

**ESNEK İŞLEMLİ AKIŞ ATÖLYELERİNDE İŞLEM ATAMASI VE PARÇA
SIRALAMASI**

SEDA SUCU

**YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ**

**TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

AĞUSTOS 2013

ANKARA

Fen Bilimleri Enstitü onayı

Prof. Dr. Necip CAMUŐCU

Müdü

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

Prof. Dr. Tahir HANALIOĐLU

Anabilim Dalı Başkanı

SEDA SUCU tarafından hazırlanan ESNEK İŐLEMLİ AKIŐ ATÖLYELERİNDE İŐLEM ATAMASI VE PARŐA SIRALAMASI adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. Hakan GÜLTEKİN

Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Yrd. Doç. Dr. Sibel ALUMUR ALEV

Üye : Yrd. Doç. Dr. Hakan GÜLTEKİN

Üye : Doç. Dr. Muhammed Fatih DEMİRCİ

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Seda SUCU

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü : Fen Bilimleri
Anabilim Dalı : Endüstri Mühendisliği
Tez Danışmanı : Yrd. Doç. Dr. Hakan GÜLTEKİN
Tez Türü ve Tarihi : Yüksek Lisans – Ağustos 2013

Seda SUCU

ESNEK İŞLEMLİ AKIŞ ATÖLYELERİNDE İŞLEM ATAMASI VE PARÇA SIRALAMASI

ÖZET

Bu çalışmada 2 veya 3 makineli n adet farklı tip parça üreten akış tipi üretim sistemleri ele alınmıştır. Sistemdeki her parça akış tipi üretim sisteminin gereği olarak her makinede aynı sırayla işlenmektedir. Her parçanın sistemdeki her makinede işlenmesi gereken belirli işlemler vardır. Bu işlemler "sabit" işlemlerdir. Sabit işlemlerin yanında sistemdeki belirli ardışık makineler tarafından işlenebilen ve esnek işlem adı verilen bir işlem daha bulunmaktadır. Çalışmada ele alınan 2 makineli sistemlerde her parçanın işlenmesi gereken üç farklı işlem vardır. İlk işlem sadece birinci, üçüncü işlem sadece ikinci makinede işlem görebilirken ikinci işlem her iki makinede de işlenebilen esnek işlemdir. 3 makineli sistemlerde esnek işlem sayısı ve esnek işlemin işlenebildiği makineler açısından değişik alternatifler ele alınmıştır. Ayrıca bu bölümde makinelerin özdeş olmadığı, yani esnek işlemin süresinin atandığı makineye göre değiştiği durum ele alınmıştır. Çalışmadaki amaç, yayılma hızını enküçükleyecek şekilde hem parça sıralamasına ve hem de esnek işlemlerin işleneceği makinelere karar vermektir. Problemin çözümü için öncelikle kesin çözüm yöntemi olarak karma tam sayılı matematiksel model geliştirilmiştir. Fakat matematiksel modelin çözüm süresinin uzun olması nedeniyle, kaliteli sonuçlar veren polinom zamanlı sezgiseller geliştirilmiştir. Deneysel çalışma yapılarak geliştirilen çözüm yöntemlerinin performans analizi gerçekleştirilmiştir.

Anahtar Kelimeler: Akış atölyesi, çizelgeleme, esnek operasyonlar, sezgisel yöntemler.

University : TOBB University of Economics and Technology
Institute : Institute of Natural and Applied Sciences
Science Programme : Industrial Engineering
Supervisor : Asst. Prof. Hakan GÜLTEKİN
Degree Awarded and Date : M.Sc. – August 2013

Seda SUCU

PART SCHEDULING AND OPERATION ALLOCATION IN A FLEXIBLE FLOWSHOP

ABSTRACT

In this study, we considered flowshops with 2 and 3 machines producing n different parts. As a requirement of the flowshop assumption, parts are processed on the machines in the same order. Each machine performs a specific operation on each of the parts. These operations are called as "fixed" operations. Apart from these fixed operations, there are some operations which can be performed by specific adjacent machines. These operations are named as "flexible" operations. For the 2-machine case, every part has three operations: The first operation can only be processed on the first machine and the third operation can only be processed on the second machine. Whereas, the second operation is the flexible one can be performed by both machines. For 3-machines systems different alternatives are considered regarding the number and the position of the flexible operations. Machines are assumed to be nonidentical meaning that the processing time of the flexible operation depends on the machine that is assigned to. The objective is to determine the part input sequence as well as the assignment of the flexible operations in order to maximize the throughput rate. For each problem alternative, mixed integer programming formulations are developed. However, since the solution times appeared to be very large, a number of polynomial time heuristic algorithms are developed. The performances of these heuristics are tested through experimental studies.

Keywords: Flowshops, scheduling, flexible operations, heuristics.

TEŞEKKÜR

Öncelikle, yüksek lisans eğitimim süresince, engin bilgi ve deneyimleriyle çalışmalarımı yönlendiren, bilimsel ve mesleki anlamında bana rehberlik eden, yardımlarını hiçbir zaman esirgemeyen çok değerli tez danışmanım ve hocam Yrd. Doç Dr. Hakan Gültekin'e sonsuz teşekkür ve saygılarımı sunarım. Önderliği ve destekleri benim için her zaman çok değerli olacaktır.

Bilgi ve deneyimlerinden yararlandığım jüri üyelerimden Yrd. Doç Dr. Sibel ALUMUR ALEV'e, tezimi okuyarak verdiği değerli tavsiyeler için samimiyetle teşekkür ederim. Tez savunmamdaki önemli görüş ve önerilerinden dolayı jüri üyelerimden Doç. Dr. Muhammed Fatih DEMİRCİ'ye teşekkürlerimi sunarım.

Yüksek lisans eğitimim sırasında herbirinin engin tecrübe ve bilgilerinden faydalandığım TOBB Ekonomi ve Teknoloji Üniversitesi Endüstri Mühendisliği Bölümü öğretim üyelerine teşekkürü bir borç bilirim.

Lisans ve yüksek lisans eğitimim süresince dostluğuyla her zaman yanımda olan dostum Özlem Ardıç'a, arkadaşlığımızın ilk gününden bu yana maddi ve manevi desteklerini hiç bir zaman esirgemeyen dostluklarıyla beni her zaman sevindiren çok sevgili arkadaşlarım Tuba Özsevinç, Aynur Gülsoy ve Burak Ataş'a gönülden teşekkür ederim.

İki sene boyunca birlikte ders aldığımız, aynı ofisi ve bölümü paylaştığımız başta proje arkadaşım Fatma İşler ve ofis arkadaşlarım olmak üzere tüm bölüm arkadaşlarıma, yardıma ihtiyacım olduğunda beni kırmadan yardımına koşan kuzenim Berker Demirer'e ve arkadaşım Çağlar Kavak'a, ayrıca 110M489 numaralı projedeki maddi desteği için TÜBİTAK'a teşekkür ederim.

En önemlisi, fedakarlıklarına her zaman hayran olduğum ve bana duydukları güvenden güç aldığım, beni bugünlere getiren Annem Yücel Sucu ve Babam Haluk Sucu'ya, neşesi ve tatlılığıyla sabırsız zamanlarımın sakinleştiricisi en büyük destekçim olan, kardeşliğin anlamını hissettiren Canım Kardeşim Arda Sucu'ya sonsuz teşekkürlerimi sunarım.

İÇİNDEKİLER

1	GİRİŞ	1
2	LİTERATÜR	4
2.1	Akış Atölyelerinde Çizelgeleme	4
2.2	Montaj Hattı Dengeleme	10
2.3	Üretim Sistemlerinde Esneklik	12
2.4	Özet	17
3	PROBLEM TANIMI VE MATEMATİKSEL MODEL	19
4	İKİ MAKİNE TEK ESNEK İŞLEME SAHİP SİSTEMLER	27
4.1	Çözüm Yöntemi	27
4.2	DeneySEL Çalışma	34
4.3	Sonuçlar	40
5	ÜÇ MAKİNE TEK ESNEK İŞLEME SAHİP SİSTEMLER	42
5.1	Esnek İşlemin Makine 2 ve 3 Üzerinde İşlenebildiği Durum	42
5.1.1	Sezgisel Çözüm Yöntemleri	43
5.1.2	DeneySEL Çalışma	59
5.1.3	Sonuçlar	68
5.2	Esnek İşlemin Üç Makine Üzerinde İşlenebildiği Durum	69

5.2.1	Sezgisel Çözüm Yöntemleri	70
5.2.2	Deneyisel Çalışma	72
5.2.3	Sonuçlar	81
6	ÜÇ MAKİNE İKİ ESNEK İŞLEME SAHİP SİSTEMLER	83
6.1	Sezgisel Çözüm Yöntemleri	85
6.2	Deneyisel Çalışma	87
6.3	Sonuçlar	99
7	SONUÇ ve DEĞERLENDİRME	101
	KAYNAKLAR	104
	ÖZGEÇMİŞ	110

ŞEKİLLERİN LİSTESİ

3.1	Problem 1: 2 Özdeş Makine Tek Esnek İşlemlili Sistemler	21
3.2	Problem 2: 3 Özdeş Olmayan Makine, 2. ve 3. Makinelerde Tek Esnek İşlem	22
3.3	Problem 2 Ters: 3 Özdeş Olmayan Makine, 1. ve 2. Makinelerde Tek Esnek İşlem	22
3.4	Problem 3: 3 Özdeş Olmayan Makine, 3 Makinede de İşlenebilen Tek Esnek İşlem	23
3.5	Problem 4: 3 Özdeş Olmayan Makine, 2 Esnek İşlem	23
4.1	Problem 1: Johnson Karşılaştırma Gantt Şeması	34

TABLoların LİSTESİ

2.1	İki Veya Daha Fazla Makine Akış Tipi Çizelgeleme Problemleri	7
4.1	Problem 1: CPLEX Çözüm Durumları	36
4.2	Problem 1: İş Sayısına Göre Sezgisel - Alt Sınır Yüzde Fark	36
4.3	Problem 1: İş Sayısına Göre Sezgisel En iyi CPLEX % Fark	37
4.4	Problem 1: 20 İş Sezgisel Ve CPLEX Ortalama Çözüm Süresi	37
4.5	Problem 1: 50 İş Sezgisel Ve CPLEX Ortalama Çözüm Süresi	37
4.6	Problem 1: Faktör Tasarımı	38
4.7	Problem 1: CPLEX Çözüm Süresi Faktörleri	39
4.8	Problem 1: CPLEX Alt Sınır Sezgisel Yüzde Fark Faktörleri	40
5.1	Problem 2: Alternatif Gupta Algoritmaları	44
5.2	Örnek 2: Parçalara Göre İşlem Süreleri	48
5.3	Örnek 2: Modifiye Gupta Algoritmalarına Göre Esnek İşlem Atamaları	48
5.4	Örnek 2: Modifiye Gupta Algoritmalarına Göre Parça Sıralaması	48
5.5	Örnek 2: Modifiye Gupta Algoritmalarına Göre Yayılma Zamanı Değerleri	49
5.6	Problem 2: Alternatif CDS Algoritmaları	49
5.7	Örnek 3: Modifiye CDS Algoritmalarına Göre Esnek İşlem Atamaları	54
5.8	Örnek 3: Modifiye CDS Algoritmalarına Göre 1. Yapay Problem Parça Sıralaması	54

5.9	Örnek 3: Modifiye CDS Algoritmalarına Göre 2. Yapay Problem Parça Sıralaması	55
5.10	Örnek 3: Modifiye CDS Algoritmalarına Göre Yayılma Zamanı Değerleri	55
5.11	Problem 2: Alternatif NEH Algoritmaları	56
5.12	Örnek 4: Parçalara Göre NEH Kriterleri	57
5.13	Örnek 4: Farklı NEH Kriterlerine Göre Parça Sıralaması	59
5.14	Baskın Olan Algoritma Alternatifleri	60
5.15	Problem 2: 20 İş İçin Sezgisellerin Alt Sınırda Yüzde Uzaklığı	61
5.16	Problem 2: 20 İş İçin Sezgisellerin En İyi Sonuca Yüzde Uzaklığı	61
5.17	Problem 2: Min_NEH Optimal Çözüm Sayısı	62
5.18	Problem 2: CPLEX Zaman Limitine Takılan Problem Sayısı	62
5.19	Problem 2: İş Sayısına Göre Min_NEH Alt Sınır Yüzde Fark	62
5.20	Problem 2: İş Sayısına Göre Min_NEH En İyi CPLEX Yüzde Fark	63
5.21	Problem 2: 20 İş Min_NEH Sezgiseli Ve CPLEX Ortalama Çözüm Süreleri	63
5.22	Problem 2: 50 İş Min_NEH Sezgiseli Ve CPLEX Ortalama Çözüm Süreleri	63
5.23	Problem 2: Faktör Tasarımı	64
5.24	Problem 2: CPLEX Çözüm Süresi Faktörleri	65
5.25	Problem 2: CPLEX Alt Sınır CPLEX En İyi Yüzde Fark Faktörleri	66
5.26	Problem 2: CPLEX Alt Sınır Sezgisel Yüzde Fark Faktörleri	67
5.27	Problem 2: CPLEX En İyi Sezgisel Yüzde Fark Faktörleri	68

5.28	Problem 3: Alternatif NEH Algoritmaları	70
5.29	Problem 3: CPLEX Çözüm Durumları	73
5.30	Problem 3: 20 İş İçin Min_NEH Alt Sınır Yüzde Fark	73
5.31	Problem 3: 50 İş İçin Min_NEH Alt Sınır Yüzde Fark	74
5.32	Problem 3: İş Sayısına göre Min_NEH Alt Sınır Yüzde Fark	75
5.33	Problem 3: İş Sayısına göre Min_NEH En İyi CPLEX Yüzde Fark	75
5.34	Problem 3: İş Sayısına göre Min_NEH Optimal Çözülen Problem Sayısı	76
5.35	Problem 3: 20 İş için Sezgisel ve CPLEX Çözüm Süreleri	76
5.36	Problem 3: 50 İş için Sezgisel ve CPLEX Çözüm Süreleri	77
5.37	Problem 3: Faktör Tasarımı	77
5.38	Problem 3: CPLEX Çözüm Süresi Faktörleri	78
5.39	Problem 3: CPLEX Alt Sınır CPLEX En İyi Yüzde Fark Faktörler	79
5.40	Problem 3: CPLEX Alt Sınır Sezgisel Yüzde Fark Faktörleri	80
5.41	Problem 3: CPLEX En İyi Sezgisel Yüzde Fark Faktörleri	81
6.1	Problem 4: Alternatif NEH Algoritmaları	85
6.2	Problem 4: CPLEX Çözüm Durumları	90
6.3	Problem 4: 20 İş İçin Sezgiseller - Alt Sınır Yüzde Fark	90
6.4	Problem 4: 50 İş için Sezgiseller - Alt Sınır Yüzde Fark	91
6.5	Problem 4: İş Sayısına göre Min NEH Alt Sınır Yüzde Fark	92
6.6	Problem 4: İş Sayısına göre Min NEH En İyi CPLEX Yüzde Fark	92
6.7	Problem 4: İş Sayısına göre Min NEH Optimal Çözülen Problem Sayısı	93

6.8	Problem 3 20 İş için Sezgisel ve CPLEX Çözüm Süreleri	93
6.9	Problem 3 50 İş için Sezgisel ve CPLEX Çözüm Süreleri	94
6.10	Problem 4: Faktör Tasarımı	95
6.11	Problem 4: CPLEX Çözüm Süresi Faktörleri	96
6.12	Problem 4: CPLEX Alt Sınır CPLEX En İyi Yüzde Fark Faktörler . .	97
6.13	Problem 4: CPLEX Alt Sınır Sezgisel Yüzde Fark Faktörleri	98
6.14	Problem 4: CPLEX En İyi Sezgisel Yüzde Fark Faktörleri	99

1. GİRİŞ

Küreselleşen dünya, gelişen teknolojiler, uluslararası pazar rekabetleri nedeniyle, esnek üretim sistemleri ortaya çıkmıştır. Esnek üretim ile birlikte verimlilik, üretkenlik ve kalite de artmıştır. Esnek üretim sistemine yönelik çalışmalar son 50 yılda hız kazanmıştır. Numerik kontrollü ilk makine 1965'te ortaya çıkmıştır. İlerleyen yıllarda CNC (Bilgisayar Sayımlı Yönetim) makineleri üretimde önem kazanmıştır. Bu makinelerde yapılacak küçük değişimler ile farklı parçaların üretimine olanak sağlanabilir. Bu kavram gelişmiş birçok esnek otomasyon, CNC makineleri dahil olmak üzere bilgisayar kontrollü dağıtım, otomatik malzeme taşıma ve depolama gibi sistemlerin entegre halidir.

Esnek üretim, CAD (bilgisayar destekli tasarım) ve CAM (bilgisayar destekli üretim) gibi araçlarla müşteri beklentilerine hızlıca yanıt vermektedir. Bunun yanında bu üretimde otomasyona bağlı sistem sayesinde işçilik maliyetlerinde azalma, hataların daha rahat farkedilmesi ve ürün standardizasyonu gibi yararlar edinilmektedir. Bu tarz avantajlar rekabeti arttırmaktadır. Diğer yandan, bu entegre sistemlerin kurulum maliyetleri çok yüksektir. Aynı zamanda sisteme uyumlu yazılımların geliştirilmesi, bu sistemleri kullanım bilgisine sahip işçi bulma gibi gereklilikleri vardır. Bu gibi harcamaların faydaya dönüştürülebilmesi için kaynakların daha dikkatli kullanılması gerekmektedir. Bu üretim sistemlerinin sağladığı avantajlardan yararlanabilmek için üretimde planlama ve çizelgeleme problemlerine verilen önem artış göstermiştir.

Esnekliğin kullanılabileceği farklı alanlar ve çok çeşitli esneklik türleri mevcuttur. Bu nedenle daha verimli sonuçlar elde etmek için esnekliğin kullanacağı alanı, esnekliğe ayrılacak maliyet ve esneklik seviyesinin belirlenerek bu esneklikle faydanın en büyüklenmesi için üretim planının belirlenmesi gerekmektedir. Çizelgeleme teknikleri bu noktada büyük öneme sahiptir.

Bu çalışmada operasyon esnekliğine sahip 2-3 makineli akış tipi üretim sistemlerinde çıktı optimizasyonu üzerine çalışılmıştır. Üretilen n adet farklı parçanın sistemdeki

makinelere sırayla geçerken tamamlamaları gereken işlemler vardır. Bu işlemlere ek olarak birden fazla makine üzerinde işlenebilen işlemler vardır. Bu işlemler esnek işlem olarak adlandırılır ve bu işlemlerin işlem görebildiği makineler ise esnek makinelerdir. Esnek makineler daha önceden belirlenmiştir.

Farklı tip parçaların makineler üzerinde farklı işlem sürelerine sahip olmaları nedeniyle, bu çalışmada ele alınan problemler için parça sıralaması önemlidir. Aynı zamanda her parça için esnek işlemlerin hangi makineye atanacağı da önemli bir karardır. Çalışmadaki amaç en son işlenecek parçanın, son makinedeki işleminin tamamlanma zamanını en küçüklemeektir. Parça sıralaması ve esnek işlemlerin ataması amaç fonksiyonunu optimize edecek şekilde belirlenmelidir.

Parçaların makineler üzerindeki işlem süreleri stokastik parametrelere bağlı değildir. Önceden belirlenmiş ve sabittir. Herhangi bir parçanın işlem gördüğü makine üzerinde işlemi tamamlanmadan o makineden çıkarılmaz. İş kesmeye izin verilmez. Bununla birlikte bir makine üzerinde parçanın işlemi bitmesine rağmen bir önceki makine üzerindeki parçanın işlemi bitmediyse işlemi tamamlanan makine yeni parçayı işlemeye başlayamaz. Sistemdeki işlerin bölünmesine izin verilmemektedir.

Sistemdeki makineler arasında sınırsız ara stok olduğu varsayılmıştır. Üretim çıktısının televizyon, beyaz eşya gibi büyük elektronik aletler olması durumunda, makineler arasında stok alanının sınırlı olduğu düşünülebilir fakat PCB kart gibi üzerinde küçük devre elemanlarının yerleştirildiği sistemlerde, yeterli stok alanı bulunabilmektedir. Bu nedenle küçük parçalar için bloke olma durumu sınırsız ara stok bulunması nedeniyle engellenebilmektedir.

