y., “Analysis of approximately balanced production lines”, *Int. J. Prod. Res.*, 54 (3), 647-664, (2016).
- Liberopoulos, G. and Tsarouhas, P., “Systems analysis speeds up Chipita’s food-processing line”. *Interfaces*, 32 (3), 62-76, (2002).
- Liberopoulos, G., “Comparison of optimal buffer allocation in flow lines under installation buffer, echelon buffer, and CONWIP policies”, *Flex. Serv. Manuf. J.*, 32, 297-365, (2020).
- Lin, Z., Frigerio, N., Matta, A. and Du, S., “Multi-fidelity surrogate-based optimization for decomposed buffer allocation problems”, *OR Spectrum*, <https://doi.org/10.1007/s00291-020-00603-y>, (2020).
- Lopes, T. C., Sikora, C. G. S., Michels, A. S. and Magatao, L., “An iterative decomposition for asynchronous mixed-model assembly lines: combining balancing, sequencing, and buffer allocation”, *Int. J. Prod. Res.*, 58 (2), 615-630, (2020).
- Loudni, S., Boizumault, P. and David, P., “On-line resources allocation for ATM networks with rerouting”, *Comput. Oper. Res.*, 33, 2891-2917, (2006).

Mak, K. L., "The allocation of interstage buffer storage capacity in production lines", *Comput. Ind. Eng.*, 10 (3), 163-169, (1986).

Massim, Y., Yalaoui, F., Amodeo, L., Chatelet, E. and Zebblah, A., "Efficient combined immune-decomposition algorithm for optimal buffer allocation in production lines for throughput and profit maximization", *Comput. Oper. Res.*, 37 (4), 611-620, (2010).

Matheron, G., "Principles of geostatistics", *Econ. Geol.*, 58 (8), 1246-1266, (1963).

Matta, A. and Simone, F., "Analysis of two-machine lines with finite buffer, operation-dependent and time-dependent failure modes", *Int. J. Prod. Res.*, 54 (6), 1850-1862, (2016).

Matta, A., Pezzoni, M. and Semeraro, Q., "A kriging-based algorithm to optimize production systems approximated by analytical models", *J. Intel. Manuf.*, 23 (3), 587-597, (2012).

McClain, J. O. and Moodie, D. R., "A comment on Buffer Space Allocation in Automated Assembly Lines", *Oper. Res.*, 39 (5), 857-860, (1991).

Meester, L. E. and Shanthikumar, J. G., "Concavity of the throughput of tandem queuing systems with finite buffer storage space", *Adv. Appl. Probab.*, 22, 764-767, (1990).

Menendez, B., Bustillo, M., Pardo, E. G. and Duarte, A., "General variable neighborhood search for the order batching and sequencing problem", *Eur. J. Oper. Res.*, 263 (1), 82-93, (2017a).

Menendez, B., Pardo, E. G., Alonso-Ayuso, A., Molina, E. and Duarte, A., "Variable neighborhood search strategies for the order batching problem", *Comput. Oper. Res.*, 78, 500-512, (2017b).

Meric, L., Pesant, G. and Pierre, S., "Variable neighborhood search for optical routing in networks using latin routers", *Ann Telecommun.*, 59 (3), 261-286, (2004).

Milthens, N. P. M., "Solving VRP using voronoi diagrams and adaptive large neighborhood search", MSc. Thesis, *University of Copenhagen*, Copenhagen, (2009).

Mladenovic, N., "A variable neighborhood algorithm-A new metaheuristic for combinatorial optimization", Presented at Optimization Days, Montreal, (1995).

Mladenovic, N. and Hansen, P., "Variable neighborhood search", *Comput. Oper. Res.*, 24 (11), 1097-1100, (1997).

Mladenovic, N., Labbe, M. and Hansen, P., "Solving the p-center problem by tabu search and variable neighborhood search", *Networks* 42,48-64, (2003a).

Mladenovic, N., Petrovic, J., Kovacevic-Vujcic, V. and Cangalovic, M., "Solving spread spectrum radar polyphase code design problem by tabu search and variable neighborhood search", *Eur. J. Oper. Res.*, 151, 389-399, (2003b).

Mladenovic, N., Alkandari, A., Pei, J., Todosijevic, R. and Pardalos, P. M., "Less is more approach: basic variable neighborhood search for the obnoxious p-median problem", *Int. Trans. in Oper. Res.*, 27, 480-493, (2020).

Mobin, M., Mousavi, S.M., Komaki, M. and Tavana, M., "A hybrid desirability function approach for tuning parameters in evolutionary optimization algorithms", *Meas.*, 114, 417-427, (2018).

Mohtashami, A., "A new hybrid method for buffer sizing and machine allocation in unreliable production and assembly lines with general distribution time-dependent parameters", *Int. J. Adv. Manuf. Technol.*, 74 (9-12), 1577-1593, (2014).

Moslehi, G. and Khorasanian, D., "A hybrid variable neighborhood search algorithm for solving the limited-buffer permutation flow shop scheduling problem with the makespan criterion", *Comput. Oper. Res.*, 52 (B), 260-268, (2014).

Motlagh, M. M., Azimi, P., Amiri, M. and Madraki, G., "An efficient simulation optimization methodology to solve a multi-objective problem in unreliable unbalanced production lines", *Expert Syst. Appl.*, 138, <https://doi.org/10.1016/j.eswa.2019.112836>, (2019).

Mucherino, A. and Liberti, L., "A VNS-Based Heuristic for Feature Selection in Data Mining", (eds: Talbi E. G.), *Hybrid Metaheuristics. Studies in Computational Intelligence*, 434, 353-368, Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-30671-6_13, (2013).

Muelas, S., LaTorre, A. and Pena, J. M., "A variable neighborhood search algorithm for the optimization of a dial-a-ride problem in a large city", *Expert Syst. Appl.*, 40, 5516-5531, (2013).

Nahas, N., Ait-Kadi, D. and Nourelfath, M., "A new approach for buffer allocation in unreliable production lines", *Int. J. Prod. Econ.* 103, 873-881, (2006).

Nahas, N., Nourelfath, M. and Ait-Kadi, D., “Selecting machines and buffers in unreliable series-parallel production lines”, *Int. J. Prod. Res.*, 47 (14), 3741-3774, (2009).

Nahas, N., Nourelfath, M. and Gendreau, M., “Selecting machines and buffers in unreliable assembly/disassembly manufacturing networks”, *Int. J. Prod. Econ.*, 154, 113-126, (2014).

Nahas, N. and Nourelfath, M., “Joint optimization of maintenance, buffers and machines in manufacturing lines”, *Eng. Optimiz.*, 50 (1), 37-54, (2018).

Narasimhamu, K. L., Venugopal Reddy, V. and Rao, C. S. P., “Optimal buffer allocation in tandem closed queuing network with single server using PSO”, *Procedia Mater. Sci.*, 5, 2084-2089, (2014).

Niyomubyeyi, O., Sicaio, T.E., Diaz Gonzalez, J. I., Pilesjö, P. and Mansourian, A., “A Comparative Study of Four Metaheuristic Algorithms, AMOSA, MOABC, MSPSO, and NSGA-II for Evacuation Planning”, *Algorithms*, 13(1), <https://doi.org/10.3390/a13010016>, (2020).

Nourelfath, M., Nahas, N. and Ait-Kadi, D., “Optimal design of series production lines with unreliable machines and finite buffers”, *J. Qual. Maint. Eng.*, 11 (2), 121-138, (2005).

Okamura, K. and Yamashina, H., “Analysis of the effect of buffer storage capacity in transfer line systems”, *AIIE Trans.* 9, 127-135, (1977).

Ouazene, Y., Yalaoui, A., Yalaoui, F. and Chehade, H., “Non-linear programming method for buffer allocation in unreliable production lines”, In Sericola B, Telek M, Horvath G, editors, *Lecture Notes in Computer Science, Analytical and Stochastic Modeling Techniques and Applications*, Springer, United States, 8499, 80-94, (2014).