Çalışmada ele alınan problemlerin gerçek hayat uygulamalarına sıkça rastlanmaktadır. Esnek üretimden elektronik, otomotiv ve beyaz eşya üretimleri başta olmak üzere bir çok alanda yararlanılmaktadır. CNC makineleri yüksek esnekliğe sahip makinelerdir. Bu makinelerin kullanıldığı sanayi, tarım, otomotiv, baskı ve matbaa, pres işleri gibi alanlarda, PCB kart üzerindeki montaj sisteminde, çapraz eğitimli işçilerin çalıştığı montaj hatlarında benzer problemlere rastlanmaktadır.

Bölüm 3'te tanımlı detaylı şekilde verilecek olan problemlerin çözümü için öncelikle karma tamsayı matematiksel modeller geliştirilmiştir. Çalışmaya konu olan problemlerden çoğunun NP-Zor olması nedeniyle, geliştirilen matematiksel modeller

ile çözüme ulaşmak için çok fazla zamana ihtiyaç duyulduğu gözlemlenmiştir. Bu nedenle problemlere sezgisel çözüm yöntemleri ile yaklaşarak etkili sonuçlara ulaşılması hedeflenmiştir. Önerilen sezgiseller optimal çözümü garanti etmemektedir fakat kısa sürede yanıt veren kaliteli sonuçlara ulaşmaktadır.

Yedi bölümde incelenen bu tez çalışmasında bir sonraki bölümde incelenen probleme yönelik literatür taramasına yer verilmektedir. Bölüm 3'te ele alınan problemlerin tanımı ve problemlere yönelik genel bir matematiksel model bulunmaktadır. 4, 5 ve 6'ncı bölümler, esnek işlem sayısına ve sistemde bulunan makine sayısına göre değişim gösteren problemleri detaylı bir şekilde incelemektedir. Bu bölümler 3 ana kısma ayrılarak öncelikle önerilen çözüm yöntemleri sunulmuş, sonrasında çözüm yöntemlerine ait performans analizi gerçekleştirilmiş ve son olarak elde edilen sonuçlardan bahsedilmiştir. 7'nci bölüm çalışmada elde edilen sonuçlara ait yorumları kapsamaktadır. Aynı zamanda, probleme yönelik yapılabilecek gelecek çalışmalara dair öneriler de 7'nci bölümde yer almaktadır.

2. LİTERATÜR

Birinci bölümde anlatıldığı gibi, bu çalışmada 2 ve 3 makineli esnek operasyonlu akış tipi üretim sistemlerinde tamamlanma zamanı en küçüklenmesi problemi ele alınmaktadır. Literatürde yer alan çizelgeleme problemlerine yönelik çalışmaların başlangıcı 1950'li yıllarda temellenmiştir. Akış tipi çizelgeleme problemleri 60 yıl içerisinde çokça çeşitlilik göstermiştir. Çalışmalar, tek makineli çizelgeleme problemleri ile başlamıştır ve seri makinelerden oluşan akış atölyesi, iş atölyesi ve açık atölye tipi üretim sistemlerine yönelik çizelgeleme problemleri ile devam etmiştir. Esnek işleme olanak sağlayan iş gücü ve makinelerin kullanımının yaygınlaşması, esnek işlemleri üretim sistemlerini konu alan çalışmaların yapılmasına ve ilerletilmesine olanak sağlamıştır.

Bu bölümün içerisinde yer alan alt bölümlerde çalıştığımız konunun literatürdeki yeri ve bu konu ile ilgili yapılan çalışmalardan bahsedilmektedir. Literatür çalışmasında temel alınan konular, klasik akış atölyelerinde çizelgeleme, üretim sistemlerinde esnek operasyonlar ve montaj hattı dengeleme olmak üzere üç ana başlıkta incelenecektir.

2.1 Akış Atölyelerinde Çizelgeleme

Akış atölyelerinde işler üretim sisteminde bulunan tüm makineleri aynı sıra ile ziyaret etmektedir. İşlem süreleri makinelere ve parçalara göre değişim gösterebilmektedir. Bu tip üretim sistemleri Gupta ve Stafford [35] tarafından sürekli ve bölünemeyen işlerin bir seri makine üzerindeki akışı olarak da tanımlanmıştır. Bu tip üretim sistemlerinin çizelgelenmesinde bir çok farklı amaç fonksiyonunun en iyilenmesi üzerine çeşitli çalışmalar yapılmıştır. Bu çalışmada da ele alınan ve son sıradaki işin sistemdeki son makinedeki tamamlanma zamanı olarak tanımlanan yayılma zamanı (makespan) en küçüklenmesi literatürde önemli bir yere sahiptir. Bizim bu çalışmada ele aldığımız

problem 2 ve 3 makineli sınırsız ara stok alanına sahip akış tipi üretim ortamı üzerinden ilerlemektedir.

Akış tipi üretim sistemlerinde çizelgeleme ile ilgili literatürde yer alan çalışmaların başlangıcı Johnson'ın [39] 2 makineli akış tipi üretim sistemi için geliştirdiği optimal sonucu veren algoritma ile başlamıştır. Bu algoritma 2 makine akış atölyesi problemlerini polinom zamanda çözebilmektedir. Johnson [39] algoritmasında, öncelikle makine 1'deki işlem süreleri makine 2'deki işlem sürelerinden kısa olan işler bir kümede (küme L); makine 1'deki işlem süreleri makine 2'deki işlem sürelerinden uzun olan işler başka bir kümede (küme R) toplanır. İkinci adımda küme L 'de bulunan işler makine 1'deki işlem zamanlarına göre SPT (küçük işlem süresinden büyük işlem süresine doğru) kuralına göre, küme R 'de bulunan işler de makine 2'deki işlem zamanlarına göre LPT (büyük işlem süresinden küçük işlem süresine doğru) kuralına göre dizilmektedir. Son adımda L kümesindeki sıralanmış işlerin ardına R kümesindeki sıralanmış işler eklenerek optimal iş sırası elde edilmektedir. Johnson [39] algoritması 3 makineli akış atölyelerinde işlem sürelerinin özel bir durumu için de optimal sonuç verebilmektedir.

Literatürde akış tipi üretim sistemi çizelgeleme ile ilgili bir çok araştırma makalesi yer almaktadır. Hejazi ve Saghafian [38] akış tipi çizelgeleme problemi için geliştirilen temel algoritmaları, Johnson [39] algoritmasının dinamik programlama yorumunu ve metasezgiselleri derledikleri bir literatür taraması hazırlamışlardır. Gupta ve Stafford [35] akış tipi çizelgeleme problemlerinin Johnson'dan [39] başlayarak ivme kazanan serüveninin elli yıl içerisinde nasıl gelişim gösterdiğini özetlemişlerdir. Akış tipi üretim sistemlerinde bazı durumlarda parçalar beklemeden işlenmek durumundadır. Bu tarz üretim sistemleri "beklemesiz ortam" olarak bilinmektedir. Bununla birlikte "bloke olma" durumu da bazı akış tipi üretim sistemlerinde dikkate alınmaktadır. Bloke olma durumunda, bir işin işlemi bir makinede tamamlandıktan sonra bir sonraki işlem göreceği makine uygun değilse (başka bir iş işlenmekte ise) ve makineler arasında ara stok alanı yok veya o sırada dolu ise işlem gördüğü makinede beklemek zorundadır. Bu durumda, bu makineye başka bir parça yüklenemeyeceği için bloke olmuştur. Hall ve Sriskandarajah [37] beklemesiz ve bloke olma özelliğine sahip akış tipi üretim sistemlerini ele alan çalışmaları derlemişlerdir. Bizim ele aldığımız problemlerde makinelerdeki ayar zamanları, her parça için aynı kabul edildiğinden dolayı işlem sürelerinin içerisinde olduğu varsayılmaktadır. Fakat literatürde, iş

sırasına ve makineye göre deęişen ayar zamanlarının olduęu problemleri ele alan çok sayıda çalışma mevcuttur. Ayar zamanını dikkate alarak geliştirilmiş çalışmalar Allahverdi v.d. [2] tarafından işlerin sıralamaya baęlı olup olmayışına ve parti üretimi yapılıp yapılmamasına göre dört gruba ayrılarak derlenmiştir.

Çizelgeleme problemlerinin notasyonuna yönelik çalışma ilk defa Graham v.d. [27] tarafından gerçekleştirilmiştir. Bu çalışmada problem gösterimi üç bölüm ($\alpha|\beta|\gamma$) içermektedir. Makine ortamı α ile, kısıtlar β ile ve amaç fonksiyonu γ ile gösterilmektedir. Bu çalışmada ele alınan makine ortamı akış atölyesidir. Bu ortam Graham v.d. [27] tarafından yapılmış olan notasyona göre F ile gösterilmektedir. Akış tipi problemler kısıtlara ve amaç fonksiyonlarına göre deęişim göstermektedir. Çalışmada odaklanılan amaç fonksiyonu yayılma zamanının enküçüklenmesidir. Bu amaç fonksiyonun gösterimi C_{max} şeklinde yer almaktadır. Literatürde yer alan ve bu çalışma içinde bahsedilen farklı kısıtların gösterimi ve anlamları:

1. pmtn: iş kesme
2. no wait: beklemez ortam
3. no idle: makinelerde boş zamana izin verilmez
4. proportionate: işlem zamanlarının belirli bir oranla deęişim göstermesi şeklindedir.

İki ve daha fazla makineye sahip akış tipi üretim sistemlerinde çizelgeleme üzerine literatürde çeşitli çalışmalar yer almaktadır. Yayılma zamanını en küçükleme odaklı akış tipi çizelgeleme problemlerinin varyasyonları ve bu çalışmalardan elde edilen sonuçlar Tablo 3.1'de yer almaktadır.

Problem	Karmaşıklık	Kaynaklar	Sonuçlar
$F_2 C_{max}$	$O(n \log n)$	Johnson [39]	Polinom zamanda optimal çözüm veren algoritma
$F_2 pmtn C_{max}$	$O(n \log n)$	Gonzalez ve Sahni [26]	Johnson [39] algoritması ile optimal çözüm
$F_2 r_j, pmtn C_{max}$	Güçlü NP-Zor	Cho ve Sahni [15]	Problemin güçlü NP-Zor olduğu kanıtı
$F_3 C_{max}$	Güçlü NP-Zor	Garey v.d. [22]	3 makine akış tipi problemlerinde yayılma zamanı minimizasyonu güçlü NP-Zor olduğu ispatı
$F_3 pmtn C_{max}$	Güçlü NP-Zor	Gonzalez ve Sahni[26]	Öncelik ilişkisi olan veya olmayan akış atölyeleri ve iş atölyelerinde sistemlerinde yayılma zamanı enküçükleme problemlerinin NP-Tam ispatı ve iyi sonuçlar veren sezgiseller
$F_m no - wait C_{max}$		Wisner [64]	n iş, m makine beklemesiz akış tipi çizelgeleme probleminde yayılma zamanını en küçükleyen algoritma (TSP yaklaşımı)
$F_2 no - wait C_{max}$	$O(n \log n)$	Gilmore ve Gomory [24]	TSP yaklaşımı ile polinom zamanlı algoritma
$F_3 no - wait C_{max}$	Güçlü NP-Zor	Röck [50]	Problemin güçlü NP-Zor olduğunun ispatı
$F_3 no - idle C_{max}$	Güçlü NP-Zor	Baptiste ve Hguny [6]	Problemin güçlü NP-Zor olduğu kanıtı
$F_m no - idle C_{max}$	Güçlü NP-Zor	Saadani v. d. [51]	Problemin TSP modeli ve en yakın komşuluk sezgiseli ile çözümü
$F_m proportionate C_{max}$	$O(n \log n)$	Shacklevich v. d. [57]	İşlem zamanları belirli bir oranla birbirine bağlı olan m makineli akış tipi çizelgeleme problemini polinom zamanlı çözen algoritma

Tablo 2.1: İki Veya Daha Fazla Makine Akış Tipi Çizelgeleme Problemleri

2 makineden daha fazla makine içeren akış tipi üretim sistemlerinin çizelgelenmesi için bu problemlerin güçlü NP-Zor olduğu ispatlanmıştır [22] ve bu problemleri kısa zamanda çözebilmek için bir çok sezgisel yöntem geliştirilmiştir. Bu sezgisel yöntemlerden performans olarak öne çıkanlar CDS [14], Gupta [32], NEH [42] ve Palmer [44] algoritmalarıdır. Bu algoritmalar bu çalışmada geliştirilen çeşitli sezgisellere temel oldukları için bu bölümde detaylı bir şekilde açıklanacaktır.

1. **Palmer algoritması [44]**: Her bir iş için, makine sayısına bağlı olarak değişen bir eğim dizini hesaplamıştır. Bu dizinler büyükten küçüğe sıralanmaktadır. İşler, bu şekilde elde edilen sıralamaya göre çizelgelenir. Bu algoritmanın altında yatan mantık küçük işlem süresine sahip işlerin daha önce işlenmeye eğilimli olmasıyla alakalıdır. Palmer [44], algoritmasında kullandığı eğim indeksini

hesaplarken aşağıda belirtilen formülü kullanmıştır;

$f(i)$ = i işinin eğim indeksi

t_{im} = i işinin makine m 'deki işlem süresi

M = Sistemdeki toplam makine sayısı

$$f(i) = - \sum_{m=1}^M [M - (2m - 1)] \frac{t_{im}}{2}$$

2. **CDS algoritması [14]** : Campbell, Dudek ve Smith [14] tarafından geliştirilmiştir. m makine probleminden $(m - 1)$ tane yapay iki makine problemi elde edilmektedir ve bu problemler Johnson algoritması ile çözülmektedir.

t'_{i1} = Yapay problem için 1. makedeki işlem süresi

t'_{i2} = Yapay problem için 2. makedeki işlem süresi

$$t'_{i1} = \sum_{j=1}^k t_{ij}$$

$$t'_{i2} = \sum_{j=1}^k t_{ij}$$

$k = 1$ 'den $k = m$ değerine kadar her k değeri için t'_{i1} ve t'_{i2} işlem süreleri yenilenmektedir ve her yenilenen işlem süreleri için Johnson [39] algoritması uygulanarak yayılma zamanı değeri hesaplanmaktadır. Bu şekilde oluşturulan problemler Johnson algoritmasıyla çözülmektedir. $(m - 1)$ problem arasından en küçük yayılma zamanını veren sıralama, CDS algoritması için de çözüm olmaktadır.

3. **Gupta algoritması [32]** : Gupta [32]'nin yaklaşımı Palmer [44] algoritmasına benzerlik göstermektedir. Bu algoritmada kullanılan eğim hesabında işlerin ilk makedeki ve son makedeki işlem sürelerinin birbirlerine göre büyüklükleri dikkate alınmaktadır. Gupta [32] algoritması önemli bir büyüklükteki durumlar için Palmer [44] algoritmasından daha iyi sonuçlar vermektedir [38]. Gupta [32] algoritmasında kullanılan eğim indeksi aşağıda belirtilen formülle belirlenmektedir;

$f(i) = i$ işinin eğim indeksi

$t_{im} = i$ işinin makine m 'deki işlem süresi

$$A = \begin{cases} 1, & t_{im} \leq t_{i1} \\ -1, & \text{diğer durumlarda} \end{cases}$$

$$f(i) = \frac{A}{\min_{1 \leq m \leq M-1} (t_{im} + t_{i(m+1)})}$$

4. **NEH Algoritması [42] (1983):** Nawaz v.d. [42] tarafından geliştirilmiştir. Algoritma, 5 işten 500 işe; 5 makineden 25 makineye kadar optimale yakın sonuçlar vermektedir. Her i işi için tüm makinelerdeki işlem zamanları toplanır ve işler bu toplama göre artmayan şekilde sıralanır. İlk adımda elde edilen sıralamaya göre ilk iş 1., ikinci iş 2. ve ilk iş 2., ikinci iş 1. sırada olacak şekilde iki alternatif sıralama ele alınır. Bunlardan daha küçük yayılma zamanına sahip olan sıralama sabitlenir. İlk adımda bulunan sıralama göz önünde tutularak 3. sıradaki işten n . sıradaki işe kadar bütün işler benzer şekilde, sabitlenmiş sıralamadaki bütün olası pozisyonlara denenir. Sıradaki işin eklenmesi, k tane olası durum oluşturur ve alternatifler arasından yayılma zamanı en küçük olan seçilir ve sıralama sabitlenir.

Literatürde akış tipi çizelgeleme problemleri yukarıda bahsedilenler dışında farklı amaç fonksiyonlarıyla da incelenmiştir. Bu amaç fonksiyonlarından bazıları, ortalama tamamlanma zamanı en küçüklenmesi, toplam gecikme en küçüklenmesi, toplam tamamlanma zamanı en küçüklenmesi, maksimum gecikme zamanı problemleridir. Literatürde birden fazla amaç fonksiyonuna sahip akış tipi çizelgeleme problemleri Parveen ve Ullah [47] tarafından bir tarama makalesinde toplanmıştır. Bununla birlikte [2], [35], [37], [38] tarama makalelerinden farklı amaç fonksiyonlarına sahip akış tipi çizelgeleme problemleri ile ilgili çalışmalar incelenebilir. Bu çalışmalara ek olarak Pan ve Ruiz [46] akış tipi üretim sisteminde toplam ve ortalama akış ve tamamlanma zamanını en küçüklemek için oluşturulmuş sezgiselleri deney tasarımı modelleri ile karşılaştırarak anlatmış ve yeni sezgiseller önermişlerdir.

Bu alt bölümde akış tipi çizelgeleme ile ilgili literatürde bugüne kadar yapılmış ve ele alınan problemle ilgili önemli çalışmalardan bahsedilmiştir. Bir sonraki bölüm bu

çalışmada ele alınan problemle yakından ilgili bir başka önemli problemin, montaj hattı dengeleme probleminin literatürünü taramaktadır.

2.2 Montaj Hattı Dengeleme

Montaj hattı, bir ürünün ortaya çıkmasını sağlayan parçaların birleştirildiği bir üretim hattıdır. Bu hat genel olarak hareketlidir veya bir taşıyıcı sistemden oluşur. Bu sistem sayesinde, birbiri ardına eklenen iş istasyonları parça üretimini gerçekleştirir. Her istasyonda, geçerli istasyonun kullanılabilir oluşunu sağlayan ekipmanlar bulunmaktadır. Aynı zamanda, bu hatlar akış tipi üretim sistemlerinin özel durumlarıdır [9]. İlk montaj hattı örneği Henry Ford tarafından kurulan montaj hattıdır [12]. Henry Ford'un 1900'lü yıllarda önerdiği bu montaj hattı günümüze kadar çeşitlendirilerek geliştirilmektedir. Maliyet fonksiyonları, ekipman seçimi, paralel istasyonlu hatlar, U-şekilli montaj hatları ve karma model üretimi gibi genel montaj hattı problemlerine çözümler üretilmiştir. Konu ile ilgili bir çok sorun tespit edilmesine ve modellenmesine rağmen gelişmiş çözüm yöntemleri henüz bulunmamaktadır [9].

Montaj hattı dengeleme, kaynak ve kapasite kısıtları göz önüne alınarak ürün çıktısını en büyükleyecek şekilde işlerin istasyonlara atamasını belirleme problemidir. Montaj hattı dengeleme ile ilgili ilk matematiksel modelleme 1955 yılında Salveson [54] tarafından yapılmıştır. Montaj hattında parçalar bir hat üzerinde art arda sıralanmaktadır ve bu hat parçalar sıranın sonuna ulaşana kadar hareket etmelerini sağlamaktadır. Parçalara istasyonlardan birinde bulunurken belirlenmiş bir işlem kümesi uygulanır. Bu şekilde bir istasyona giren iki parça arasında uzun dönemde geçen ortalama süre çevrim zamanı olarak adlandırılmaktadır [12]. Bir başka tanım da sistemden çıkan ardışık iki parça arasında uzun dönemde geçen ortalama süre şeklindedir.

Montaj hattı dengeleme problemleri Tip-1 ve Tip-2 olarak sınıflandırılmıştır. Her iki alternatifte de işlem süreleri, işlemler arası öncelik ilişkileri gibi sistem parametreleri bilinmektedir. Tip-1 montaj hattı dengeleme probleminde çevrim zamanı verilmiştir. Buna göre minimum istasyon kullanacak şekilde işlemlerin istasyonlara atanması bulunmaya çalışılır. Tip-2 montaj hattı dengeleme probleminde ise istasyon

sayısı bilinirken minimum çevrim zamanını elde etmek üzere işlemlerin yapılacağı istasyonlara karar verilmeye çalışılır.