Pacheco, J., Prorras, S., Casado, S. and Baruque, B., “Variable neighborhood search with memory for a single-machine scheduling problem with periodic maintenance and sequence-dependent set-up times”, *Knowl. -Based Syst.*, 145, 236-249, (2018).

Papadopoulos, H. T., Heavey, C. and Browne J., “Queueing Theory in Manufacturing Systems Analysis and Design”, London: Chapman & Hall, (1993).

Papadopoulos, H.T. and Heavey, C., “Queueing theory in manufacturing systems analysis and design: A classification of models for production and transfer lines”, *Eur. J. Oper. Res.*, 92 (1), 1-27, (1996).

- Papadopoulos, H.T. and Vidalis, M.I., “A heuristic algorithm for the buffer allocation in unreliable unbalanced production lines”, *Comput. Ind. Eng.*, 41 261-277, (2001).
- Papadopoulos, C. T., O'Kelly, M. E. J., Vidalis, M. J. and Spinellis, D., “Analysis and Design of Discrete Part Production Lines”, Springer, United States, (2009).
- Papadopoulos, C. T., O'Kelly, M. E. J., and Tsadiras, A. K., “A DSS for the buffer allocation of production lines based on a comparative evaluation of a set of search algorithms”, *Int. J. Prod. Res.*, 51 (14), 4175-4199, (2013).
- Park, T., “A two-phase heuristic algorithm for determining buffer sizes of production lines”, *Int. J. Prod. Res.*, 31 (3), 613-631, (1993).
- Pedrielli, G., Matta, A., Alfieri, A. and Zhang, M., “Design and control of manufacturing systems: a discrete event optimisation methodology”, *Int. J. Prod. Res.*, 56 (1-2), 543-564, (2018).
- Pepin, A. S., Desaulniers, G., Hertz, A. and Huisman, D., “A comparison of five heuristics for the multiple depot vehicle scheduling problem”, *J. Sched.*, 12, 17–30, (2009).
- Pereira, J. and Vila, M., “Variable neighborhood search heuristics for a test assembly design problem”, *Expert Syst. Appl.*, 42 (10), 4805-4817, (2015).
- Perros, H. G., “Stochastic analysis of computer and communication systems”, in: *Open Queueing Networks with Blocking*, ed. H. Takagi, Amsterdam, (1989).
- Pisinger, D. and Ropke, S., “A general heuristic for vehicle routing problems”, *Comput. Oper. Res.*, 34, 2403-2435, (2007).
- Pisinger, D. and Ropke, S., “Large neighborhood search”, In: Gendreau M, Potvin J-Y, editors. *Handbook of Metaheuristics of International Series in Oper. Res. Manage. Sci.*, 146, Boston, Springer, 399-419, (2010).
- Polacek, M., Hartl, R. F., Doerner, K. and Reimann, M., “A variable neighborhood search for the multi depot vehicle routing problem with time Windows”, *J. Heuristics*, 10 (6), 613-627, (2004).
- Powell, S. G. and Pyke, D. F., “Allocation of buffers to serial production lines with bottlenecks”, *IIE Trans.*, 28 (1), 18-29, (1996).

Qudeiri, J. A., Yamamoto, H., Ramli, R., and Jamali, A., “Genetic algorithm for buffer size and work station capacity in serial–parallel production lines”, *Artif Life Robot.*, 12, 102-106, (2008).

Rahimian, E., Akartunalı, K. and Levine, J., “A hybrid integer programming and variable neighborhood search algorithm to solve nurse rostering problems”, *Eur. J. Oper. Res.*, 258 (2), 411-423, (2017).

Ribeiro, G. M. and Laporte, G., “An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem”, *Comput. Oper. Res.*, 39, 728-735, (2012).

Romero-Silva, R. and Shaaban, S., “Influence of unbalanced operation time means and uneven buffer allocation on unreliable merging assembly line efficiency”, *Int. J. Prod. Res.*, 57 (6), 1645-1666, (2019).

Ropke, S. and Pisinger, D., “A unified heuristic for a large class of vehicle routing problems with backhauls”, *Eur. J. Oper. Res.*, 171 (3), 750-775, (2006a).

Ropke, S. and Pisinger, D., “An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows”, *Transp. Sci.*, 40 (4), 393-546, (2006b).

Rousseau, L. M., Gendreau, M. and Pesant, G., “Using constraint-based operators to solve the vehicle routing problem with time windows”, *J Heuristics*, 8, 43-58, (2002).

Sabuncuoğlu, İ., Erel, E. and Göçgün, Y., “Analysis of serial production lines: characterization study and a new heuristic procedure for optimal buffer allocation”, *Int. J. Prod. Res.*, 44 (13), 2499-2523, (2006).

Sarasola, B. and Doerner, K. F., “Adaptive large neighborhood search for the vehicle routing problem with synchronization constraints at the delivery location”, *Networks*, 75, 64-85, <https://doi.org/10.1002/net.21905>, (2020).

Sarasola, B., Doerner, K. F., Schmid, V. and Alba, E., “Variable neighborhood search for the stochastic and dynamic vehicle routing problem”, *Ann. Oper. Res.*, 236, 425-461, (2016).

Sedighzadeh, M., Ahmadi, S. and Sarvi, M., “An efficient hybrid big bang-big crunch algorithm for multi-objective reconfiguration of balanced and unbalanced distribution systems in fuzzy framework”, *Electr. Pow. Compo. Sys.*, 41 (1), 75-99, (2013).

Shaw, P., “A new local search algorithm providing high quality solutions to vehicle routing problems”, *Comput. Sci.*, <https://www.semanticscholar.org/paper/A-new-local-search-algorithm-providing-high-quality-Shaw/8ac9d5287ede9463d33cc0a1daf4191a5e34608d>, (1997).

Shaw, P., “Using constraint programming and local search methods to solve vehicle routing problems”, in: *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 1998.

Shi, C. and Gershwin, S. B., “An efficient buffer design algorithm for production line profit maximization”, *Int. J. Prod. Econ.*, 122 (2), 725-740, (2009).

Shi, C. and Gershwin, S. B., “A segmentation approach for solving buffer allocation problems in large production systems”, *Int. J. Prod. Res.*, 54 (20), 6121-6141, (2016).

Shi, L. and Men, S., “Optimal buffer allocation in production lines”, *IIE Trans.*, 35 (1), 1-10, (2003).

Singh, A. and Smith, J. M., “Buffer allocation for an integer nonlinear network design problem”, *Comput. Oper. Res.*, 24 (5), 453-472, (1997).

Smith, J. M., “Joint optimisation of buffers and network population for closed finite queueing systems”, *Int. J. Prod. Res.*, 54 (17), 5111-5135, (2016).

Smith, J. M., “Simultaneous buffer and service rate allocation in open finite queueing networks”, *IIE Trans.*, 50 (3), 203-216, (2018).

Smith, J. M. and Chikhale, N., “Buffer allocation for a class of nonlinear stochastic knapsack problems”, *Ann. Oper. Res.*, 58, 323-360, (1995).

Smith, J. M. and Cruz, F. R. B., “The buffer allocation problem for general finite buffer queueing networks”, *IIE Trans.*, 37 (4), 343-365, (2005).

Smiti, N., Dhiaf, M. M., Jarboui, B. and Hanafi, S., “Skewed general variable neighborhood search for the cumulative capacitated vehicle routing problem”, *Int. Trans. in Oper. Res.*, 27, 651-664, (2020).

Sohlenius, G., “The manufacturing system: our motor of welfare”, *Royal Institute of technology, Department of Production Engineering*, Stockholm, ISSN 1650-1888, (2000).

Spinellis, D. D. and Papadopoulos, C. T., “A simulated annealing approach for buffer allocation in reliable production lines”, *Ann. Oper. Res.*, 93 (1), 373-384, (2000).

Spinellis, D. D., Papadopoulos, C. T. and Smith, J. M., “Large production line optimization using simulated annealing”, *Int. J. Prod. Res.*, 38 (3), 509-541, (2000).