Baybars [7] Tip-1 ve Tip-2 MHD (Montaj Hattı Dengeleme) problemleri üzerine bir tarama makalesi yayınlamıştır. Bu çalışmada, MHD problemleri üzerine modifikasyonlar ve genellemeler kronolojik bir sırayla irdelenmiş ve formülasyonlarla birlikte kesin çözüm yöntemleri önerilmiştir. Montaj hattı dengeleme problemleri için olurlu bir çözüm bulmanın NP-Zor olduğunun kanıtı Wee ve Magazine [63] tarafından gösterilmiştir. Tip-1 montaj hattı dengeleme problemi için alt sınırlar, baskınlık kuralları, dinamik programlama ve dal-sınır algoritmaları geliştirilmiştir. Yapılan çalışmalar kesin ve geçerli eşitsizlikler bulmayı sağlamasına rağmen dal-sınır algoritmasına dayanan sezgisel kıyaslanamayacak kadar iyi sonuçlar vermiştir, [55]. Tip-2 montaj hattı dengeleme problemlerine çözüm olarak Klein ve Scholl [40] dal-sınır algoritması önermiştir. Bu iki tip problem için üretilen çözümlere ek olarak bazı sezgisel yöntemler de geliştirilmiştir. Boctor [11] , Tip-1 için geliştirilen sezgisellere yönelik bir analiz yapmış ve bir sezgisel yöntem önermiştir. Tip-2 için geliştirilen sezgiseller Hackman v.d. [36], Liu v.d. [41], Rekiek v.d. [49] ve Ugurdag v.d. [60]'nin çalışmalarından incelenebilir.

Son yıllarda yapılmış çalışmalardan biri Anuar ve Buchkin'in [3] ele aldığı dinamik montaj hatlarıdır. Dinamik montaj hattında ardışık iş istasyonları arasında esnek işlem bulunabilmektedir. Paylaşılan görev bir istasyonun son elemanı veya o istasyonu takip eden istasyonun ilk elemanı olarak yer alabilmektedir. Dolayısıyla dinamik montaj hattı dengeleme probleminde de karar verilmesi gereken noktalar; esnek işlem sayısının ve atamanın belirlenmesidir [3]. Dinamik montaj hattı dengeleme problemi, esnek işlem atamasına karar verilmesi açısından bizim ele aldığımız problemle benzerlik göstermektedir. Fakat bizim problemimizde tanımlanan esneklik farklıdır. Bizim problemimizde işler paylaşılabilir. Sadece bir işlem birden fazla makine üzerinde işlem görebilmektedir. Aynı zamanda, işlerin iş istasyonlarına atanmasına değil sıralamalarına karar verilmelidir.

Bu çalışmaya benzer bir çalışma Guo v.d. [31] tarafından geliştirilmiştir. Esnek montaj hattı problemi ele alınmıştır. Montaj hattındaki boş zamanı ve çevrim zamanını en küçükleme için matematiksel model verilmiştir. Bununla birlikte iki aşamalı bir genetik algoritma önerilmiştir. Bu algoritma iş istasyonlarına işlem atamasına

ve esnek işlemin farklı iş istasyonlarındaki görev paylaşımına karar vermek için geliştirilmiştir. Esnek montaj hattı dengeleme probleminin çözümü için önerilen optimizasyon modelinin verimliliğini göstermek için deney sonuçları kullanılmıştır. Guo v.d. [31] tarafından ele alınan bu çalışmanın bizim çalışmamızdan farkı paralel makinelerin kullanılması ve esneklik tanımının farklı olmasıdır.

Bir başka çalışma Vincent ve Ponnambalam [62] tarafından gerçekleştirilmiştir. Esnek montaj hattı çizelgeleme problemini diferansiyel evrim algoritması ile çözerek genetik algoritmayla kıyaslamıştır ve bu yöntemin daha yüksek performansa sahip olduğu sonucuna varılmıştır. Gong v.d. [25] çalışmalarında, kısmi çapraz eğitilmiş çalışanlara sahip U-şekilli bir montaj hattı problemini ele almışlardır. Çalışanlardan birinin olmadığı durumda birbiri yerine çalışabilecek çalışanlar performans düşüklüğünü engelleyebilmektedir. Çalışmada U-şekilli montaj hattında çalışan işçilerin eğitimi için yeni bir politika önerilmiştir ve klasik eğitim politikası ile kıyaslanarak bazı yönetsel kavramlar ortaya çıkarılmıştır.

2.3 Üretim Sistemlerinde Esneklik

Esneklik, mikroişlemci teknolojilerin gelişmesiyle birlikte tasarım, operasyon ve üretim yönetimi konuları üzerinde önemli bir konu haline gelmiştir [56]. Bir sistemin esnekliği, geniş bir çeşitliliğe sahip olan çevrelere adapte olabilmesidir [33]. Bu bölümde, çalışmada ele alınan esnekliğin üretimdeki yeri ve bizim problemimizde göz önünde bulundurulmuş esneklik ile ilgili çalışmaların detaylı anlatımı yer almaktadır.

Bu çalışmada esnekliğin üretim sistemlerindeki tanımı öne çıkmaktadır. Üretimde esneklik, farklı ürünleri gerekli kalitede verimli bir şekilde üretmek için üretim kaynaklarının yeniden düzenlenebilmesidir [56]. Üretimde esneklik ile ilgili yapılan çalışmalar geçen 30 yıl içinde önemli ölçüde artmıştır [53]. Literatürde bu konuyla ilgili yapılmış çalışmalar Beach'ın [8], Gerwin'in [23] ve Sethi ve Sethi'nin [56] tarama makalelerinden incelenebilir.

Günümüze kadar yapılan çalışmalar 50'nin üzerinde esneklik tanımı var olduğunu söylemektedir. Literatürde yer alan bu tanımların naiv tanımlar olduğu ve zamanla değişebilecekleri belirtilmektedir. Bununla birlikte, esnekliğin kullanımı açısından

esneklik çeşitlerinin içiçe geçebilme durumları olduğu belirtilmiştir ([10], [16], [56] ve [58]). Esneklik konusunda temel alınan Sethi ve Sethi [56]'nin makalesinde esneklik türleri 11 başlık altında; makine, malzeme elleçleme, operasyon, süreç, ürün, rotalama, hacim, büyüme, program, üretim ve pazar esnekliği adlarıyla incelenmiştir. Bu belirtilen türlerden makine ve operasyon esnekliği bizim çalışmamızda ele alınan esneklik türleridir. Bu türlerin kısaca tanıtılması konunun daha iyi takip edilmesine yardımcı olacaktır.

Makine esnekliği, Sethi ve Sethi'nin [56] tanımına göre, farklı türden operasyonların bir engel içermeden bir makinede işlenebilmesini sağlayan esnekliktir. Verilmiş bir parça kümesini üretmek için makine üzerinde gereken değişiklikleri yapabilme kolaylığı olarak da bilinmektedir [13]. Bir üretim sisteminde makine esnekliği; araç ve parça yükleme gibi cihazların teknolojik gelişimiyle, farklı parçaların işlenmesine olanak sağlayan değişik kesme aletlerinin tek bir makinede bulunmasını sağlayan teknolojiyle ve düzgün bir işlem atamasının getirisiyle kesici alet değişiminin enküçüklenmesiyle elde edilir [13]. Nümerik kontrol (NC) ve bilgisayar nümerik kontrol (CNC) cihazları makine esnekliğini kullanan en önemli teknolojiler arasındadır.

Operasyon esnekliği ise bir parçanın farklı şekillerde üretilebilmesi ile ilgilidir ve başka esneklik türlerinin oluşmasına katkı sağlamaktadır. Operasyon esnekliğine sahip parçaların üretimi CAD/CAM, bilgisayarlı süreç planlama ve grup teknolojisi gibi sistemlerin yardımıyla gerçekleşmektedir [56]. Bununla birlikte Browne v.d. [13] operasyon esnekliğinin parçaların çizelgelenmesini kolaylaştırdığını söylemektedir.

Literatürde, esnekliğin getirilerini gösteren bir çok çalışma yapılmıştır. Askin ve Chen [4] iş sıralamasının bilindiği fakat iş gücünün kısmen çapraz eğitilmiş olduğu durumu ele almışlardır. Bazı işler iş istasyonları arasında değişebilmektedir. Bu çalışmadaki esneklik çapraz eğitilmiş işçilerle sağlanmaktadır. Bu esnekliğin ölçülebilir derecede daha fazla çıktı sağladığı gözlenmiştir. Mc. Clain v.d. [43] esnek işlemlerin stok alanı bulunmaması durumunda bile verimliliği arttırdığını göstermiştir.

Ruiz-Torres vd. [52], operasyon esnekliğinin akış tipi üretim sisteminde toplam yayılma zamanını en küçükleme ve iş istasyonu kullanımı konusundaki yararlarını araştırmıştır. Buradaki operasyon esnekliği işlerin istasyonlara atanması ile ilgilidir.

Çalışma bu yönüyle bizim çalışmamıza benzerlik göstermektedir. Fakat bizim çalışmamızda operasyon sayısı farklıdır. Aynı zamanda operasyonlar farklı makinelerde işlendiklerinde işlem zamanları makineye göre değişim göstermektedir. Ruiz ve Torres'in ele aldığı çalışmada ise işlem süresi sadece operasyona göre farklılık göstermektedir. Ek olarak, bu çalışmada [52] operasyon sayısı ve makine sayısı oranı önemli bir yere sahiptir. Çalışmada bu problemi çözmek için iki fazdan oluşan bir sezgisel önerilmiştir. İlk faz atamayı, ikinci faz ise sıralamayı gerçekleştirmektedir. Yapılan deneysel çalışmalar, problemin karmaşıklığı arttıkça, operasyon esnekliğinin sağladığı yararların arttığına işaret etmektedir. Bununla birlikte ilerleyen bölümlerde yer verilen montaj hattı dengeleme problemleri ile akış tipi üretim sisteminde operasyon esnekliği arasında operasyonların iş istasyonlarına atanması problemi yönünden benzerlik olduğu söylenebilmektedir.

Tang ve Tomlin [59] esnekliğin tedarik zincirine getirisi ve esnekliğin ne kadar olması gerektiği üzerine bir çalışma düzenlemişlerdir. Çalışmaya göre yüksek dereceli esnekliğin riski azaltmayacağı ve düşük seviyede esnekliğin daha çok yarar sağlayacağı kanısına varılmıştır. Jordan ve Graves [21] süreç esnekliğinin faydalarını anlatmak için esnekliğin ne kadar ve ne şekilde kullanılması gerektiğinden ve bunların getirdiği analitik sonuçlardan bahsetmişlerdir. Esnekliğin doğru oranda ve doğru şekilde uygulanması halinde, yatırım ve üretim maliyetini düşürmeye yardımcı olduğu sonucuna varmışlardır.

Daniels ve Mazzola [19] kaynak esnekliğinin bir işlemin işlem süresinin büyük bir kısmı, ayrılan kaynak miktarı ile ters olarak değişen sistemlerde avantaj getirdiğini yaptıkları hesaplamalarla belirtmişlerdir. Kaynak esnekliği normal akış tipine göre yapılan ölçümlerle %81 oranında performans artışı göstermiştir. Daniels v.d. [20] akış tipi üretim sisteminde kısmi kaynak esnekliğini ele almışlardır. İşçi esnekliğine odaklanarak, buna uygun bir model formüle etmişlerdir. Yaptıkları hesaplamalar sonucu işçi esnekliğinin getirdiği avantaj çapraz eğitime yapılan küçük yatırımlarla elde edilmektedir. Rastgele uygulanan kısmi esnekliğin esnek olmayan sistemden daha kötü sonuçları getirdiği kanısına varıldığı için; çizelgeleme problemi, işçi ve kaynak atama problemlerinin aynı anda ele alınmasının çok daha iyi sonuçlar vereceği önerisinde bulunmuşlardır.

Yukarıdaki alıntılara bağlı olarak esnekliğin bir çok çeşidi olduğu ve bu çeşitlerin

farklı yararları olduğu söylenebilmektedir. Bu bağlamda, bizim çalışmamıza konu olan esnek operasyonlu akış tipi üretim sistemi için de bu durum geçerlidir. Esnek operasyonlar birden fazla makine tarafından işlenebilmektedir, bu da üretilen çıktı miktarının artmasına olanak sağlamaktadır. Literatürde, esnek operasyonlu akış tipi çizelgeleme problemlerine yönelik çalışmalar son yıllarda önem kazanmıştır.

Crama v.d. [17] baskılı devre (PCB) montajı üzerine üretim planlama en iyilemesini sağlamaya çalışmışlardır. Burada ele alınan problemde, çözüm PCB kart tipinin makine gruplarına atanması, bileşenlerin makinelere yerleştirilmesi ve optimal ürün sıralaması üzerine olduğu için atama ve sıralama problemi olması yönüyle çalışmamıza benzerlik göstermektedir. Fakat bizim çalışmamızda akış atölyesinde parçaların esnek işlem ataması gerçekleşmektedir. Bu çalışmadaki [17] atama makine gruplarına atamayı hedeflemektedir.

Vairaktarakis ve Lee [61] iki aşamalı sistemde, çok işlemcili akış tipi üretim sisteminde yayılma zamanı en küçüklenmesi üzerine çalışmışlardır. n iş ve her iş için iki operasyon bulunmaktadır. Birinci (ikinci) operasyon ilk (ikinci) makinede işlenebilmekte veya her iki operasyon da iki makinede ortak olarak işlenebilmektedir. Problemin çözümü için işlerin makinelerde nasıl ve hangi sıralamayla işleneceğine karar verilmelidir. Çalışmada, problemi optimal olarak çözen dinamik programlama yöntemi önerilmiş, optimalite koşulları belirtilmiş ve yöntemin performansı ölçülmüştür.

Aktürk v.d. [1] operasyon esnekliğini iki makine robotik hücre çizelgeleme probleminde kullanmışlardır. Çalışmada yer alan özdeş parçalar belirli sayıda operasyona sahiptir. Sistemdeki makineler tüm operasyonları yapabilme özelliğine sahiptir. Bu özelliklere sahip bir üretim ortamında, minimum çevrim zamanını elde etmek amacıyla, operasyon atamasına ve optimal robot döngüsüne karar vermişlerdir. Benzer bir çalışma Gültekin v.d. [28] tarafından kesici uç haznesi kısıtı göz önünde bulunduralarak düzenlenmiştir. Bu kısıtla birlikte bazı operasyonlar sadece ilk, bazı operasyonlar da sadece ikinci makinede işlem görebilmektedir. Kalan operasyonlar için her iki makinede de işlenebilme durumu söz konusudur. Buradaki operasyon esnekliği, robotik hücrelerin içindeki makinelerin CNC makinesi olmasından kaynaklanmaktadır. Dolayısıyla, problemin çözümü kalan operasyonların doğru makineye atanmasıyla ilgilidir. Çalışmanın sonucu olarak her döngünün optimal

olduğu parametre aralıkları belirlenmiş ve duyarlılık analizi yapılmıştır. Gültekin v.d. [29] özdeş parçaların üretimini sağlayan 2-3 makineli akış tipi çizelgeleme problemi ele alınmıştır. Bu üretim sisteminde malzeme taşıma robotlar tarafından gerçekleştirilmektedir. Performans ölçütleri çevrim zamanı ve toplam üretim maliyeti olan bu problem için robot hareket sırası ve işlem zamanlarına karar vermişlerdir.

Babayan ve He [5] n iş ve 3 basamaklı esnek akış tipi üretim sistemi için ajan-temelli yaklaşım ile çözüm üretmişlerdir. Geliştirilen yaklaşım, tamamlanma zamanını enküçüklemeye yöneliktir. Bu problemde esneklik işlerin her basamakta özdeş olan herhangi bir makine tarafından işlenebilmesiyle sağlanmıştır. Önerilen çözüm yöntemine yönelik performans analizi alt sınıra bağlı hata oranı ile test edilmiştir.

Gupta v.d. [34] 3 operasyona sahip 2 makine akış tipi üretim sisteminde yayılma zamanını en küçüklemeyi hedeflemişlerdir. İlk operasyon birinci makinede ve son operasyon ikinci makinede işlenmektedir. Ancak, ikinci operasyon birinci veya ikinci makinede işlenebilen esnek bir işlemdir. Çalışmada NP-Zor olan bu probleme polinom zamanlı, biri rastgele sıralamaya ve biri rastgele atamaya dayanan iki yaklaşım algoritması geliştirilmiştir ve bu algoritmaların performans analizi gerçekleştirilmiştir. Bu problemdeki esneklik bizim çalışmamızda kullandığımız esneklik tanımıyla aynıdır fakat esnek operasyon sayısı ve sistemde bulunan makine sayısı farklıdır.

Gultekin [30] esnek işleme sahip iki makine akış tipi üretim sisteminde özdeş işlerin üretiminde çıktı optimizasyonu üzerine çalışmıştır. Üç operasyondan biri ilk makinede, diğeri ikinci makinede işlenebilmektedir ve esnek olan işlem ise iki makinede birden işlem görebilmektedir. Makineler özdeş olmamaları nedeniyle, işlem süreleri makineye göre değişim göstermektedir. Bu nedenle problemin çözümü için, çıktı optimizasyonunu sağlayacak esnek işlem atamasına karar vermek ve esnekliğin seviyesini belirlemek önemlidir. Çalışmanın ilk kısmında, ara stok alanının sıfır ve sınırsız olduğu durumlar için çözüm yöntemleri sunulmuştur. İkinci kısımda ise, esnekliğin yararları gösterilmiştir. Esneklik seviyesi çok düşük iken bile, yayılma zamanında azalma olduğu sonucuna varılmıştır. Buna ek olarak, esneklik seviyesine karar vermek için yayılma zamanı ve maliyet kriterlerinin aynı anda göz önünde bulundurulması gerektiği gösterilmiştir.

Crama ve Gültekin [18] özdeş parçalar üreten esnek operasyonlu 2 makine akış

tipi üretim sisteminde çıktı optimizasyonu üzerinde çalışmışlardır. Ele alınan üretim sisteminde ilk (ikinci) işlem birinci (ikinci) makinede, esnek işlem ise her iki makinede de işlenebildiği durum ele alınmıştır. Bu özellikler bağlamında problemin çözümü esnek işlem atamasına karar vermek üzerine kurulmuştur. Makineler arasındaki stok kapasitesinin bulunmadığı, belirli bir kapasiteye sahip olduğu veya sınırsız stok alanının bulunduğu durumlar ayrı ayrı değerlendirilmiştir. Aynı zamanda üretilen parça sayısının sonlu ve sonsuz olduğu durumlar ayrı ayrı göz önüne alınmıştır. Farklı stok kapasitesi ve farklı parça sayıları için polinom zamanlı çözüm yöntemleri sunulmuştur.

Crama ve Gultekin [18], Gultekin [30] ve Gupta vd. [34] esneklik tanımı ve akış tipi üretim sistemine sahip olmaları açısından bizim ele aldığımız problemlere benzerlik göstermektedir. Fakat bizim çalışmamızda parçalar ve makineler özdeş değildir. Sabit ve esnek işlem süreleri işlendikleri makineye ve parçaya göre değişim göstermektedir. Aynı zamanda bu çalışmalar esnek ve sabit işlem sayısı bakımından da bizim çalışmamızdan farklıdır. Bizim ele aldığımız problemler sistemdeki makine sayısına bağlı olarak iki veya üç tane sabit ve en az bir esnek işleme sahiptir. Esnek işlemler her zaman birinci ve ikinci makine arasında yer almak durumunda değildir. Ek olarak, stok alanı sınırsız kapasiteye sahiptir.

2.4 Özet

Bu bölümde, literatür akış tipi üretim sistemlerinde çizelgeleme, montaj hattı dengeleme ve üretim sistemlerinde esneklik konuları altında üç ayrı bölümde taranmıştır.

Bu tez çalışmasında ele alınan probleme en çok benzerlik gösteren problemler, Anuar ve Bukchin [3], Crama ve Gultekin [18], Gultekin [30] ve Gupta v.d. [34] tarafından yapılan çalışmalarda incelenmiştir. Bu çalışmalardaki problem tanımı, kullanılan esneklik türü bizim araştırdığımız problemlerle benzerlik göstermektedir. Bu tez çalışması, Gupta v.d. [34] tarafından gerçekleştirilen çalışmanın üç makinenin bulunduğu ve esnek işlem sayısının daha farklı olduğu ve Crama ve Gultekin'in [18] ele aldığı iki makineli sistemde farklı tip parça üretiminin gerçekleştiği sistemleri araştırmaktadır. Bununla birlikte, Crama ve Gultekin'in [18] çalışmasında incelenen

problem Gupta v.d. [34] tarafından ele alınan probleme benzerlik göstermektedir fakat bu çalışmada problemler özdeş işler üzerinden çeşitlendirilerek incelenmiştir. Gultekin [30] ise esnek işlem sürelerinin makineye göre değişim gösterdiği problem üzerine çalışmıştır ve polinom zamanlı çözüm algoritmaları geliştirmiştir.