Staley, D. R. and Kim, D. S., “Experimental results for the allocation of buffers in closed serial production lines”, *Int. J. Prod. Econ.*, 137 (2), 284-291, (2012).

Su, C., Shi, Y. and Dou, J., “Multi-objective optimization of buffer allocation for remanufacturing system based on TS-NSGA-II hybrid algorithm”, *J. Clean. Prod.*, 166, 756-770, (2017).

Sze, J. F., Salhi, S. and Wassan, N., “The cumulative capacitated vehicle routing problem with min-sum and min-max objectives: An effective hybridisation of adaptive variable neighborhood search and large neighborhood search”, *Transport. Res. Part B*, 101, 162-184, (2017).

Talbi, E. G. *Metaheuristics: From Design to Implementation*, 2009.

Tang, H., Zhou, J., Xue, S. and Xie, L., “Big Bang-Big Crunch optimization for parameter estimation in structural systems”, *Mech. Syst. Signal Process.*, 24(8), 2888-2897, (2010).

Taşan, S. O., and Tunali, S., “A review of the current applications of genetic algorithms in assembly line balancing”, *J. Intel. Manuf.*, 19(1), 49-69, (2008).

Taşgetiren, M. F., Pan, Q. K., Kızılay, D. and Suer, G., “A differential evolution algorithm with variable neighborhood search for multidimensional knapsack problem”, *IEEE Congress on Evolutionary Computation*, 2797-2804, (2015).

Tempelmeier, H., “Practical considerations in the optimization of flow production systems”, *Int. J. Prod. Res.*, 41 (1), 149-170, (2003).

Thierens, D., “Adaptive mutation rate control schemes in genetic algorithms”, *Proceedings of the Congress on Evolutionary Computation*, 1, 980-985, (2002).

Tiacci, L., “Simultaneous balancing and buffer allocation decisions for the design of mixed-model assembly lines with parallel workstations and stochastic task times”, *Int. J. Prod. Econ.*, 162, 201-215, (2015).

- Tsadiras, A. K., Papadopoulos, C. T. and O'Kelly, M. E. J., "An artificial neural network based decision support system for solving the buffer allocation problem in reliable production lines", *Comput. Ind. Eng.*, 66 (4), 1150-1162, (2013).
- Waibel, M. W., Steenkamp, L. P., Moloko, N. and Oosthuizen, G.A., "Investigating the effects of Smart Production Systems on sustainability elements", 14th Global Conference on Sustainable Manufacturing, *Procedia Manufacturing*, 8, 731-737, (2017).
- Wang, G., Shin, Y. W. and Moon, D. H., "Comparison of three flow line layouts with unreliable machines and profit maximization", *Flex. Serv. Manuf. J.*, 28 (4), 669-693, (2016).
- Weiss, S. and Stolletz, R., "Buffer allocation in stochastic flow lines via sample-based optimization with initial bounds", *OR Spectrum*, 37 (4), 869-902, (2015).
- Weiss, S., Matta, A. and Stolletz, R., "Optimization of buffer allocations in flow lines with limited supply", *IISE Trans.*, 50, 191-202, (2018).
- Weiss, S., Schwarz, J. A. and Stolletz, R., "The buffer allocation problem in production lines: Formulations, solution methods, and instances", *IISE Trans.*, 51 (5), 456-485, (2019).
- Wen, M., Linde, E., Ropke, S., Mirchandani, P. and Larsen, A., "An adaptive large neighborhood search heuristic for the electric vehicle scheduling problem", *Comput. Oper. Res.*, 76, 73-83, (2016).
- Wu, B. and Chang, Y., "A framework for strategically-driven evaluation of enterprise information systems", *Int. J. Prod. Res.*, 38 (17), 4071-4084, (2000).
- Wu, B., "A unified framework of manufacturing systems design", *Ind. Manag. Data Syst.*, 101 (9), 446-469, (2001).
- Xi, S., Chen, Q., Smith, J. M., Mao, N., Yu, A. and Zhang, H., "A new method for solving buffer allocation problem in large unbalanced production lines", *Int. J. Prod. Res.*, <https://doi.org/10.1080/00207543.2019.1685709>, (2019).
- Xia, B., Xi, L. and Zhou, B., "An improved decomposition method for evaluating the performance of transfer lines with unreliable machines and finite buffers", *Int. J. Prod. Res.*, 50 (15), 4009-4024, (2012).
- Xu, Z. and Cai, Y., "Variable neighborhood search for consistent vehicle routing problem", *Expert Syst. Appl.*, 113, 66-76, (2018).

Xuemei, L., Huan, S., Rui, Z., Yongqi, J. and Aiping, L., “Collaborative optimization of transfer line balancing and buffer allocation based on polychromatic set”, *Procedia CIRP*, 63, 213-218, (2017).

Yu, V. F. and Maghfiroh, M. F. N., “A variable neighborhood search with path-relinking for the capacitated location routing problem”, *J. Ind. Prod. Eng.*, 31 (3), 163-176, (2014).

Zandieh, M., Joreir-Ahmadi, M. N., and Fadaei-Rafsanjani, A., “Buffer allocation problem and preventive maintenance planning in non-homogenous unreliable production lines”, *Int. J. Adv. Manuf. Technol.*, 91, 2581-2593, (2017).

Zhang, M. and Matta, A., “Models and algorithms for throughput improvement problem of serial production lines via downtime reduction”, *IISE Trans.*, 52 (11), 1189-1203, (2020).

Zhang, X. Y., Zhou, J. and Wei, L., “Multi-Objective hybrid heuristic algorithm to buffer allocation in unreliable production line”, *Comput. Eng. Appl.*, 50, 240-244, (2014).

Zhou, B., Liu, Y., Yu, J. and Tao, D., “Optimization of buffer allocation in unreliable production lines based on availability evaluation”, *Optim. Contr. Appl. Met.*, 39, 204-219, (2018).

Zulj, I., Kramer, S. and Schneider, M., “A hybrid of adaptive large neighborhood search and tabu search for the order-batching problem”, *Eur. J. Oper. Res.* 264, 653-664, (2018).

10. ÖZGEÇMİŞ

Adı Soyadı : Mehmet Ulaş Koyuncuoğlu
Doğum Yeri ve Tarihi : Gürün – 10.02.1990
Lisans Üniversite : Pamukkale Üniversitesi
Y. Lisans Üniversite : Pamukkale Üniversitesi
Elektronik posta : ulas@pau.edu.tr; mehmetulask@gmail.com
İletişim Adresi : Pamukkale Üniversitesi, Bilgi İşlem Daire Başkanlığı, 20160, Denizli

Yayın Listesi :

- Koyuncuoğlu, M. U. and Demir, L., “Buffer capacity allocation in unreliable production lines: An adaptive large neighborhood search approach”, *Eng. Sci. Technol. an Int. J.*, <https://doi.org/10.1016/j.jestch.2020.08.012>, (2020).
- Koyuncuoğlu, M. U. and Demir, L., “A comparison of combat genetic and big bang-big crunch algorithms for solving the buffer allocation problem”, *J Intell. Manuf.*, <https://doi.org/10.1007/s10845-020-01647-1>, (2020).
- Demir, L. and Koyuncuoğlu, M. U., “The impact of the optimal buffer configuration on production line efficiency: A VNS-based solution approach”, *Expert Syst. Appl.*, Revised, (2020).
- Koyuncuoğlu, M. U., “Bir Devlet Üniversitesinin Mühendislik Fakültesi Mezunlarının Kazançlarının Çeşitli Değişkenler Açısından İncelenmesi”, *Mehmet Akif Ersoy Üniv. Sos. Bilim. Enst. Derg.*, 9 (20), 314-326, (2017).
- Taner, M. E., Kulak, O. and Koyuncuoğlu, M. U., “Layout analysis affecting strategic decisions in artificial container terminals”, *Comput. Ind. Eng.*, 75, 1-12, (2014).