Bu tez çalışmasında da çözüm yöntemi aranan problem için esnek işlemler makineye göre farklılık göstermektedir ve buna ek olarak işler de özdeş değildir. Aynı zamanda problem 3 makineli akış tipi üretim sistemi için genişletilmiştir. İncelenen probleme çözüm yöntemi olarak esnek işleme sahip olmayan 3 makine akış tipi üretim sistemi için geliştirilen çözüm yöntemleri temel alınmıştır. Esnek işlemlerin çıktısı en büyüklenmesi üzerine olan etkisi literatürde yapılan çalışmalarca açıkça görülmektedir. Fakat literatürde bu zamana kadar yapılan çalışmalarda bu tez çalışmasını konu alan bir çalışma bulunmamaktadır. 3 makineye sahip ve bazı makinelerin birden fazla işlem yapabildiği akış tipi üretim sistemleri için problemin karmaşıklığı ve çözümü hakkında fikirler sunmak adına bu çalışma önem arz etmektedir.

Bir sonraki bölümde problemin tanımı ve matematiksel model yer almaktadır.

3. PROBLEM TANIMI VE MATEMATİKSEL MODEL

Çalışmanın bu bölümünde ele alınan problem detaylarıyla birlikte anlatılacak ve problemlere yönelik matematiksel modeller verilecektir. Çalışmada esnek işlemlerin sayısına ve esnek işlemlerin işlenebildiği makinelere göre farklılık gösteren dört farklı problem ele alınmıştır. Çalışılan problemlerin genel tanımı, özellikleri ve bu problemler için oluşturulmuş ortak matematiksel model bu bölümde verilmektedir.

Genel anlamda problemler özdeş olmayan 2 veya 3 makineden oluşan bir akış tipi üretim sisteminde yayılma zamanı en küçükleme amaçlı çizelgeleme problemidir. Ele alınan üretim sistemlerinin hepsinde n adet farklı tip parçanın üretimi gerçekleştirilmektedir. 2-makine sistemde her parça için 3 işlem bulunmakta iken 3-makine sistemlerde esnek işlem sayısının bir veya iki olduğu farklı alternatifler ele alınmıştır. Akış tipi sistemlerde, tüm parçalar her makineden aynı sırada geçmektedir. Bu nedenle, örnek olarak iki makine bir akış tipi üretim sisteminde, üretim gerçekleşirken her parça önce 1. sonra 2. makinede işlem görmektedir. Bu işlemler sabit işlemlerdir. Dolayısıyla, 2-makine sistemlerde 2 sabit işlem, 3-makine sistemlerde 3 sabit işlem bulunmaktadır. Tüm problemler için makineler üzerindeki sabit işlem süreleri her parça için farklıdır. Aynı zamanda, her parça sabit işlemlerine ek olarak birden fazla makinede işlenebilen işlemlere sahiptir. Bu işlemler esnek işlem olarak adlandırılır. 2-makine sistemde tek bir esnek işlem var iken 3-makine sistemde tek veya iki esnek işlem bulunabilmektedir.

Esnek işlemler önceden belirlenmiş alternatif makinelerden birinde işlenebilmektedir. Esnek işlemlerin işlem gördüğü makineler esnek makine olarak adlandırılmaktadır. Sistemdeki esnek makineler özdeş olabilir veya olmayabilir. Makinelerin özdeş olması durumunda, esnek işlem, atandığı makineden bağımsız olarak aynı işlem süresine sahiptir. Makinelerin özdeş olmaması durumunda ise, esnek işlemler atandığı makineye bağlı olarak değişik değerler alabilir. Bu durum, sistemdeki makinelerin

farklı teknolojik özelliklere sahip olması durumunda ortaya çıkmaktadır. Bu çalışmada özdeş makine durumu ele alınmıştır. 3-makinelili sistemlerde ise daha genel olan özdeş olmayan makine durumu ele alınmıştır. Bu değişim makineler arasındaki teknolojik farklılıklara bağlı olabilir. Çalışmada ele alınan tüm sistemler için işlem süreleri deterministik bir yapıya sahiptir.

Sabit ve esnek işler için iş kesmeye izin verilmemektedir. Her parçanın esnek işlemi esnek makinelerden sadece biri üzerinde gerçekleştirilebileceği için esnek işlemin hangi makineye atanmış olduğu problemin çözümü için büyük önem taşımaktadır. Ek olarak, parçaların özdeş olmaması nedeniyle parçaların optimal sıralamasına karar verilmelidir.

Bu çalışmadaki tüm problemler için makineler arasında sınırsız stok alanı bulunduğu varsayılmaktadır. n adet parçanın üretildiği bir sistemde $n - 1$ stok alanının bulunması sınırsız ara stok alanını temsil etmektedir. Belirli bir makine üzerinde parçalardan biri işleme devam ederken, işlemi tamamlanan diğer parçaların bu parçayı bekleyebilecekleri bir stok alanı bulunmadığı durumda işlem gördükleri makine üzerinde beklemeleri gerekmektedir. Beklemesi gereken parça sayısının işlem gören parça hariç sistemdeki bütün parçalar olduğu durumda, en fazla stok alanına ihtiyaç duyulacaktır. Dolayısıyla, n parçanın üretileceği bir sistemde $n - 1$ kapasiteli stok alanının bulunması, ara stok alanının sınırsız kabul edilmesi için yeterlidir. Bu varsayım, parça işlendikten sonra makinenin o parçayı bırakabilmesini sağlamaktadır.

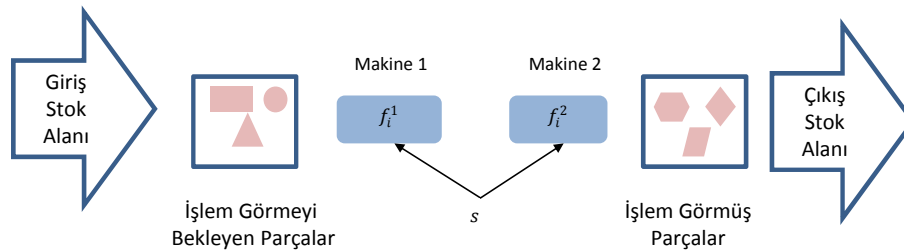
Çalışmadaki amaç hem parçaların hangi sırada işleneceğini belirlemek ve hem de her parçanın esnek işleminin işleneceği makineyi belirlemektir. Yayılma zamanı (makespan) sıralamadaki en son parçanın sistemdeki en son makinede tamamlandığı an olarak tanımlanmıştır. Yayılma zamanının en küçüklenmesi aynı zamanda üretim çıktı miktarının en büyüklenmesini sağlamaktadır. Akış tipi üretim sistemlerinde makineler üzerindeki işlem süreleri girdi olarak kullanılmaktadır. Bu çalışmada ise sabit işlemlere ek olarak esnek işlemlerin de makineler üzerindeki işlem süreleri problem girdisini oluşturmaktadır. Bunların yanında, parçaların ve makinelerin özdeş olmama durumları her parçanın her makine üzerinde farklı sabit ve esnek işlem sürelerine sahip olması durumunu doğurmuştur. Bu duruma bağlı olarak f_i^k , i parçasının k makinesi üzerindeki sabit işlem süresi olarak; s_i^k , i parçasının k makinesi üzerindeki esnek işlem süresi olarak belirlenmiştir. Daha önce de belirtildiği gibi,

çalışmada yer alan problemler arasında makine özdeşliği, esnek işlem sayısı ve esnek makinelerin hangi makineler olduğu yönünde farklılıklar bulunmaktadır. O nedenle, esnek işlemler kümesi değişim göstermektedir ve s_i^k parametresi esnek işlem süresinin makineye göre değişmediği durumlarda s , iki esnek işlemin bulunduğu durumlarda s_i^{lk} olarak kullanılmaktadır.

Çalışmada ele alınan problemler aşağıdaki gibi listelenebilir.

1. **Problem 1:** 2-Makine Tek Esnek İşleme Sahip Sistemler

Bu problemde her parçanın üç işlemleri olduğu, bu işlemlerden ikisinin sırasıyla birinci ve ikinci makinelerde işlenmesi gerektiği diğer işlemin ise iki makineden herhangi birisinde işlenebildiği varsayılmıştır. Makine 1'deki sabit işlem süresi f_i^1 , makine 2'deki sabit işlem süresi f_i^2 , esnek işlem süresi ise s ile gösterilmiştir. İşlemler birbirinden farklı olsa bile, esnek işlemin tüm parçalar için aynı olduğu varsayılmıştır. Bu tarz durumlar ile gerçek hayatta farklı kesici uçların kullanıldığı CNC makineler yardımıyla ürün çeşitliliğinin fazla olduğu işletmelerde, PCB kart üzerinde devre elemanlarının yerleştirildiği sistemlerde ve çapraz eğitilmiş işçilerin yer aldığı montaj hatlarında karşılaşılabılır. Bu problemde ele alınan sistem Şekil 3.1'de görülebilir.

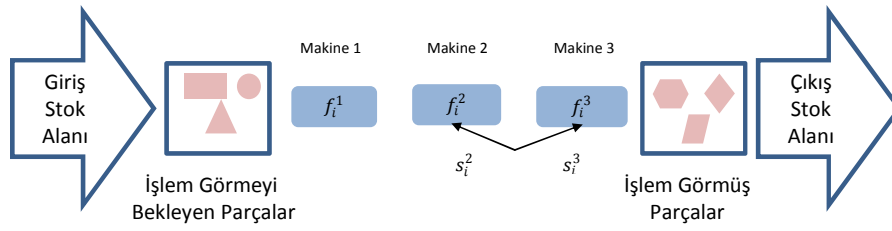


Şekil 3.1: Problem 1: 2 Özdeş Makine Tek Esnek İşlemlili Sistemler

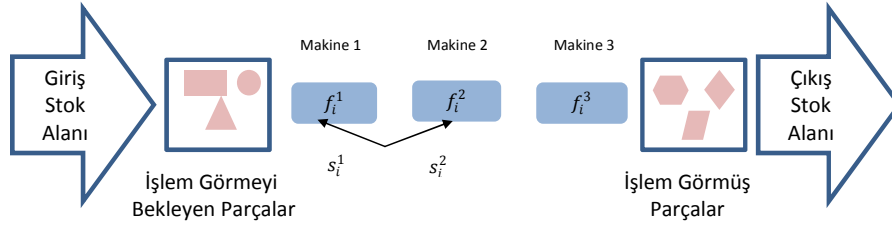
2. **Problem 2:** 3-Makine 2. ve 3. Makine Arasında Tek Esnek İşleme Sahip Sistemler

Bu problemde her parça dört işleme sahiptir. Bu işlemlerden üç tanesi sırasıyla makine 1, 2 ve 3'te işlenen sabit işlemlerdir. Bunların işlem süreleri sırasıyla f_i^1 , f_i^2 , ve f_i^3 olarak gösterilmiştir. Diğer işlemin ise makine 2 veya 3'te işlenebildiği varsayılmıştır. Bu problemde Problem 1'den farklı olarak makineler özdeş değildir. Ayrıca esnek işlem parçalar arasında değişim göstermektedir.

i parçasının esnek işleminin makine 2'ye atanması durumunda esnek işlem süresi s_i^2 , makine 3'e atanması durumunda s_i^3 olacaktır. Bu problemde ele alınan sistem Şekil 3.2'de gösterilmektedir. Diğer taraftan esnek işlemin 1. ve 2. makineler arasında olduğu durum Problem 2'nin *ters* problemidir. Pinedo'nun [48] ispatladığı ters dönebilirlik özelliğine göre Problem 2 için optimal çözümü veren çözüm yöntemi, ters problem için de optimal çözümü verecektir. Bu sebeple, bu çalışmada esnek işlemin makine 2 ve 3 arasında olduğu durum ele alınacaktır. Ters probleme dair gösterim Şekil 3.3'de yer almaktadır.



Şekil 3.2: Problem 2: 3 Özdeş Olmayan Makine, 2. ve 3. Makinelerde Tek Esnek İşlem



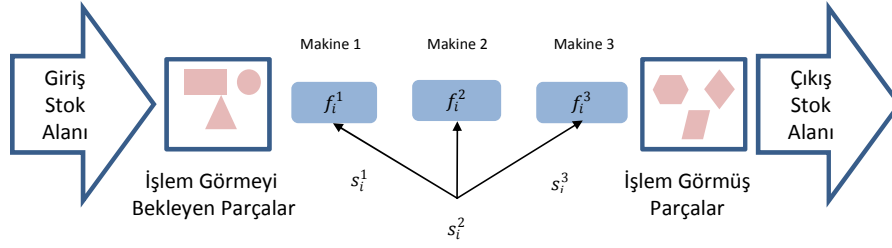
Şekil 3.3: Problem 2 Ters: 3 Özdeş Olmayan Makine, 1. ve 2. Makinelerde Tek Esnek İşlem

3. Problem 3: 3 Makinede de İşlenebilen Tek Esnek İşleme Sahip Sistemler

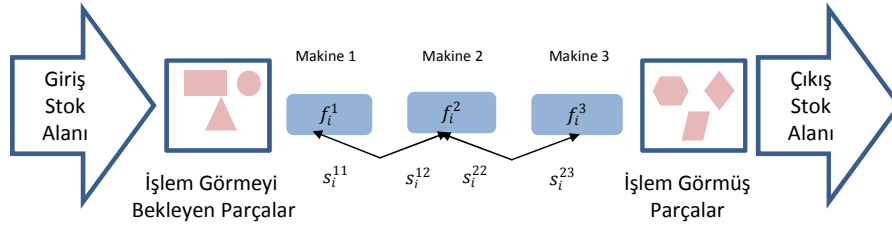
Bu problem de Problem 2 ile benzer olarak dört işleme sahiptir. Farklı olarak, esnek işlem tüm makinelerde işlenebilmektedir ve esnek işlem parçaya ve makineye göre değişmektedir. Bu probleme dair gösterim Şekil 3.4'te verilmiştir.

4. Problem 4: 3- Makine İki Esnek İşleme Sahip Sistemler

Son problemde ise 3-makinelikli akış tipi sistemde üç sabit işleme ek olarak 2 esnek işlem olduğu varsayılmıştır. İlk esnek işlem makine 1 ve 2'de işlenebilir. i parçasının ilk esnek işlemi makine 1'e atanırsa s_i^{11} , makine 2'e atanırsa s_i^{12} kadar işlem süresi gerekecektir. Aynı parçanın ikinci esnek işlemi ise makine 2 ve 3'te işlenebilecektir. Makine 2'deki süresi s_i^{22} , makine 3'deki süresi s_i^{23} . Ele alınan sisteme dair gösterim Şekil 3.5'de yer almaktadır.



Şekil 3.4: Problem 3: 3 Özdeş Olmayan Makine, 3 Makinede de İşlenebilen Tek Esnek İşlem



Şekil 3.5: Problem 4: 3 Özdeş Olmayan Makine, 2 Esnek İşlem

Bu çalışmada ele alınan problem için yayılma zamanını enküçükleyen karma tamsayılı doğrusal programlama modeli ve metin boyunca kullanılacak notasyon aşağıda yer almaktadır.

Kümeler

$M = \{1, \dots, m\}$; Makine kümesi

$E =$ Esnek makineler kümesi

$S =$ Esnek olmayan makineler kümesi

$N = \{1, \dots, n\}$; Parçalar kümesi

Parametreler

$f_i^k = i$ parçasının k makinesindeki sabit işlem süresi, $i \in N \quad k \in M$.

$s_i^k =$ parça i için k makinesindeki esnek işlem süresi, $i \in N \quad k \in E$.

Karar Değişkenleri

$$x_{ij} = \begin{cases} 1 & \text{parça } i, j. \text{ pozisyona atanmışsa} \\ 0 & \text{diğer durumlarda} \end{cases}$$

$$y_j^k = \begin{cases} 1 & j. \text{ pozisyondaki parçanın esnek işlemleri } k \text{ makinesine atanmışsa} \\ 0 & \text{diğer durumlarda} \end{cases}$$

$$b_{ij}^k = \begin{cases} 1 & \text{parça } i, j. \text{ pozisyona atanmışken esnek işlemleri } k \text{ makinesine atanmışsa} \\ 0 & \text{diğer durumlarda} \end{cases}$$

$$T_j^k = j. \text{ pozisyona atanan parçanın } k \text{ makinesindeki başlama zamanı}$$

$$C_{max} = n. \text{ pozisyona atanan parçanın sonuncu makinede işleminin tamamlanma zamanı} \\ (\text{yayılma zamanı})$$

Min C_{max}

Öyle ki

$$C_{max} \geq T_n^m + \sum_{i=1}^n f_i^m .x_{in} + s_i^m .b_{in}^m \quad \forall i \in N \quad (3.1)$$

$$T_j^k \geq T_j^{k-1} + \sum_{i=1}^n f_i^{k-1} .x_{ij} + s_i^{k-1} .b_{ij}^{k-1} \quad \forall j \in N, (k-1) \in E \quad (3.2)$$

$$T_j^k \geq T_j^{k-1} + \sum_{i=1}^n f_i^{k-1} .x_{ij} \quad \forall j \in N, (k-1) \in S \quad (3.3)$$

$$T_{j+1}^k \geq T_j^k + \sum_{i=1}^n f_i^k .x_{ij} \quad \forall j \in N - \{n\}, k \in S \quad (3.4)$$

$$T_{j+1}^k \geq T_j^k + \sum_{i=1}^n f_i^k .x_{ij} + s_i^k .b_{ij}^k \quad \forall j \in N - \{n\}, k \in E \quad (3.5)$$

$$\sum_{i=1}^n x_{ij} = 1 \quad \forall j \in N \quad (3.6)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i \in N \quad (3.7)$$

$$y_j^k \geq \sum_{i=1}^n b_{ij}^k \quad \forall i \in N, \forall j \in N, k \in E \quad (3.8)$$

$$b_{ij}^k + 1 \geq x_{ij} + y_j^k \quad \forall i \in N, j \in N, k \in M \quad (3.9)$$

$$\sum_k y_j^k = 1; \quad \forall j \in N, \forall k \in E \quad (3.10)$$

$$b_{ij}^k, y_j^k, x_{ij} \in \{0, 1\} \quad \forall i \in N, j \in N, k \in M \quad (3.11)$$

$$T_j^1 \geq 0, C_{max} \geq 0 \quad \forall i \in N, j \in N, k \in M \quad (3.12)$$

Bu formülasyonda, ilk kısıt amaç fonksiyonunu açıklamaktadır. Amaç fonksiyonu n . pozisyondaki parçanın son makinedeki tamamlanma zamanını (C_{max}) hesaplar. Kısıt 3.2, (j). pozisyondaki işin ($k-1$). makinedeki sabit işlemi ve eğer ($k-1$). makineye atandıysa esnek işlemi tamamlanmadan k . makinede işlenmeye başlamasını engeller. 3.3 numaralı kısıt ise j . pozisyondaki işin makine k 'da başlama zamanının, esnek işlem ($k-1$) makinesine atanmamışsa, bu parçanın ($k-1$)inci makinedeki sabit işleminin tamamlanma zamanından büyük eşit olması gerektiğini belirtir. 3.4 ve 3.5 numaralı kısıtlar makinelerin esnek olup olmamaları açısından atama ile ilgili

bir farklılık içermektedirler. $(j - 1)$ 'inci pozisyondaki işin k makinesindeki işlemini tamamlanmadan (j) 'inci pozisyondaki işin aynı makinede işleme başlamamasını sağlar. 3.6 ve 3.7 numaralı kısıtlar her işin bir pozisyonu olduğu ve her pozisyona sadece bir iş gelebildiğini göstermektedir. 3.8 ve 3.9 numaralı kısıtlar ise pozisyon ve esnek işlem ataması dolayısıyla iki karar değişkenini birbiriyle çarpılmasına neden olmasından kaynaklanan doğrusal olmayan durumu önlemek amacıyla oluşturulmuş kısıtlardır. b_{ij}^k 'nin x_{ij} ve y_j^k 'nin çarpımına eşit olmasını sağlar. 3.10 numaralı kısıt j . pozisyondaki işin esnek işleminin esnek makinelerden birine atanması gerektiğini söylemektedir. 3.11 numaralı kısıt, değişkenlere yönelik işaret kısıtlarını vermektedir. 3.12 numaralı kısıt ise 1. pozisyondaki işin ilk makinede 0 anından sonra başlamasını, yani çizelge başlangıç zamanını belirtir.

Ele alınan problemlerin karmaşıklığına yönelik Problem 1'in NP-Zor olup olmadığı bilinmemektedir. Problem 2, 3 ve 4 için Garey vd.'nin [22] çalışmasından yola çıkarak NP-Zor oldukları söylenebilmektedir. Garey vd. [22] 3-makine esnek işlemlerin olmadığı klasik akış tipi sistemde farklı tip parça çizelgelemenin NP-Zor olduğunu ispatlamışlardır. Problem 2, 3 ve 4 için akış tipi sisteme ek olarak esnek işlem atamasının da belirleniyor olması, problemin karmaşıklığını arttırmaktadır.

Bu bölümde, incelenen problemlerin genel özellikleri, ortak yönleri ve farklılaştıkları noktalara yönelik bilgi verilmiştir. Bununla birlikte problemleri optimal sonuca ulaştıran karma tamsayılı matematiksel model genel ifadelerle gösterilmiştir. İlerleyen bölümlerde çalışmaya konu olan problemler ayrı ayrı incelenerek kendilerine ait özellikleri, çözüm yöntemleri ve deneysel çalışma sonuçları her biri için detaylı bir şekilde sunulacaktır.

4. İKİ MAKİNE TEK ESNEK İŞLEME SAHİP SİSTEMLER

Bu bölümde iki makineli ve tek esnek işlemlili akış atölyeleri ele alınmıştır. Her bir parça üç ayrı işlemde meydana gelmektedir. İlk işlem 1. makinede yapılır, üçüncü işlem 2. makinede yapılır. İkinci işlem ise 1. ya da 2. makine tarafından yapılabilir. Problem, bir parça için esnek işlemin hangi makineye atanacağına kararlaştırılmasıdır. Bu atamalar her bir parçadan diğerine farklılık gösterebilir. Amaç 2. makinede işlem görecektir son parçanın tamamlanma zamanını minimize etmektir. Bu problem (Problem 1) için Bölüm 3'te verilen genel matematiksel model kullanılabilir. Bu problemde her parçanın esnek işlemin süresi aynı olduğu için matematiksel modelde verilen parçaların makineler üzerindeki işlem süresini gösteren notasyon (s_i^k) yerine (s) yeterli olmaktadır.

Problemin NP-Zor olup olmadığı bilinmemektedir. Johnson [39] algoritması, esnek işlem olmadan 2-makine akış tipi sistemlerde farklı tip parça çizelgelemesi için polinom zamanda optimal çözümü vermektedir. Fakat Problem 1 için esnek işlem atamasına karar vermek önem arz etmektedir. Makineler özdeş olduğu için iki makinede de işlenebilen esnek işlemin makineler üzerindeki işlem süreleri değişim göstermemektedir. Sistemde farklı tip parçalar üretildiği için sabit işlemlerin süresi makinelere göre değişmektedir. Bu çalışmada ara stok kapasitesinin sınırsız olduğu durum göz önüne alınmıştır.

4.1 Çözüm Yöntemi

Parçaların birinci ve ikinci makinedeki sıralamalarının aynı olduğu çizelgeleme permütasyon çizelge olarak adlandırılır [48]. Permütasyon çizelgelerinin 2-makine akış tipi sistemler için optimal olduklarının kanıtı Önteorem 1'de verilmiştir.

Önteorem 1 2-makine tek esnek işlemlerli sistemlerde optimal olan bir permütasyon çizelgesi vardır.

İspat: Permütasyon çizelge olmayan optimal bir çizelge olduğunu varsayalım. Bu durumda öyle j ve k işleri vardır ki makine 1’de j işi k işinden önce işlenir. Olabilecek değişik j ve k işi kombinasyonlarından, makine 2’de ardışık olan ve belirtilen duruma uyan j ve k işlerini seçelim. (Eğer permütasyon çizelgesi değilse belirtilen duruma uyan makine 2’de ardışık olan en az bir tane j ve k çifti vardır.)

Bu işler için esnek işlemlerin makinelere atanması yapıldıktan sonra makine 1 ve makine 2’deki toplam işlem sürelerini p_j^1, p_k^1 ve p_j^2, p_k^2 ile gösterelim. Buna göre, makine 2’de k işinin tamamlanma zamanı, C_k^2 , bu işin makine 2’de başlama zamanı üzerine bu makinedeki işlem zamanının eklenmesiyle bulunacaktır. T_k^2 bu işin makine 2’deki başlama zamanı ise, $C_k^2 = T_k^2 + p_k^2$ ’dir. j işinin makine 2’deki başlama zamanı için de $C_j^2 = T_j^2 + p_j^2$ yazılabilir. Diğer taraftan $C_j^1 \leq C_k^1$ olduğu için, makine 2’de iki iş arasında boş zaman oluşmayacaktır. Bu sebeple, $T_j^2 = C_k^2$ geçerli olacaktır. Sonuç olarak $C_j^2 = T_k^2 + p_k^2 + p_j^2$ ’dir.

Şimdi permütasyon çizelgeye uyacak şekilde makine 1’de herhangi bir değişiklik yapmadan bu iki işin makine 2’deki sıralamasını değiştirelim. Makine 2’deki yeni tamamlanma zamanları için $\hat{C}_j^2 = \hat{T}_j^2 + p_j^2$ ve $\hat{C}_k^2 = \hat{T}_k^2 + p_k^2$ geçerli olacaktır. $C_j^1 < C_j^2$ olduğu için $\hat{T}_j^2 \leq \hat{T}_k^2$ ’dir. $\hat{T}_k^2 \leq \max\{C_k^1, \hat{T}_j^2 + p_j^2\}$ ’dir.

Bu durumda $\hat{C}_k^2 \leq \max\{C_k^1, \hat{T}_j^2 + p_j^2 + p_k^2\}$ ’dir.

$C_k^1 = T_k^2$ ve $\hat{T}_j^2 \leq T_k^2$ olduğundan $\hat{C}_k^2 \leq T_k^2 + p_j^2 + p_k^2$ ’dir.

Sonuç olarak, bu iki parçadan makine 2’de sonra tamamlananları her iki çizelge için kıyasladığımızda, $\hat{C}_k^2 \leq T_k^2 + p_j^2 + p_k^2 = C_j^2$ ’dir. Yani permütasyon çizelgeye uymayan iki iş değiştirildiğinde diğer parçaların tamamlanma zamanları artmadan bu iki parça daha erken tamamlanabilmektedir. Bu da eğer eski çizelge optimalse yeni çizelgenin de optimal olduğu anlamına gelmektedir. Benzer ikili değişiklikler, permütasyon çizelgeye uymayan bütün iş ikilileri için yapılarak sonuçta optimal olan bir permütasyon çizelge elde edilebilir. \square

Önteorem 2 k^* esnek işlemli 2. makineye atanan iş sayısı olsun. İlk k^* parçanın esnek

işlemi 2. makineye atanmış olduğu ve son $(n - k^)$ parçanın esnek işlemlerinin ilk makineye atandığı bir optimal çizelge vardır.*

İspat: (Çelişki ile) $k^* = 0$ ve $k^* = n$ durumları önemsizdir. $1 \leq k^* \leq (n - 1)$ olduğu ve ilk k^* parçanın esnek işleminin 2. makineye atanmış olduğu varsayalım. Böyle bir durumda en az bir defa esnek işlem parça i için ilk makineye, onun öncülü parça j için 2. makineye atanmıştır. Aynı zamanda, parça h 'nin parça i 'ye öncül olduğu varsayalım. Bu durumda:

$$C_j^1 = T_i^1 + f_i^1 + s + f_j^1.$$

$$C_j^2 = \max\{\max\{C_h^2, T_i^1 + f_i^1 + s\} + f_i^2, T_i^1 + f_i^1 + s + f_j^1\} + f_j^2 + s.$$

Şimdi, parça i ve j için esnek işlem atamalarını değiştirilsin. Bu durumda parça i 'nin esnek işlemi 2. makineye, j 'nin esnek işlemi ilk makineye atanmaktadır. Bu değişim parça i 'ye kadar olan çizelgeye ait tamamlanma zamanlarını etkilemez. Dolayısıyla, parça j 'ye ait yeni tamamlanma zamanı makinelere göre şu şekilde hesaplanabilir:

$$C_j^1(\text{yeni}) = T_i^1 + f_i^1 + f_j^1 + s = C_j^1.$$

$$C_j^2(\text{yeni}) = \max\{\max\{C_h^2, T_i^1 + f_i^1\} + f_i^2 + s, T_i^1 + f_i^1 + f_j^1 + s\} + f_j^2.$$

Bir kaç düzenleme sonrası bu denklemler elde edilmektedir:

$$\begin{aligned} C_j^2(\text{yeni}) &= \max\{\max\{C_h^2, T_i^1 + f_i^1\} + f_i^2, T_i^1 + f_i^1 + f_j^1\} + f_j^2 + s \\ &\leq \max\{\max\{C_h^2, T_i^1 + f_i^1 + s\} + f_i^2, T_i^1 + f_i^1 + f_j^1 + s\} + f_j^2 + s = C_j^2. \end{aligned}$$

Bu da, i ve j için yapılan esnek işlem atamasına dair değişimler sonrası tamamlanma zamanının, parça j için ilk makinede değişmediğini ikinci makine için de azaladığını veya aynı kaldığını göstermektedir. Sonuç olarak, ilerleyen parçalar için de tamamlanma zamanının şimdikinden daha geç olmayacağını göstererek öntoremi kanıtlamaktadır. \square

Bu lemmmanın kanıtından sonra, problemin çözümü $(n + 1)$ elemanlı bir vektörle ifade edilebilir: $(k^* : j_1, j_2, \dots, j_n)$. Bu notasyonda, k^* optimal çözümde esnek işlemi

2. makineye atanan parça sayısını ve kalan n eleman da optimal parça sıralamasını vermektedir.

f_j^1, f_j^2 ve s şeklinde verilen 2-makineli akış tipi probleme ek olarak, \hat{f}_j^1, \hat{f}_j^2 ve \hat{s} şeklinde ikinci bir 2-makine problemi tanıtalım. Öyle ki;

$$f_j^1 = \hat{f}_j^2, f_j^2 = \hat{f}_j^1 \text{ ve } s = \hat{s} \text{ olsun.}$$

Bu problem ters problem olarak adlandırılmıştır. Aşağıdaki önteorem orjinal ve ters problemler arasındaki bağlantıyı sağlamaktadır.

Önteorem 3 ($k^* : j_1, j_2, \dots, j_n$) ilk problem için optimal çözüm ise $((n - k^*) : j_n, j_{n-1}, \dots, j_1)$ ters problem için optimal çözümdür.

İspat: Bu ifade, Pinedo [48] tarafından 2-makine klasik akış tipi sistemler için yapılan ispat ile benzerlik göstermektedir.

Orjinal problem için herhangi bir olurlu çizelgenin aynı yayılma zamanına sahip ters problem için olurlu bir çizelgeye dönüştürülebildiği gösterilebilir. Fakat detaya inmeden, ters çizelge \hat{T} 'nin Gantt şemasını orjinal çizelge T 'nin Gantt şemasının aynadaki yansıması olarak gözlemlemek ispat için yeterlidir. Bu çizelge zaman eksenin ters çevrilmesi ve iki makinenin rol değişimiyle elde edilmektedir.

Klasik 2-makineli sistemler için optimal çizelgeler Johnson kuralı (SPT1-LPT2) ile bakılabilir. Bu kurala göre, makine 1'deki işlem süresi makine 2'den kısa olan işlemler, makine 1'deki işlem sürelerine göre Shortest Processing Time (SPT), diğerleri ise makine 2'deki işlem sürelerine göre Longest Processing Time (LPT) şeklinde sıralanır ve iki sıra birleştirilir. Aşağıdaki önteorem mevcut problem için bu kuralın nasıl kullanılabileceğini göstermektedir. \square

Önteorem 4 Johnson (SPT1-LPT2) kuralına uyan bir optimal çizelge mevcuttur.

Bu önteorem klasik 2-makineli akış tipi problem için (SPT1-LPT2) kuralının optimallik ispatının benzeri şekilde ispatlanabilmektedir ([48]). Aslında, esnek işlemlerin ataması gerçekleştiği zaman, problem klasik 2-makineli akış tipi probleme

indirgenbilmektedir. Bu önteoreme göre Johnson kuralına uyan optimal bir çizelge vardır. Ancak, Johnson kuralına uymayan diğer birçok alternatif optimal çözümler de olabilmektedir.

Önteorem 5 *Problemin parametrelerinden bağımsız olarak ilk parçanın esnek işleminin ikinci makineye atandığı ve son parçanın esnek işleminin birinci makineye atandığı optimal bir çizelge vardır.*

İspat: Öncelikle çizelgedeki ilk işin esnek işleminin makine 2'ye atanacağını ispatlayalım. Bu işi j_1 ile gösterelim. Sıralamadaki ikinci parça j_2 için atamaya bağlı iki durum göz önünde bulundurulmalıdır.

1. Durum 1. Eğer j_2 'nin esnek işlemi ilk makineye atanırsa:

$$C_{j_2}^1 = f_{j_1}^1 + s + f_{j_1}^2 + s \quad \text{ve} \quad C_{j_2}^2 = \max\{f_{j_1}^1 + s + f_{j_1}^2, f_{j_1}^1 + s + f_{j_2}^1 + s\} + f_{j_2}^2$$

j_1 işinin ataması makine 1'den 2'ye değiştirildiğinde, j_2 işinin makineler üzerindeki yeni tamamlanma zamanlarını makine 1 ve makine 2'de $\hat{C}_{j_2}^1$ ve $\hat{C}_{j_2}^2$ olarak gösterelim:

$$\hat{C}_{j_2}^1 = f_{j_1}^1 + f_{j_2}^1 + s \quad \text{ve} \quad \hat{C}_{j_2}^2 = \max\{f_{j_1}^1 + f_{j_1}^2 + s, f_{j_1}^1 + f_{j_2}^1 + s\} + f_{j_2}^2 + s.$$

$\hat{C}_{j_2}^1 \leq C_{j_2}^1$ ve $\hat{C}_{j_2}^2 \leq C_{j_2}^2$ 'dir. Takip eden işlerde de herhangi bir artma olmayacağı için yeni çizelge optimal olacaktır.

2. Durum 2. Eğer j_2 esnek işlemi ikinci makineye atanırsa:

$$C_{j_2}^1 = f_{j_1}^1 + s + f_{j_2}^1 \quad \text{ve} \quad C_{j_2}^2 = \max\{f_{j_1}^1 + s + f_{j_1}^2, f_{j_1}^1 + s + f_{j_2}^1\} + f_{j_2}^2 + s.$$

j_1 işinin ataması makine 1'den 2'ye değiştirilsin:

$$\hat{C}_{j_2}^1 = f_{j_1}^1 + f_{j_2}^1 \quad \text{ve} \quad \hat{C}_{j_2}^2 = \max\{f_{j_1}^1 + f_{j_1}^2 + s, f_{j_1}^1 + f_{j_2}^1\} + f_{j_2}^2 + s.$$

Tekrardan, bütün ardıl işlerin tamamlanma zamanları $\hat{C}_{j_2}^1 \leq C_{j_2}^1$ ve $\hat{C}_{j_2}^2 \leq C_{j_2}^2$, şimdiki tamamlanma zamanlarından büyük olmayacaktır. Sonuç olarak, bu yeni çizelge de optimaldir.

Kanıtın ikinci kısmı, ilk kısmın ve ters dönebilirlik (Önteorem 3) özelliğinin direk sonucudur. \square

Bu ispatlardan sonra, parçalar öncelikle Johnson kuralına göre sıralanarak, ona bağlı esnek işlem ataması yapmak optimal çözüme ulaştırır denilebilir. Fakat Örnek 1 bu ifadenin yanlış olduğunu göstermektedir.

Örnek 1 $s = 8$ ve $(f_j^1, f_j^2) = (5, 10), (6, 8), (15, 13), (8, 12), (16, 15)$ $j = 1, 2, \dots, 5$ olarak verilmiş olsun. Bu örnekte parçalar esneklik göz ardı edilerek Johnson kuralına göre sıralandığında parça sıralaması $(1, 2, 4, 5, 3)$ şeklinde olur. Bu sıralama kullanıldığında ise, mümkün olan en iyi esnek işlem atamasının ilk iki iş için 2. makineye olduğu ve kalan işler için de ilk makineye olduğu sonucuna varılmıştır. Bu atama ve sıralama sonucunda yayılma zamanı 87 olmuştur. Bu çözüme dair Gantt Şeması Şekil 4.1(a)'da yer almaktadır. Diğer yandan, optimal sonuç $(2 : 1, 4, 5, 3, 2)$ şeklindedir ve yayılma zamanı 82'dir. Optimal çözüme ait Gantt Şeması Şekil 4.1(b)'dedir.

Problem için basit çözüm yolları üretilebilmekte fakat optimal çözüme ulaşmak kolay olmamaktadır. Problemin karmaşıklığı henüz bilinmemektedir fakat çözüm yöntemi olarak bir çok durumda optimal değer ile aynı sonuçları veren polinom zamanlı bir sezgisel geliştirilmiştir.

Algoritma 1 Problem 1 için geliştirilen sezgisel algoritmanın adımlarını vermektedir. Bu algorithmada amaç 2-makine akış tipi problemini polinom zamanda optimal çözen Johnson algoritmasından yararlanmaktır. Algoritmanın ilk adımında parçalar esnek işlemi birinci ve ikinci makineye atanacak parçalar kümesi olarak ikiye ayrılır. Öncelikle tüm parçaların esnek işlemi 2. makineye atanır. Bu atama sonucu elde edilen işlem süreleri üzerinden Johnson algoritması uygulanarak, yayılma zamanı değeri elde edilir. İlk başta oluşturulan kümelerin ikisinden de birer parça seçilerek parçalara ait esnek işlem atamaları değiştirilir. Değişim sonrası yayılma zamanı

Algoritma 1 Problem 1: Genel Çözüm Algoritması 1

Girdi: $J = 1, 2, \dots, n, f_j^1, f_j^2, j \in J, s$

Çıktı: $(k^* : j_1, j_2, \dots, j_n), p_j^1, p_j^2, j \in J$

$L = j : f_j^1 \leq f_j^2$ olarak belirle

$R = j : f_j^1 > f_j^2$ olarak belirle

L kümesindeki işleri f_j^1 değerlerine göre SPT kuralı ile (küçükten büyüğe) sırala

R kümesindeki işleri f_j^2 değerlerine göre LPT kuralı ile (büyükten küçüğe) sırala

Önce L sonra R 'deki işler gelecek şekilde sıraları birleştir

$r = 0; r \leq n; r++$

Rastgele r tane işin esnek işlemini ilk makineye, geriye kalan $n - r$ işin esnek işlemini ikinci makineye ata

İLK=esnek işlemleri ilk makineye atanan işler kümesi, İKİNCİ= $J - \{\text{İLK}\}$

Johnson kuralına göre işleri sırala ve C_{max} değerini hesapla

Tekrarla

İLK ve İKİNCİ kümelerinden birer eleman seç

Seçilen elemanların esnek işlem atamalarını değiştir, C_{max} değerini yeniden hesapla

Son tekrarlar: İLK ve İKİNCİ kümelerinin tüm elemanları için tekrarlar

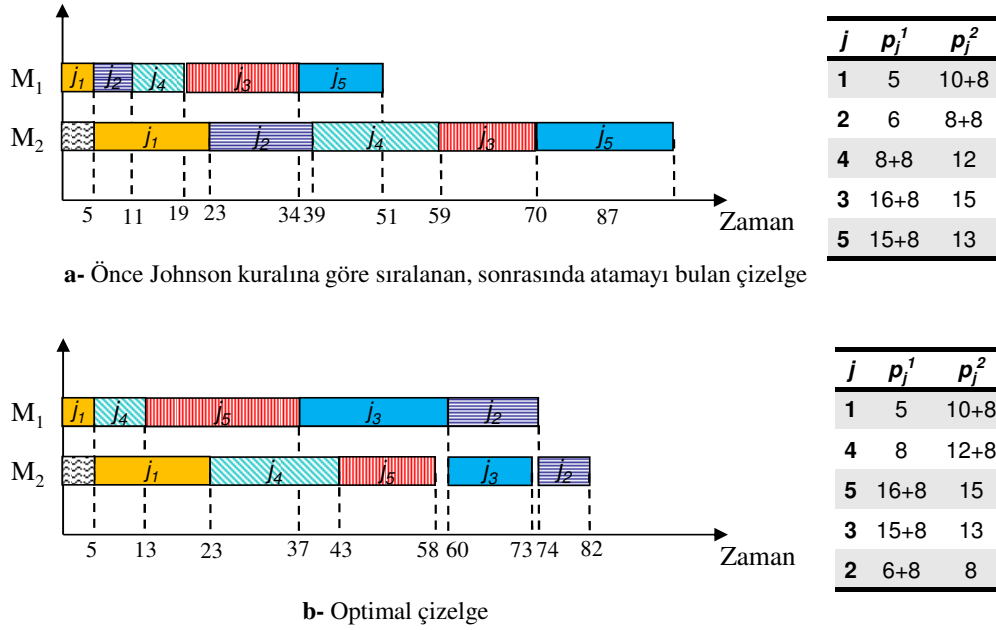
C_{max} değeri en düşük olan alternatifi seç.