Konferans listesi :

- Koyuncuoğlu, M. U. and Demir, L., “A Big Bang-Big Crunch Algorithm for Buffer Allocation Problem”, 10th International Symposium on Intelligent Manufacturing and Service Systems (IMSS'19) (Tam Metin Bildiri/Sözlü Sunum), (2019).
- Koyuncuoğlu, M. U. and Demir, L., “A combat genetic algorithm for optimal buffer allocation in unreliable production lines”, The International Conference on Data Science, Machine Learning and Statistics 2019 (DMS-2019) (Özet Bildiri/Sözlü Sunum), (2019).
- Demir, L. and Koyuncuoğlu, M. U., “A new initialization procedure for adaptive LNS in solving buffer allocation problem”, 20th EU/ME Workshop on Metaheuristics. (Özet Bildiri/Sözlü Sunum), (2019).
- Koyuncuoğlu, M. U. and Demir, L., “Large Neighborhood Search Algorithm for Buffer Allocation Problem”, 29th European Conference on Operational Research (Özet Bildiri/Sözlü Sunum), (2018).
- Koyuncuoğlu, M. U. and Demir, L., “The Effect of the Initial Solution Algorithms Used in VNS for Solving Buffer Allocation Problem”, Global Joint Conference On Industrial Engineering and Its Application Areas (Özet Bildiri/Sözlü Sunum), (2018).
- Demir, L., Koyuncuoğlu, M. U., Köse-Yelkenci, S. and Tunalı, S., “Solving Buffer Allocation Problem for a Heat Exchanger Company: A Variable Neighborhood Search Based Approach”, 19th EU/ME Workshop on Metaheuristics for Industry (Tam Metin Bildiri/Sözlü Sunum), (2018).
- Sarıoğlu, D. Ö., Ören, A. and Koyuncuoğlu, M. U., “Lojistik Programı Öğrencilerinin Eğitimlerine ve Mesleklerine Bakışı”, EYİ 2018 (Özet Bildiri/Sözlü Sunum), (2017).
- Yeşilyurt M. E., Elbi M. D., Emrouznejad, A., Koyuncuoğlu, M. U., Şahin, E., Yeşilyurt, F. and Kızılkaya, A., “Computing Single Outputs for DEA”, 15th International Conference on Data Envelopment Analysis (Özet Bildiri/Sözlü Sunum), (2017).
- Koyuncuoğlu, M. U. and Demir, L., “A Variable Neighborhood Search Based Heuristic for Buffer Allocation Problem in Production Lines”, The 14th International Symposium on Operational Research in Slovenia (Tam

Metin Bildiri/Sözlü Sunum). (SCOPUS, Computer Science Index, DOAJ), (2017).

- Taner, M. E., Koyuncuoğlu, M. U. and Yılmaz, Y., Talaşlı İmalat Yapan Bir Üretim Tesisinde Hücreyel Üretim Sistemi ve Fonksiyonel Yerleşim Düzenlemesinin Simülasyon ile Analizi. Üretim Araştırmaları Sempozyumu'16 (Tam Metin Bildiri/Sözlü Sunum), (2016).
- Yaldır, A., Koyuncuoğlu, M. U. and Demir, B., “Üniversiteler İçin Öğrenci Topluluk Yönetim Bilgi Sistemi Uygulaması”, Akademik Bilişim 2016 (Tam Metin Bildiri/Sözlü Sunum), (2016).
- Yaldır, A., Koyuncuoğlu, M. U. and Çakır, E., “Üniversiteler İçin Proje Yönetim Bilgi Sistemi”, Akademik Bilişim 2016 (Tam Metin Bildiri/Sözlü Sunum), (2016).
- Koyuncuoğlu, M. U. Küçük, M. A., Kılıç, G., Zerey, S. and Oral, T. U., “Üniversiteler İçin Süreç Yönetimi Bilgi Sistemi”, 35. Yöneylem Araştırması - Endüstri Mühendisliği Kongresi (Tam Metin Bildiri/Sözlü Sunum), (2015).
- Yaldır, A., Koyuncuoğlu, M. U. Yanmaz-Arpacı, Ö., Taner, M. E., Özgür-Polat, L. and Yildirim, T., “Process Management Information System for Higher Education Institutions: A Case Study of Pamukkale University”, International Conference on Quality in Higher Education'15 (Tam Metin Bildiri/Sözlü Sunum), (2015).
- Yaldır, A., Özgür-Polat, L., Taner, M. E., Yanmaz-Arpacı, Ö., Yildirim, T., Koyuncuoğlu, M. U., “A Case Study on Institutional Data Assessment Systems for Universities and their Benefits”, International Conference on Quality in Higher Education'15 (Tam Metin Bildiri/Sözlü Sunum), (2015).
- Supçiller, A. A., Koyuncuoğlu, M. U. and İçin, Y., “Toplam Üretkenlik Artışının Simülasyon ile Analizi”, Üretim Araştırmaları Sempozyumu'15 (Tam Metin Bildiri/Sözlü Sunum), (2015).
- Yılmaz, Y., Kulak, O., Taner M. E. and Koyuncuoğlu, M. U., “A Hybrid Algorithm for Quay Crane Scheduling Problem”, 26th European Conference on Operational Research (Tam Metin Bildiri/Sözlü Sunum), (2013).