İLK ve İKİNCİ kümelerini güncelle

İyileştirme olmayana kadar devam et

bitiş döngü{r}

En düşük yayılma zamanına sahip alternatifi seç



Şekil 4.1: Problem 1: Johnson Karşılaştırma Gantt Şeması

değeri hesaplanarak, alternatifler arasında en küçük C_{max} değerine sahip olan seçilir. İyileştirme gerçekleşmeyinceye kadar değişime devam edilir. Bu çözüm yöntemini test etmek amacıyla yapılmış deneysel çalışma ilerleyen alt bölümde yer almaktadır.

4.2 Deneysel Çalışma

Değişik tip parça üreten 2-makineli akış tipi sistemde esnek işlemin bulunduğu problem için bir önceki bölümde geliştirilen sezgiseli test etmek için bir deneysel çalışma düzenlenmiştir. Üretilen problemler için optimal çözümler matematiksel model ve GAMS CPLEX 12.0 çözücüsü ile elde edilmiştir. Geliştirilen sezgiseller Visual C++ Express 2010 ile kodlanmıştır. Hem matematiksel model ve hem de sezgisellerin çözümünde Intel Xeon E5645 2.4 GHz 12 çekirdekli, 12GB bellek ve 4 paralel işlemcili bir makine kullanılmıştır. Matematiksel modeller için 900 saniye zaman limiti konulmuştur. Bu süre içerisinde optimal çözüme ulaşılamaması durumunda eldeki en iyi çözüm ve bu çözümün optimale olan uzaklığı alınmıştır.

Bu bölümde Problem 1 için geliştirilen sezgiselin performans analizi için

gerçekleştirilen deneysel çalışmadan ve elde edilen sonuçlardan bahsedilmektedir. Deneysel çalışmada 20 ve 50 parçalık problemler türetilmiştir. İşlem sürelerinin nicelik olarak büyük ya da küçük olmalarının yanında, parçalar arası işlem süreleri varyansının düşük ya da yüksek olmasının problemin zorluğuna ve geliştirilen çözüm yöntemlerinin performansına etkilerini incelemek için, bütün işlem süreleri farklı aralıklardan türetilmiştir. Oluşturulan deneysel çalışmada, f_i^1, f_i^2 ve s değerleri için alt sınır ve işlem sürelerinin alabileceği değer aralığı anlamında ikişer seviye yer almaktadır. Alt sınırlar 10 ve 100 olarak belirlenmiştir. Aralıklar ise alt sınırlar üzerinden belirlenerek yüksek seviyede (4-taban değer), düşük seviyede (taban değer/2) şeklindedir. Dolayısıyla işlem süreleri, f_i^1, f_i^2 ve s değerlerinin her biri için $([10, 15], [10, 50])$ veya $([100, 150], [100, 500])$ aralıklarında tek düze dağılım gösteren değerlerden rastgele olarak elde edilmiştir. Aralıkların ve taban değerlerin iki seviyeli olması, sabit ve esnek işlemlerin taban değerlerinin ve aralık seviyelerinin ayrı birer faktör olarak alınması, aynı zamanda 20 ve 50 parça üreten problemlerin ayrı ayrı ele alınması nedeniyle ($2^6 = 64$) farklı durum oluşmaktadır. Her durum için 5 tekrar yapılmıştır. Sonuç olarak 320 farklı problem seti üretilmiştir.

Sezgisel değer ile CPLEX ile elde edilen değerler arasındaki fark üzerinde ve CPLEX çözüm süresi üzerinde parça sayısının, işlem sürelerinin taban değerlerinin ve aralık seviyeleri gibi karakteristik özelliklerinin etkisi olup olmadığını anlayabilmek ve etkilerini gösterebilmek için bu değişkenlere farklı parametrik değerler verilmiştir. Üretilen problemler için optimal çözümler, karma tamsayı programlama ile modellenmiş ve GAMS CPLEX 12.0 çözücüsü ile çözdürülmüştür. Her problemin optimal çözüme ulaşabilmesi için 900 saniyelik bir zaman limiti konulmuştur ve bazı problemlerde bu süre içerisinde optimal çözüme erişilememektedir. Çizelgeleme problemleri, üretim öncesinde günlük ve sayıca fazla sayıda çözülmesi gereken problemler olduğu için 900 saniyelik bir zaman kısıtı ile çözümleri anlamlıdır. İşlem sürelerinin taban değerlerine göre değerlendirilen GAMS CPLEX 12.0 sonuçlarına yönelik özet Tablo 4.1'de yer almaktadır. Tablo 4.1'de problemler işlem süreleri aralık seviyelerine göre ayrılmadan aynı işlem süresi taban değeri seviyesine sahip olmalarına ve üretilen parça sayısına göre göre gruplanmışlardır.

20 parçalı problemler için verilen süre limiti problemlerin çözümü için yeterli olduğu Tablo 4.1 ile gösterilmektedir. Türetilen 160 örnekten hiç biri zaman limitine takılmamıştır. Fakat parça sayısı 50'ye çıktığında ise 160 problemde 99'u zaman

Tablo 4.1: Problem 1: CPLEX Çözüm Durumları

	Optimal Çözüm			Zaman Limiti		
	Düşük (10)	Yüksek (100)	Toplam	Düşük (10)	Yüksek (100)	Toplam
20 İş (160)	80	80	160	0	0	0
50 İş (160)	30	31	61	50	49	99

limitine takılmıştır. Parça sayısındaki artış problemin çözüm süresini arttırdığı için, geliştirilen sezgisel çözüm yöntemi önem kazanmaktadır. Önerilen bu sezgisele ait performans analizi CPLEX ile bulunmuş yayılma zamanı alt sınır değerleri, önerilen sezgisellerin yayılma zamanı değerleri arasındaki yüzde fark kullanılarak yapılmıştır. İki farklı yüzde fark hesabı yapılmaktadır.

CPLEX Alt sınır Yüzde Fark= (Sezgisel değeri - CPLEX Alt Sınır Değeri)/(CPLEX Alt Sınır değeri·100)

En iyi CPLEX Yüzde Fark= (Sezgisel değeri - En İyi CPLEX değeri)/(En İyi CPLEX Değeri·100)

Tablo 4.2 Problem 1 için geliştirilmiş olan sezgisel çözüm yöntemi ile elde edilen değerler ile CPLEX üzerinden elde edilen alt sınır değerleri arasındaki yüzde farkları iş sayısına ve işlem sürelerinin taban değerlerine göre göstermektedir.

Tablo 4.2: Problem 1: İş Sayısına Göre Sezgisel - Alt Sınır Yüzde Fark

	Sezgisel - CPLEX Alt Sınır Fark (%)	İşlem Süresi Taban Değeri Seviyesi	
		Düşük (10)	Yüksek (100)
20 iş	Ortalama	0	0
	Maksimum	0	0
50 iş	Ortalama	0.24	0.22
	Maksimum	1.59	1.42

Tablo 4.3'de ise CPLEX'in bulduğu en iyi çözüme olan uzaklıklar farklı işlem süresi seviyelerine göre gruplandırılarak verilmiştir.

CPLEX çözümlerine göre hata oranları Tablo 4.2 ve 4.3'de verilmiştir. CPLEX'in alt

Tablo 4.3: Problem 1: İş Sayısına Göre Sezgisel En iyi CPLEX % Fark

	Sezgisel -En iyi CPLEX % Fark	İşlem Süresi Taban Değeri Seviyesi	
		Düşük(10)	Yüksek (100)
20 iş	Ortalama	0	0
	Maksimum	0	0
50 iş	Ortalama	0	0
	Maksimum	0	0

sınır maksimum uzaklık %1.59'dur.

Tablo 4.4: Problem 1: 20 İş Sezgisel Ve CPLEX Ortalama Çözüm Süresi

Çözüm Süresi (saniye)	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (10)	Yüksek (100)
Sezgisel	6.54	19.41
CPLEX	7.11	4.75

Tablo 4.5: Problem 1: 50 İş Sezgisel Ve CPLEX Ortalama Çözüm Süresi

Çözüm Süresi (saniye)	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (10)	Yüksek (100)
Sezgisel	6.95	21.01
CPLEX	625.17	612.69

Tablo 4.4'e göre CPLEX çözüm süresi 20 parçalı problemlerde, işlem süresi taban değeri 100 olan problemlerde ortalama olarak sezgisele göre daha kısa sürede çözüm vermektedir. Fakat parça sayısı 50'ye yükseldiğinde CPLEX ortalama çözüm süresi de hızla düşük seviye için 7.11 saniyeden 625.17 saniyeye; yüksek seviye için 4.75 saniyeden 612.69 saniyeye yükselmektedir. Geliştirilen sezgisel için ise bu rakam sadece ortalama 1.5 saniyelik bir artış göstermektedir (Tablo 4.5). Bununla birlikte CPLEX 900 saniyelik bir zaman limiti ile çalışmasına rağmen sonuçlar bu denli yüksek olmuştur. Problemler için CPLEX çözümünde zaman limiti kaldırıldığı takdirde, zaman limitine takılan problemler optimal çözüme 900 saniyeden daha uzun bir sürede ulaşacakları için Tablo 4.5'de yer alan CPLEX çözüm süresi de artacaktır.

Üretilecek parça sayısının, işlem sürelerini taban değerlere göre büyüklüğünün ve işlem sürelerinin alındığı aralığın çözüm süresi ve yüzde farklar üzerindeki etkisinin

istatistiksel olarak anlamlı olup olmadığını test etmek adına 2^4 faktöriyel tasarım oluşturulmuştur. Tasarımda, sabit ve esnek işlem sürelerinin aralıklarının etkisinin görülebilmesi için esnek işlem sürelerinin hepsi ya düşük ya da yüksek aralık seviyesine sahiptir. Tablo 4.6’da yer alan tasarımda işlem süreleri için belirlenen aralık seviyeleri tüm işlemler için sabitlenerek 2^4 faktöriyel tasarıma dönüştürülmüştür ve tabloda bulunan her kombinasyon için oluşturulan 5 koşturumun 5’i de analiz için kullanılmıştır.

Analiz Minitab 16.0 paket program ile gerçekleştirilmiştir. Bu programa ait çıktıların değerlendirilmesi sırasında tablolarda yer alan p-değerleri kullanılmaktadır. Yapılan testler sırasında kullanılan hipotezler şu şekildedir.

H_0 = Faktör istatistiksel olarak %95 güven düzeyinde anlamlı değildir.

$H_{alternatif}$ = Faktör istatistiksel olarak %95 güven düzeyinde anlamlıdır.

Tablo 4.6: Problem 1: Faktör Tasarımı

Parça Sayısı	İşlem Süresi	Taban Değeri	Aralık f^1	Aralık f^2	Aralık s
0	0	0	0	0	0
0	0	0	0	0	1
0	0	0	1	1	0
0	0	0	1	1	1
0	1	1	0	0	0
0	1	1	0	0	1
0	1	1	1	1	0
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	0	0	1
1	0	0	1	1	0
1	0	0	1	1	1
1	1	1	0	0	0
1	1	1	0	0	1
1	1	1	1	1	0
1	1	1	1	1	1

Tablolarda faktörlere karşılık gelen p-değerleri 0.05’ten küçükse boş (null) hipotez reddedilmekte ve alternatif hipotez doğrulanmaktadır. Aksi takdirde boş hipotez reddedilemeyerek, faktörün etkisiz olduğu anlaşılmaktadır. Analizin geri kalan kısmı bu şekilde değerlendirilmiştir.

CPLEX çözüm süresine ait istatistiksel değerlerin Minitab çıktısı Tablo 4.7 verilmiştir. Bu tabloya göre p-değeri 0.05'ten küçük olan bir faktör veya faktör etkileşimi gözlenmemektedir. Bu da faktörlerden hiçbirinin istatistiksel anlamda çözüm süresi üzerinde etkili olmadığını göstermektedir. Parça sayısı 20'den 50 işe çıktığında CPLEX çözüm süresi ortalamalara bakıldığında çok büyük artış göstermektedir. Fakat deneysel çalışmadaki diğer faktörlerle birlikte gözlemlendiğinde parça sayısı çözüm süresi üzerinde istatistiksel olarak anlamlı bir faktör olarak görülmemektedir. Çözüm süresini etkilediği düşünülen bu faktörün istatistiksel olarak anlamlı olmamasının nedeni uç değerler olabilir. Uç değerlerin etkisi daha büyük örneklemeler kullanılarak engellenebilir.

Tablo 4.7: Problem 1: CPLEX Çözüm Süresi Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		115468	210230	0,55	0,585
Parça Sayısı	98708	49354	91395	0,54	0,591
Taban	140877	70439	173577	0,41	0,686
Aralık f	-198868	-99434	210230	-0,47	0,638
Aralık s	580280	290140	210230	1,38	0,172
Parça Sayısı-Taban	60167	30083	75461	0,40	0,691
Parça Sayısı-Aralık f	-85347	-42674	91395	-0,47	0,642
Parça Sayısı-Aralık s	248781	124390	91395	1,36	0,178
Taban-Aralık f	-165383	-82691	173577	-0,48	0,635
Taban-Aralık s	477927	238963	173577	1,38	0,173
Aralık f-Aralık s	157208	78604	210230	0,37	0,710
Parça Sayısı-Taban-Aralık f	-70972	-35486	75461	-0,47	0,640
Parça Sayısı-Taban-Aralık s	204903	102451	75461	1,36	0,179
Parça Sayısı-Aralık f-Aralık s	66975	33487	91395	0,37	0,715
Taban-Aralık f-Aralık s	131596	65798	173577	0,38	0,706

Çalışmada ele alınan faktörlerin sezgiselin alt sınıra olan uzaklığı üzerindeki etkisi incelendiğinde istatistiksel olarak anlamlı bir faktör bulunamamıştır. Tablo 4.8 analize ait istatistiksel verileri barındırmaktadır. Faktörlerin p-değerlerinin 0.05'ten büyük olduğu gözlemlenmiştir. Bu durumun nedeni geliştirilen sezgiselin çok

kaliteli sonuçlar vermesidir. Belirlenen parametrik değerlerin değişimi sezgiselin performansını etkileyememektedir. Problem 1 için CPLEX En iyi Çözümü ile sezgisel arasındaki yüzde farkı etkileyen faktörler incelenmemiştir. Çünkü deneysel çalışmada yer alan bütün problemlerin hepsi için bu yüzde 0'a eşittir ve faktöre bağlı bir değişim beklenmesi anlamsızdır.

Tablo 4.8: Problem 1: CPLEX Alt Sınır Sezgisel Yüzde Fark Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		161,1	138,67	1,16	0,250
Parça Sayısı	138,1	69,0	60,28	1,15	0,256
Taban	249,0	124,5	114,49	1,09	0,281
Aralık f	-395,9	-198,0	138,67	-1,43	0,158
Aralık s	174,1	87,0	138,67	0,63	0,532
Parça Sayısı-Taban	106,7	53,4	49,77	1,07	0,288
Parça Sayısı-Aralık f	-169,7	-84,8	60,28	-1,41	0,164
Parça Sayısı-Aralık s	74,6	37,3	60,28	0,62	0,538
Taban-Aralık f	-322,3	-161,2	114,49	-1,41	0,164
Taban-Aralık s	140,3	70,2	114,49	0,61	0,542
Aralık f-Aralık s	187,4	93,7	138,67	0,68	0,502
Parça Sayısı-Taban-Aralık f	-138,1	-69,1	49,77	-1,39	0,170
Parça Sayısı-Taban-Aralık s	60,1	30,1	49,77	0,60	0,548
Parça Sayısı-Aralık f-Aralık s	80,3	40,2	60,28	0,67	0,508
Taban-Aralık f-Aralık s	158,2	79,1	114,49	0,69	0,492
Parça Sayısı-Taban-Aralık f-Aralık s	67,8	33,9	49,77	0,68	0,498

4.3 Sonuçlar

Bu bölümde 2-makineli akış tipi sistemde özdeş makinelerde farklı tip parça üretimi üzerine bazı önteoremlerden yararlanılarak çözüm yöntemi geliştirilmiş ve bu çözüm yöntemine yönelik deneysel çalışma yapılmıştır. Geliştirilen matematiksel model 320 problem için CPLEX ile çözdürülerek ulaşılan sonuçlar sezgisel çözüm yöntemi ile

elde edilen sonuçlar ile kıyaslanmıştır. Çözdürülen 320 problemin hepsi için CPLEX En İyi sonuçları ile sezgisel sonuçları birebir örtüşmektedir. 50 parçalık problemlerde 160 problemden 99 tanesi zaman limitine takılmıştır (Tablo 4.1). Fakat bu noktada alt sınıra olan uzaklıkla ilgili bir değerlendirme yapılmıştır. Bu değerlendirme sonucu sezgisel ile alt sınır arasındaki maksimum uzaklığın %1.69 olduğu gözlemlenmiştir (Tablo 4.2). Ayrıca yapılan genel faktöriyel tasarımında CPLEX çözüm süresi ve sezgisel - alt sınır yüzde fark üzerinde parça sayısı, işlem süreleri taban değerlerinin ve aralık seviyelerin istatistiksel olarak anlamlı faktörler olmadığı gözlemlenmiştir.

5. ÜÇ MAKİNE TEK ESNEK İŞLEME SAHİP SİSTEMLER

Bu bölümde 3 makineli ve tek esnek işlemlili akış atölyeleri ele alınmıştır. 3 makineli akış tipi sistemlerde esnek makinelerin makine 1 ve 2 olduğu, makine 2 ve 3 (Problem 2) olduğu veya üç makinenin de esnek olduğu (Problem 3) alternatifler mevcuttur. Daha önce de belirtildiği üzere esnek işlemin makine 1 ve 2’de işlenebilme durumu esnek işlemin makine 2 ve 3’te işlenebilme durumunun ters durumudur. Pinedo’nun [48] ispatladığı ters dönebilirlik özelliği ile bu problemlerden biri için geliştirilen çözüm yöntemi diğer problemi de çözmektedir. Bu bölümde esnek işlemin makine 2 ve 3 arasında olduğu durum ve esnek işlemin üç makinede de gerçekleştirilebildiği durumlar ayrı ayrı ele alınmıştır. NP-Zor olan bu problemler için sezgisel algoritmalar geliştirilmiş ve küçük boyutlu problemler ve geliştirilmiş olan matematiksel model kullanılarak kapsamlı deneysel çalışmalarla sezgisellerin performansları test edilmiştir.

5.1 Esnek İşlemin Makine 2 ve 3 Üzerinde İşlenebildiği Durum

Bu problem Bölüm 3’te, Problem 2 olarak tanıtılmıştır. Bu problemde esnek makineler 2. ve 3. makinelerdir. Probleme yönelik matematiksel model Bölüm 3’te verilmiştir. Problemin çözümü için esnek işlemin atanması ve parçaların sıralanması gereklidir. Garey v.d [22] 3-makine akış tipi sistemde esnek işlem olmasa bile yayılma zamanı en küçüklenmesi probleminin NP-Zor sınıfına girdiğini ispatlamışlardır. Bu problemde 3-makine akış tipi problemine ek olarak esnek işlem atanmasının da belirlenmesi problemin karmaşıklığını arttırmaktadır. Bu durum çalışmanın bu kısmında ele aldığımız bu problemin de NP-zor sınıfına ait olduğunu kanıtlamaktadır. Geliştirilen matematiksel model ile özellikle parça sayısının fazla olduğu büyük problemler için

optimal sonuçlar elde etmek çok uzun süreler almaktadır. Bu sebeple problemin çözümü için polinom zamanda çözüm üreten çeşitli sezgiseller önerilmiştir. Bir sonraki alt bölümde bu çözüm yöntemleri anlatılacaktır.

5.1.1 Sezgisel Çözüm Yöntemleri

Bu bölümde ele alınan problem 2'nin çözümü için, literatür kısmında genişçe anlatılan CDS, Gupta ve NEH sezgisellerini temel alan toplam 11 farklı sezgisel geliştirilmiştir. m -makinelikli sistemler için geliştirilmiş bu sezgiseller, literatürde önemli yere sahip olan sezgisellerdir. Özellikle NEH algoritması [42] optimal değerlere çok yakın etkin sonuçlar vermektedir. CDS [14] ve Gupta [32] algoritmalarında belirlenen eğim indekslerine göre parçalar sıralanarak etkili sonuçlara ulaşmak hedeflenmiştir. NEH [42] algoritmasında ise parçalar, işlem süreleri toplamı üzerinden belirlenen bir kriterle sıralanmaya çalışılmıştır. Bizim çalışmamızda bu sezgiseller modifiye edilerek hem esnek işlem atamasına hem de parça sıralanmasına karar verilmektedir.

Gupta [32] algoritmasını esas alan farklı kriterlere sahip toplam dört farklı sezgisel geliştirilmiştir. p_i^2 , p_i^3 sembolleri, esnek işlem ataması sonrası parça i 'nin makine 2 ve 3 üzerindeki işlem zamanlarını gösterebilir. Tablo 5.1 Gupta temel alınarak geliştirilmiş sezgisel alternatiflerinin atama ve sıralama kriterlerini göstermektedir. Sezgisellerden en verimli sonucu elde etmek amacıyla, minimum değeri veren alternatifi seçilmektedir. Kolay anlaşılır olması amacıyla, bu dört sezgiselden edinilen en iyi sonuç "Min_Gupta" olarak adlandırılmıştır. Bu algoritmaların sözde kodları Algoritma 2 – 5 olarak sonraki sayfalarda verilmiştir.

Min_Gupta: $\min\{\text{Gupta 1, Gupta 2, Gupta 3, Gupta 4}\}$

Geliştirilen ilk sezgiselde, parçaların makineler üzerindeki sabit ve esnek işlem zamanları toplamından minimum değere sahip olan makineye esnek işlem ataması yapılmaktadır. Atama işlemi her parça için gerçekleştirildikten sonra, eğim indeksi esnek işlem ataması sonrası elde edilen p_i^2 ve p_i^3 değerleri üzerinden hesaplanmaktadır.

Tablo 5.1: Problem 2: Alternatif Gupta Algoritmaları

Alternatifler	Atama	Sıralama
Gupta 1	$\min(f_i^2 + s_i^2, f_i^3 + s_i^3)$	p_i^m
Gupta 2	$\min(s_i^2, s_i^3)$	f_i^m
Gupta 3	$\min(f_i^2 + s_i^2, f_i^3 + s_i^3)$	f_i^m
Gupta 4	$\min(s_i^2, s_i^3)$	p_i^m

Algoritma 2 Modifiye Gupta Algoritması 1

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: $T_i^1, T_i^2, T_i^3, C_{max}$.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$a_i \leftarrow f_i^2 + s_i^2$.

$b_i \leftarrow f_i^3 + s_i^3$.

Eğer $a_i < b_i$ ise

$p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$.

Eğer $a_i > b_i$ ise

$p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$,

Eğer $a_i = b_i$ ise

Hesapla $\min_i(s_i^2, s_i^3)$.

Eğer $s_i^2 < s_i^3$ ise $p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$,

Eğer $s_i^2 > s_i^3$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$.

Gupta İndeksi Hesabı yap:

Eğer $p_i^1 > p_i^3$ ise Hesapla $GuptaIndex_i = \frac{1}{\min_i(p_i^1 + p_i^2, p_i^2 + p_i^3)}$

Eğer $p_i^1 < p_i^3$ ise Hesapla $GuptaIndex_i = \frac{-1}{\min_i(p_i^1 + p_i^2, p_i^2 + p_i^3)}$

bitiş döngü{i}

$i = 1; i \leq n; i++$ (bütün parçalar için)

Hesaplanan $GuptaIndex_i$ değerlerini küçükten büyüğe sırala

bitiş döngü{i}

Oluşturulan sıra için C_{max} değerini bul

Alternatif 2 için ise eğim indeksi sadece sabit işlemler üzerinden hesaplanmış ve her parça için esnek işlem ataması sadece esnek işlem süresi hangi makinede en küçük değere sahip ise o makineye atanmıştır. Gupta [32] eğim indeksi ile elde edilen sıralamaya göre sıralanan parçalar, belirlenen esnek işlem atamalarına göre çizelgelenirler. Çizelge sonucu elde edilen yayılma zamanı, C_{max} değerini verir.

Algoritma 3 Modifiye Gupta Algoritması 2

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: $T_i^1, T_i^2, T_i^3, C_{max}$.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$a_i \leftarrow f_i^2 + s_i^2$.

$b_i \leftarrow f_i^3 + s_i^3$.

Eğer $s_i^2 < s_i^3$ ise

$p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$.

Eğer $s_i^2 > s_i^3$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$,

Eğer $s_i^2 = s_i^3$ ise

Hesapla $\min_i(a_i, b_i)$.

Eğer $a_i < b_i$ ise $p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$,

Eğer $a_i > b_i$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$.

bitiş döngü{i}

$i = 1; i \leq n; i++$ (bütün parçalar için)

Eğer $f_i^1 < f_i^3$ ise $GuptaIndex_i = \frac{-1}{\min_i(f_i^1 + f_i^2, f_i^2 + f_i^3)}$

Eğer $f_i^1 \geq f_i^3$ ise $GuptaIndex_i = \frac{1}{\min_i(f_i^1 + f_i^2, f_i^2 + f_i^3)}$

Hesaplanan $GuptaIndex_i$ değerlerini küçükten büyüğe sırala

bitiş döngü{i}

Oluşturulan sıra için C_{max} değerini bul

3. alternatif için de 2. alternatife benzer olarak eğim indeksi sadece sabit işlemler üzerinden hesaplanmıştır; fakat, her parça için esnek işlem ataması sabit ve esnek işlem süresi toplamının en küçük değere sahip olduğu makineye göre belirlenmiştir atanmıştır.

Alternatif 4'te her parça için esnek işlem ataması sadece esnek işlem süresinin en küçük değere sahip olduğu makineye göre belirlenmiştir. Aynı zamanda, eğim indeksinin belirlenmesi için esnek işlem atamaları belirlendikten sonra elde edilen işlem zamanlarına göre belirlenmiştir.

Algoritma 2 – 5'de verilen sözde kodlar küçük bir probleme uygulanmıştır. Probleme ait veriler ve uygulama sonucu elde edilen değerler Örnek 2'de yer almaktadır.

Algoritma 4 Modifiye Gupta Algoritması 3

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: $T_i^1, T_i^2, T_i^3, C_{max}$.

$i = 1; i \leq n; i ++$ (bütün parçalar için)

$$a_i \leftarrow f_i^2 + s_i^2.$$

$$b_i \leftarrow f_i^3 + s_i^3.$$

Eğer $a_i < b_i$ ise

$$p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3.$$

Eğer $a_i > b_i$ ise

$$p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3,$$

Eğer $a_i = b_i$ ise

Hesapla $\min_i(s_i^2, s_i^3)$.

Eğer $s_i^2 < s_i^3$ ise $p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$,

Eğer $s_i^2 > s_i^3$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$.

Gupta İndeksi Hesabı yap:

Eğer $f_i^1 < f_i^3$ ise hesapla $GuptaIndex_i = \frac{-1}{\min_i(f_i^1 + f_i^2, f_i^2 + f_i^3)}$

Eğer $f_i^1 \geq f_i^3$ ise hesapla $GuptaIndex_i = \frac{1}{\min_i(f_i^1 + f_i^2, f_i^2 + f_i^3)}$

bitiş döngü{i}

$i = 1; i \leq n; i ++$ (bütün parçalar için)

Hesaplanan $GuptaIndex_i$ değerlerini küçükten büyüğe sırala

bitiş döngü{i}

Oluşturulan sıra için C_{max} değerini bul

Algoritma 5 Modifiye Gupta Algoritması 4

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: $T_i^1, T_i^2, T_i^3, C_{max}$.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$a_i \leftarrow f_i^2 + s_i^2$.

$b_i \leftarrow f_i^3 + s_i^3$.

Eğer $s_i^2 < s_i^3$ ise

$p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$.

Eğer $s_i^2 > s_i^3$ ise

$p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$,

Eğer $s_i^2 = s_i^3$ ise hesapla $\min_i(a_i, b_i)$.

Eğer $a_i < b_i$ ise $p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$,

Eğer $a_i > b_i$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$.

Gupta İndeksi Hesabı yap:

Eğer $p_i^1 < p_i^3$ ise hesapla $GuptaIndex_i = \frac{-1}{\min_i(p_i^1 + p_i^2, p_i^2 + p_i^3)}$

Eğer $p_i^1 \geq p_i^3$ ise hesapla $GuptaIndex_i = \frac{-1}{\min_i(p_i^1 + p_i^2, p_i^2 + p_i^3)}$

bitiş döngü{i}

$i = 1; i \leq n; i++$ (bütün parçalar için)

Hesaplanan $GuptaIndex_i$ değerlerini küçükten büyüğe sırala

bitiş döngü{i}

Oluşturulan sıra için C_{max} değerini bul

Örnek 2 İşlem süreleri Tablo 5.2’de verilen 5 parçalı problem dört Gupta algoritması ile çözdürülmüştür.

Modifiye Gupta algoritmalarından elde edilen sonuca göre atama sıralama ve yayılma zamanı (C_{max}) değerleri sırasıyla Tablo 5.3, 5.4 ve 5.5’da yer almaktadır. Tablo 5.3’ya göre Gupta 1 ve Gupta 4’ e göre tüm parçaların esnek işlemleri 3. makineye atanmıştır. Benzer şekilde Gupta 2 ve Gupta 3’e uygun yapılan atamalarda parça 1 ve 4’ün esnek işlemleri makine 2’ye atanırken, diğer parçaların esnek işlemleri makine 3’tedir. Algoritma adımlarına göre Gupta indekslerine göre yapılan sıralamada Gupta 1, 2 ve 4 aynı sıralamayı vermektedir. Bu sıralamaya göre ilk önce parça 2 işlenmekte ardından parça 3, sonrasında sırasıyla parça 4, 5 ve 1 işlem görmektedir. Gupta 3’te ise bu sıralamadan farklı olarak parça 4 ve 5’in yeri değişmektedir.

Yapılan bu atama ve sıralamalar sonucu elde edilen yayılma zamanı değeri Tablo 5.5’da yer almaktadır. Bu örnekte tesadüf sonucu Gupta 1 ve 4 aynı atama ve sıralama ile sonuçlanmıştır dolayısıyla yayılma zamanları da aynı değeri vermiştir. Gupta 2 ve 3 için sıralamalar farklı olmasına rağmen aynı sonuca ulaşılmıştır. Bu örnek için

Gupta 2 ve 3 daha iyi sonuç vermiştir. Gupta 1, 2 ve 4'ün aynı sıralamaya sahip olduğu düşünülürse esnek işlem atamasının işlem süresinin daha az olduğu makineye yapmak bu örnek için avantaj sağlamıştır. Bu sonuçların hiç birinde optimal değere ulaşamamıştır.

Tablo 5.2: Örnek 2: Parçalara Göre İşlem Süreleri

	f^1	f^2	f^3	s^2	s^3
Parça 1	12	182	77	182	195
Parça 2	15	82	56	173	170
Parça 3	15	140	70	188	181
Parça 4	12	147	50	158	161
Parça 5	15	146	36	156	155

Tablo 5.3: Örnek 2: Modifiye Gupta Algoritmalarına Göre Esnek İşlem Atamaları

	Parça 1	Parça 2	Parça 3	Parça 4	Parça 5
Gupta 1	3	3	3	3	3
Gupta 2	2	3	3	2	3
Gupta 3	2	3	3	2	3
Gupta 4	3	3	3	3	3

Tablo 5.4: Örnek 2: Modifiye Gupta Algoritmalarına Göre Parça Sıralaması

	Sıra 1	Sıra 2	Sıra 3	Sıra 4	Sıra 5
Gupta 1	2	3	4	5	1
Gupta 2	2	3	4	5	1
Gupta 3	2	3	5	4	1
Gupta 4	2	3	4	5	1

Önerilen çözüm yöntemlerinden bir diğeri CDS [14] algoritmasının temel alındığı 4 sezgiselden oluşmaktadır. CDS [14] algoritmasında amaç m -makine akış tipi sistemi $(m - 1)$ tane yapay 2-makine problemine dönüştürerek Johnson [39] algoritmasından

Tablo 5.5: Örnek 2: Modifiye Gupta Algoritmalarına Göre Yayılma Zamanı Değerleri

Yayılma Zamanı Değerleri	
<i>Gupta 1</i>	1248
<i>Gupta 2</i>	1129
<i>Gupta 3</i>	1129
<i>Gupta 4</i>	1248

yararlanmaktadır. Johnson [39] algoritmasından elde edilen sıralama kullanılarak orjinal problemin yayılma zamanı değeri hesaplanır. Çalışmanın bu bölümünde yer alan problem için esnek işlem atamasına karar vermek için orjinal CDS [14] algoritması üzerinden bir takım değişiklikler yapılmıştır ve bu algoritmanın 3-makine için geçerli olan formülasyonundan yararlanılmıştır. Modifiye edilmiş versiyonların hepsi için sözde kodlar Algoritma 6 – 9 olarak verilmiştir. Sözde kodlarda yer alan p_i^2 ve p_i^3 notasyonları Modifiye Gupta Algoritmalarındaki gibi esnek makineler üzerinde esnek işlem ataması sonrası sahip olunan işlem süresini göstermektedir. Farklı olarak bu algoritmada kullanılan yapay iki makine problemleri için *artificial* p_i^1 ve *artificial* p_i^2 notasyonları belirlenmiştir. Bu notasyon ise yapay problemlerdeki makine 1 ve 2 üzerindeki işlem sürelerini göstermektedir. Alternatiflerden ilki için önce minimum sabit ve esnek işlem süresi toplamı üzerinden esnek işlem ataması yapılmıştır ve esnek işlem ataması sonrası belirlenen toplam işlem süreleri ile yapay problemler oluşturulup Johnson [39] algoritması uygulanmıştır. Sonrasında ise Johnson [39] algoritmasının verdiği sıralama yapılan esnek işlem ataması ile orjinal probleme uygulanmıştır.

Tablo 5.6: Problem 2: Alternatif CDS Algoritmaları

Alternatifler	Atamalar	Yapay 1	Yapay 2
CDS 1	$\min(f_i^2 + s_i^2, f_i^3 + s_i^3)$	p_i^1, p_i^3	$p_i^1 + p_i^2, p_i^2 + p_i^3$
CDS 2	$\min(s_i^2, s_i^3)$	f_i^1, f_i^3	$f_i^1 + f_i^2, f_i^2 + f_i^3$
CDS 3	$\min(s_i^2, s_i^3)$	p_i^1, p_i^3	$p_i^1 + p_i^2, p_i^2 + p_i^3$
CDS 4	$\min(f_i^2 + s_i^2, f_i^3 + s_i^3)$	f_i^1, f_i^3	$f_i^1 + f_i^2, f_i^2 + f_i^3$

Gupta modifiyeleri ile benzer şekilde, Tablo 5.6 CDS alternatifleri için yapay makineler için işlem sürelerinin ve atamaların neye göre belirlendiğini özetlemektedir.

Bununla birlikte, CDS tabanlı sezgiseller için de en iyi sonucu elde etmek için minimum değeri alınmaktadır ve bu algoritma "Min_CDS" olarak belirtilmektedir.

Min_CDS: $\min\{\text{CDS 1, CDS 2, CDS 3, CDS 4}\}$

Algoritma 6 Modifiye CDS Algoritması 1

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: $T_i^1, T_i^2, T_i^3, C_{max}$.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$a_i \leftarrow f_i^2 + s_i^2$.

$b_i \leftarrow f_i^3 + s_i^3$.

Eğer $a_i < b_i$ ise

$p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$.

Eğer $a_i > b_i$ ise

$p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$,

Eğer $a_i = b_i$ ise

Hesapla $\min_i(s_i^2, s_i^3)$.

Eğer $s_i^2 < s_i^3$ ise $p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$,

Eğer $s_i^2 > s_i^3$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$.

Yapay 2 makine problemleri oluştur.

bitiş döngü{i}

$i = 1; i \leq n; i++$ (bütün parçalar için)

$artificialp_i^1 \leftarrow p_i^1; artificialp_i^2 \leftarrow p_i^3$ hesapla.

bitiş döngü{i}

Johnson kuralı ile işleri sırala

$i = 1; i \leq n; i++$ (bütün parçalar için)

$artificialp_i^1 \leftarrow p_i^1 + p_i^2; artificialp_i^2 \leftarrow p_i^2 + p_i^3$ şeklinde yeniden hesapla.

bitiş döngü{i}

Johnson kuralı ile işleri sırala

Elde ettiğin sıralamalar için eski p_i^1, p_i^2, p_i^3 değerleri ile C_{max}^1 ve C_{max}^2 değerlerini hesapla.

$\min = \{C_{max}^1, C_{max}^2\}$ değerini al.

İkinci alternatif için minimum esnek işlem süresi üzerinden esnek işlem ataması yapılmıştır. Yapay problemler sadece sabit işlem süreleri ile oluşturulmuştur. Johnson [39] algoritmasının uygulanması ile ulaşılmış sıralama kullanılarak her parça için esnek işlem ataması sonrası elde edilen toplam işlem zamanları ile problem çizelgenmektedir.

Algoritma 7 Modifiye CDS Algoritması 2

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: $T_i^1, T_i^2, T_i^3, C_{max}$.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$$a_i \leftarrow f_i^2 + s_i^2.$$

$$b_i \leftarrow f_i^3 + s_i^3.$$

Eğer $s_i^2 < s_i^3$ ise

$$p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3.$$

Eğer $s_i^2 > s_i^3$ ise

$$p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3,$$

Eğer $s_i^2 = s_i^3$ ise

Hesapla $\min_i(a_i, b_i)$.

Eğer $a_i < b_i$ ise $p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$,

Eğer $a_i > b_i$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$.

bitiş döngü{i}

Yapay 2 makine problemleri oluştur.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$artificial p_i^1 \leftarrow f_i^1; artificial f_i^1 \leftarrow f_i^3$ hesapla.

bitiş döngü{i}

Johnson kuralı ile işleri sırala

$i = 1; i \leq n; i++$ (bütün parçalar için)

$artificial p_i^1 \leftarrow f_i^1 + f_i^2; artificial p_i^2 \leftarrow f_i^2 + f_i^3$ şeklinde yeniden hesapla.

bitiş döngü{i}

Johnson kuralı ile işleri sırala

Elde ettiğin sıralamalar için eski p_i^1, p_i^2, p_i^3 değerleri ile C_{max}^1 ve C_{max}^2 değerlerini hesapla.

$\min = \{C_{max}^1, C_{max}^2\}$ değerini al.

Alternatif 1 ve 2'ye benzer olarak 3. ve 4. alternatiflerde de esnek işlem ataması ve yapay problemlerde kullanılan işlem zamanının esnek işlem ataması ile birlikte değerlendirilip değerlendirilmemesine bağlı olarak değişimi göstermektedir. Üçüncü alternatif için minimum esnek işlem süresi üzerinden esnek işlem ataması yapılmış ve yapay problemler bu esnek işlem ataması sonrası elde edilen işlem süreleri ile oluşturulmuştur. Diğer alternatiflerde olduğu gibi yapay 2 makine problemleri Johnson [39] algoritması ile çözümlenerek, ulaşılan sıralama orjinal probleme uygulanarak, yayılma zamanına ulaşılmaktadır.

Algoritma 8 Modifiye CDS Algoritması 3

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: $T_i^1, T_i^2, T_i^3, C_{max}$.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$a_i \leftarrow f_i^2 + s_i^2$.

$b_i \leftarrow f_i^3 + s_i^3$.

Eğer $s_i^2 < s_i^3$ ise

$p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$.

Eğer $s_i^2 > s_i^3$ ise

$p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$,

Eğer $s_i^2 = s_i^3$ ise

Hesapla $\min_i(a_i, b_i)$.