- Koyuncuoğlu, M. U., Kulak, O., Taner, M. E. and Polat, O., “A Novel Approach for Effective Solution of Yard Crane Scheduling at Seaport Container Terminals”, International Conference on Logistics and Maritime Systems (Tam Metin Bildiri/Sözlü Sunum), (2012).
- Taner, M. E., Kulak, O., Polat, O. and Koyuncuoğlu, M. U., “Comparison of Various Layout Types at Short Sea Container Terminals”, International Conference on Logistics and Maritime Systems (Tam Metin Bildiri/Sözlü Sunum), (2012).
- Koyuncuoğlu, M. U., Kulak, O., Taner, M. E. and Polat, O., “Konteyner Terminallerinde İstif Vinçlerinin Etkin Çizelgelenmesine Yeni Bir Yaklaşım”, Üretim Araştırmaları Sempozyumu'12 (Tam Metin Bildiri/Sözlü Sunum), (2012).
- Koyuncuoğlu, M. U., Kulak, O., Polat, O. and Taner, M. E., “Yard Crane Scheduling by Using a Genetic Algorithm at Seaport Container Terminals”, EURO Conferences XXV (Tam Metin Bildiri/Sözlü Sunum), (2012).
- Kulak, O., Taner, M. E., Polat, O., Koyuncuoğlu, M. U., “AGV Applications for Short Sea Container Terminals”, IX. International Logistics & Supply Chain Congress (Tam Metin Bildiri/Sözlü Sunum), (2011).
- Supçiller, A. A. and Koyuncuoğlu, M. U., “Dondurulmuş Gıda Sektöründe Faaliyet Gösteren Bir Firmanın Tedarik Zinciri Dağıtım Ağı Stratejilerinin Simülasyon ile Analizi”, 2. Uluslararası Ekonometri Yöneylem Araştırması ve İstatistik Sempozyumu-(EYİ), (12), 20-28. (Tam Metin Bildiri/Sözlü Sunum), (2011).
- Güngör, A., Çapraz, O. and Koyuncuoğlu, M. U., “Bir Üniversite Kampus Alanının Engelli Dostluğu Düzeyinin Tespiti”, 17. Ulusal Ergonomi Kongresi (Tam Metin Bildiri/Sözlü Sunum), (2011).
- Koyuncuoğlu, M. U., Polat O., Taner, M. E. and Kulak, O., “Dolgu Tipi Konteynır Terminalleri için AGV Dağıtım Uygulamaları”, 31. Yöneylem Araştırması ve Endüstri Mühendisliği Ulusal Kongresi (Tam Metin Bildiri/Sözlü Sunum), (2011).

- Kulak, O., Taner, M. E., Koyuncuođlu, M. U. and Cerit H. B., “Bir Konteynır Terminalinde Taşıyıcı Araç Tiplerinin Karşılaştırılması”, 10. Ulusal Üretim Araştırmaları Sempozyumu (Tam Metin Bildiri/Sözlü Sunum), (2010).