Eğer $a_i < b_i$ ise $p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$,

Eğer $a_i > b_i$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$.

bitiş döngü{i}

Yapay 2 makine problemleri oluştur.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$artificialp_i^1 \leftarrow p_i^1; artificialp_i^2 \leftarrow p_i^3$ hesapla.

bitiş döngü{i}

Johnson kuralı ile işleri sırala

$i = 1; i \leq n; i++$ (bütün parçalar için)

$artificialp_i^1 \leftarrow p_i^1 + p_i^2; artificialp_i^2 \leftarrow p_i^2 + p_i^3$ şeklinde yeniden hesapla.

bitiş döngü{i}

Johnson kuralı ile işleri sırala

Elde ettiğin sıralamalar için eski p_i^1, p_i^2, p_i^3 değerleri ile C_{max}^1 ve C_{max}^2 değerlerini hesapla.

$\min = \{C_{max}^1, C_{max}^2\}$ değerini al.

4. alternatifte ise esnek işlem atamaları, sabit ve esnek işlem süresi toplamının en küçük değeri verdiği makineye yapılmıştır. Yapay problemler de bu atamalara göre elde edilen işlem süreleri üzerinden oluşturulmuştur.

Algoritma 9 Modifiye CDS Algoritması 4

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: $T_i^1, T_i^2, T_i^3, C_{max}$.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$$a_i \leftarrow f_i^2 + s_i^2.$$

$$b_i \leftarrow f_i^3 + s_i^3.$$

Eğer $a_i < b_i$ ise

$$p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3.$$

Eğer $a_i > b_i$ ise

$$p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3,$$

Eğer $a_i = b_i$ ise

Hesapla $\min_i(s_i^2, s_i^3)$.

Eğer $s_i^2 < s_i^3$ ise $p_i^2 \leftarrow f_i^2 + s_i^2, p_i^3 \leftarrow f_i^3$,

Eğer $s_i^2 > s_i^3$ ise $p_i^2 \leftarrow f_i^2, p_i^3 \leftarrow f_i^3 + s_i^3$.

bitiş döngü{i}

Yapay 2 makine problemleri oluştur.

$i = 1; i \leq n; i++$ (bütün parçalar için)

$artificialp_i^1 \leftarrow f_i^1; artificialp_i^2 \leftarrow f_i^3$ hesapla.

bitiş döngü{i}

Johnson kuralı ile işleri sırala

$i = 1; i \leq n; i++$ (bütün parçalar için)

$artificialp_i^1 \leftarrow f_i^1 + f_i^2; artificialp_i^2 \leftarrow f_i^2 + f_i^3$ şeklinde yeniden hesapla.

bitiş döngü{i}

Johnson kuralı ile işleri sırala

Elde ettiğin sıralamalar için eski p_i^1, p_i^2, p_i^3 değerleri ile C_{max}^1 ve C_{max}^2 değerlerini hesapla.

$\min = \{C_{max}^1, C_{max}^2\}$ değerini al.

Örnek 3 İşlem süreleri Tablo 5.2’de verilen 5 parçalı problem Örnek 2’de olduğu gibi Örnek 3’te de modifiye CDS algoritmaları ile çözdürülmüştür. Modifiye CDS algoritmalarından elde edilen sonuca göre atama sıralama ve yayılma zamanı (C_{max}) değerleri sırasıyla Tablo 5.7, 5.8, 5.9 ve 5.10’da yer almaktadır.

Bu algoritmada da Modifiye Gupta Algoritmalarına benzer şekilde 1., 4. ve 2., 3. modifiyelerin esnek işlem atamasına aynı şekilde karar verilmektedir. Tablo 5.7’ya göre CDS 1 ve CDS 4’ e göre tüm parçaların esnek işlemleri 3. makineye atanmıştır.

Yapılan bu atama ve sıralamalar sonucu elde edilen yayılma zamanı değeri Tablo 5.10’da yer almaktadır. Bu tablodaki "Yapay 1 C_{max} " kolonun farklı CDS modifiyelerine göre oluşan iki yapay problemde ilki ile elde edilen yayılma zamanının değerini verirken "Yapay 2 C_{max} " ikinci yapay problemin sonuçlarını vermektedir. Nihai yayılma zamanı değeri ise bu iki değerden küçük olanın alınmasıyla elde edilmektedir. Gupta sonuçlarına benzer olarak raslantısal şekilde CDS 1 ve 4 ile CDS 2 ve 3 algoritmaları ile aynı sonuçlara ulaşılmıştır. CDS 2 ve 3 ile elde edilen değerler esnek işlem atamasının sonucu olarak daha iyidir. Fakat bunun için bir genelleme yapmak mümkün değildir.

Tablo 5.7: Örnek 3: Modifiye CDS Algoritmalarına Göre Esnek İşlem Atamaları

	Parça 1	Parça 2	Parça 3	Parça 4	Parça 5
CDS 1	3	3	3	3	3
CDS 2	2	3	3	2	3
CDS 3	2	3	3	2	3
CDS 4	3	3	3	3	3

Tablo 5.8: Örnek 3: Modifiye CDS Algoritmalarına Göre 1. Yapay Problem Parça Sıralaması

	Sıra 1	Sıra 2	Sıra 3	Sıra 4	Sıra 5
CDS 1	1	4	2	3	5
CDS 2	1	4	2	3	5
CDS 3	1	4	2	3	5
CDS 4	1	4	2	3	5

Tablo 5.9: Örnek 3: Modifiye CDS Algoritmalarına Göre 2. Yapay Problem Parça Sıralaması

	Sıra 1	Sıra 2	Sıra 3	Sıra 4	Sıra 5
CDS 1	2	3	4	5	1
CDS 2	2	3	4	5	1
CDS 3	2	3	4	5	1
CDS 4	2	3	4	5	1

Tablo 5.10: Örnek 3: Modifiye CDS Algoritmalarına Göre Yayılma Zamanı Değerleri

	Yapay 1 C_{max}	Yapay 2 C_{max}
CDS 1	1345	1248
CDS 2	1489	1129
CDS 3	1489	1129
CDS 4	1345	1248

Esnek makinelerin 2 ve 3 olduğu, tek esnek işleme sahip 3-makinelili akış tipi sistemde yayılma zamanını enküçüklemek için Gupta [32] ve CDS [14] tabanlı sezgisellere ek olarak Nawaz v.d. [42] tarafından geliştirilen NEH algoritmasından da yararlanılmıştır. Orjinal NEH [42] sezgiselinde NEH kriteri parçaların makineler üzerindeki işlem zamanları toplamıdır. Modifiye edilen sezgiselde ise NEH kriteri sabit işlem zamanlarına ek olarak esnek işlem sürelerinin değerlerini de barındırmaktadır. Tablo 5.11’de gösterildiği üzere 3 farklı kriter belirlenmiştir. NEH_i notasyonu NEH kriterinin değerini göstermek için kullanılmıştır. Değiştirilmiş sezgisellerin sözde kodları Algoritma 10 – 12’de yer almaktadır. En verimli sonuçlara ulaşabilmek için alternatifler arasında en küçük değeri veren "Min_NEH" sonuçları kullanılmıştır.

Min_NEH: $\min\{NEH_1, NEH_2, NEH_3\}$

Bu çalışmada çözüm yöntemlerinden biri olarak önerilen NEH [42] modifiyesinde, parçalar NEH_i değerlerine göre artmayan şekilde sıralandıktan sonra bu sıralamaya göre gelen ilk iki iş için optimal esnek işlem ataması ve optimal sıralama belirlenmektedir. Bu iki iş için elde edilen optimal çizelge sabitlenerek her yeni gelen

iş bu çizelgedeki esnek işlem atamasını ve parça sıralamasını bozmayacak şekilde çizelgelenmektedir. Yeni gelen işler, NEH kriterine göre yapılan sıralamaya göre belirlenmektedir. Her k . iş için k olası yer ve 2 farklı esnek işlem ataması mevcuttur. Olası yerler ile olası atamalardan en küçük yayılma zamanına sahip olan çizelge her k işi için seçilir ve çizelgelenmeyen iş kalmayana kadar bu basamaklar tekrarlanır.

Tablo 5.11: Problem 2: Alternatif NEH Algoritmaları

Alternatifler	Sıralama Kriteri
NEH 1	$f_i^1 + f_i^2 + f_i^3 + s_i^2 + s_i^3$
NEH 2	$f_i^1 + f_i^2 + f_i^3 + \min(s_i^2, s_i^3)$
NEH 3	$f_i^1 + f_i^2 + f_i^3 + \max(s_i^2, s_i^3)$

İlk alternatifte NEH 1 kriteri olarak sabit ve esnek işlem zamanlarının tüm makineler üzerindeki toplamı ($f_i^1 + f_i^2 + s_i^2 + f_i^3 + s_i^3$) kullanılmıştır.

Algoritma 10 Modifiye NEH Algoritması 1

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: C_{max}

$i = 1; i \leq n; i++$ (bütün parçalar için)

$NEH_i = f_i^1 + f_i^2 + s_i^2 + f_i^3 + s_i^3$

NEH_i değerlerini büyükten küçüğe sırala

bitiş döngü{i}

Sıralamaya göre öncelikle en baştaki k işi için $p_k^2 \leftarrow f_k^2 + s_k^2, p_k^3 \leftarrow f_k^3$, ve ikinci sıradaki l işi için $p_l^2 \leftarrow f_l^2, p_l^3 \leftarrow f_l^3 + s_l^3$ ataması ile k, l işleri için yayılma zamanını bul.

Yukarıdaki adımı sıralama ve atama değişecek şekilde 4 kombinasyon için uygula.

En küçük yayılma zamanını veren atama ve sıralamayı sabitle.

Çizelgelenmeyen iş kalmayana kadar Tekrarla

Yeni eklenen iş için esnek işlem atamasını sırasıyla 2. ve 3. makineye atayarak ve sıralamayı bir önceki adımda elde edilen sıralamayı bozmadan yayılma zamanını hesapla.

Son tekrar

İkinci alternatifte NEH kriteri değişmiştir fakat sezgiselin geri kalan işleyişi önerilen ilk NEH [42] algoritması ile aynıdır. Bu algoritmadaki NEH 2 parçaların 3 makine üzerindeki sabit işlem zamanları ve esnek işlem zamanlarından minimum olanın toplamına ($f_i^1 + f_i^2 + f_i^3 + \min(s_i^2, s_i^3)$) eşittir.

Algoritma 11 Modifiye NEH Algoritması 2

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: C_{max}

$i = 1; i \leq n; i ++$ (bütün parçalar için)

$$NEH_i = f_i^1 + f_i^2 + f_i^3 + \min(s_i^2, s_i^3)$$

NEH_i değerlerini büyükten küçüğe sırala

bitiş döngü{i}

Sıralamaya göre öncelikle en baştaki k işi için $p_k^2 \leftarrow f_k^2 + s_k^2, p_k^3 \leftarrow f_k^3$, ve ikinci sıradaki l işi için $p_l^2 \leftarrow f_l^2, p_l^3 \leftarrow f_l^3 + s_l^3$ ataması ile k, l işleri için yayılma zamanını bul.

Yukarıdaki adımı sıralama ve atama değişecek şekilde 4 kombinasyon için uygula.

En küçük yayılma zamanını veren atama ve sıralamayı sabitle.

Çizelgelenmeyen iş kalmayana kadar Tekrarla

Yeni eklenen iş için esnek işlem atamasını sırasıyla 2. ve 3. makineye atayarak ve sıralamayı bir önceki adımda elde edilen sıralamayı bozmadan yayılma zamanını hesapla.

Son tekrar

Üçüncü alternatifteki NEH 3 kriteri parçaların 3 makine üzerindeki sabit işlem zamanları ve esnek işlem zamanlarından maksimum olanın toplamına ($f_i^1 + f_i^2 + f_i^3 + \max(s_i^2, s_i^3)$) eşittir.

Tablo 5.2'deki veriler üzerinde NEH tabanlı oluşturulan bu algoritmalar uygulanmıştır. Bu uygulamalara dair sonuçlar Örnek 4'te yer almaktadır.

Örnek 4 İşlem süreleri Tablo 5.2 verilen 5 parçalı problem NEH 1, 2 ve 3 ile çözdürülmüştür. Algoritmanın ilk adımı olarak hesaplanan NEH kriterleri parçalara ve farklı NEH modifiyelerine göre Tablo 5.12'dedir.

Tablo 5.12: Örnek 4: Parçalara Göre NEH Kriterleri

	NEH 1	NEH 2	NEH 3
Parça 1	648	453	466
Parça 2	496	323	326
Parça 3	594	406	413
Parça 4	528	367	370
Parça 5	508	352	353

Algoritma 12 Modifiye NEH Algoritması 3

Girdi: $f_i^1, f_i^2, f_i^3, s_i^2, s_i^3$.

Çıktı: C_{max}

$i = 1; i \leq n; i++$ (bütün parçalar için)

$$NEH_i = f_i^1 + f_i^2 + f_i^3 + \max(s_i^2, s_i^3)$$

NEH_i değerlerini büyükten küçüğe sırala

bitiş döngü{i}

Sıralamaya göre öncelikle en baştaki k işi için $p_k^2 \leftarrow f_k^2 + s_k^2, p_k^3 \leftarrow f_k^3$, ve ikinci sıradaki l işi için $p_l^2 \leftarrow f_l^2, p_l^3 \leftarrow f_l^3 + s_l^3$ ataması ile k, l işleri için yayılma zamanını bul.

Yukarıdaki adımı sıralama ve atama değişecek şekilde 4 kombinasyon için uygula.

En küçük yayılma zamanını veren atama ve sıralamayı sabitle.

Çizelgelenmeyen iş kalmayana kadar Tekrarla

Yeni eklenen iş için esnek işlem atamasını sırasıyla 2. ve 3. makineye atayarak ve sıralamayı bir önceki adımda elde edilen sıralamayı bozmadan yayılma zamanını hesapla.

Son tekrar

Algoritmanın ikinci adımı olarak NEH kriterlerine göre büyükten küçüğe doğru yapılan sıralama sonucu farklı NEH'ler için elde edilmiş sıralamalar Tablo 5.13'te yer almaktadır.

NEH kriterleri tesadüfen tüm parçalar için aynı şekilde sıralanmıştır (Tablo 5.13). Algoritmanın ilerleyen adımlarının tüm NEH modifiyeleri için aynı olması nedeniyle, NEH 1, 2 ve 3 aynı esnek işlem atamasına, aynı parça sıralamasına ve dolayısıyla aynı yayılma zamanı değerlerine sahiptir. NEH modifiyeleri ile ulaşılan atama sonucu parça 2 ve 3'ün esnek işlemi makine 2'ye, parça 1, 4 ve 3'ün esnek işlemi ise makine 3'e atanmıştır. Bu atamalar gerçekleşirken parçalara dair sıralama da $(J_4, J_1, J_5, J_3, J_2)$ şeklinde olmuştur. Bu atamalar ve sıralama sonucu elde edilen yayılma zamanı değeri 1120'dir.

Oluşturulan matematiksel model ile ulaşılan optimal çözümde esnek işlem atamaları parça 1 dışında diğer tüm parçalar için makine 3'e atanmıştır. Parça 1'in esnek işlemi ise makine 2'de işlenmektedir. Optimal çözüme ait sıralama $(J_2, J_4, J_5, J_3, J_1)$ şeklindedir. Bu sıralama ve atama yayılma zamanı değeri olarak 1053'ü vermiştir. 11 algoritma arasında optimal çözüme en yakın sonuç NEH modifiyeleri ile elde edilmiştir.

Tablo 5.13: Örnek 4: Farklı NEH Kriterlerine Göre Parça Sıralaması

	Sıra 1	Sıra 2	Sıra 3	Sıra 4	Sıra 5
NEH 1	1	3	4	5	2
NEH 2	1	3	4	5	2
NEH 3	1	3	4	5	2

Bu bölümde 3-makine tek esnek işlemler bu akış tipi sistem için farklı sezgisellerden oluşan çözüm yöntemleri sunulmuştur. Bir sonraki alt bölümde bu sezgisellerin performansını ölçmek için oluşturulmuş bir deneysel çalışma yer almaktadır.

5.1.2 Deneysel Çalışma

Değişik tip parça üreten 3-makineli akış tipi sistemde makine 2 ve 3 arasında esnek işlemin bulunduğu problem için bir önceki bölümde geliştirilen sezgiselleri test etmek için bir deneysel çalışma düzenlenmiştir. Üretilen problemler için optimal çözümler matematiksel model ve GAMS CPLEX 12.0 çözücüsü ile elde edilmiştir. Geliştirilen sezgiseller Visual C++ Express 2010 ile kodlanmıştır. Hem matematiksel model ve hem de sezgisellerin çözümünde Intel Xeon E5645 2.4 GHz 12 çekirdekli, 12GB bellek ve 4 paralel işlemcili bir makine kullanılmıştır. Matematiksel modeller için 900 saniye zaman limiti konulmuştur. Bu süre içerisinde optimal çözüme ulaşılamaması durumunda elde edilen en iyi çözüm ve bu çözümün optimale olan uzaklığı alınmıştır.

Deneysel çalışmada 20 ve 50 parçalık problemler türetilmiştir. İşlem sürelerinin nicelik olarak büyük ya da küçük olmalarının yanında, parçalar arası işlem süreleri varyansının düşük ya da yüksek olmasının problemin zorluğuna ve geliştirilen çözüm yöntemlerinin performansına etkilerini incelemek için, bütün işlem süreleri farklı aralıklardan türetilmiştir. Bunun için aralık ve taban değer olarak ikişer seviye belirlenmiştir. Taban değerler ve aralık için ilk seviye sırasıyla 10 ve (taban değer/2), ikinci seviye 100 ve (4·taban değer) şeklindedir. Taban değerlere her iki seviyede de aralıklar eklenerek işlem zamanları belirlenmiştir. Dolayısıyla işlem zamanları seviyelerine bağlı $(2^5) \cdot 2 \cdot 2 = 128$ farklı problem kombinasyonu oluşmuştur her kombinasyon için 5 tekrar kullanılmıştır. Bu değişkenlerin farklılaşmasının sonucu

olarak matematiksel model ve sezgisel algoritmalar ile 20 ve 50 işlik problemlerde 320'şer toplamda ise 640 örnek çözdürülmüştür.

Elde edilen sonuçlardan aynı temele dayanan sezgisel gruplarının kendi aralarında minimum değerleri belirlenerek CPLEX ile ulaşılan sonuçlar arasındaki yüzde farklar hesaplanmış ve sezgisellerin performansları bu şekilde değerlendirilmiştir. Bununla birlikte temeli aynı olan sezgisellerden de baskınlık gösterenler belirlenmiştir.

Tablo 5.14: Baskın Olan Algoritma Alternatifleri

Parça Sayısı	NEH	GUPTA	CDS
20	2, 3	2	1
50	3	2	4

Tablo 5.14'de verilen değerlere göre NEH için sıralama kriteri olarak $NEH2 (f_i^1 + f_i^2 + f_i^3 + \min(s_i^2, s_i^3))$ ve $NEH3 f_i^1 + f_i^2 + f_i^3 + \max(s_i^2, s_i^3)$ kullanıldığında daha iyi sonuçlar elde edildiği görülmüştür. Gupta örnek alınarak geliştirilmiş sezgiseller arasından esnek işlem atamasının sabit ve esnek işlem zamanı toplamı küçük olan makineye ve eğim indeksi belirlenirken sadece sabit işlem zamanlarının kullanılmış olan *Gupta2* algoritmasının daha verimli olduğu anlaşılmıştır. Bununla birlikte CDS modifiyesinde 20 iş için *CDS1*, 50 iş için *CDS4* sezgiselinin kullanılmasının avantajlı olduğu görülmüştür. Bu analiz 640 problem de çözdürüldükten sonra elde edilmiştir. Bu algoritmalar arasında en verimli sonucun NEH tabanlı algoritmalarından elde edildiği Tablo 5.15'den anlaşılmaktadır. Fakat daha verimli sonuçlar elde edebilmek için kullandığımız "Min_CDS, Min_Gupta ve Min_NEH" sezgiselleri baskın alternatifleri önemsemeden optimal sonuca en yakın değerlerini vermektedir. Sezgiseller ile ulaştığımız çözümlerin kalitesini karşılaştırmak için bir yüzde farka formülü belirlenmiştir. Matematiksel model ile verilen zaman limiti içerisinde elde edilen en iyi sonuç (En iyi CPLEX) ve alt sınır (CPLEX Alt Sınır) olmak üzere iki farklı sonuç elde ettiğimiz için sezgisellerin performansı da farklı iki formül ile ölçülmektedir.

$$\text{CPLEX Alt Sınır Yüzde Fark} = ((\text{Sezgisel} - \text{CPLEX Alt Sınır}) / (\text{CPLEX Alt Sınır})) \cdot 100$$

En iyi CPLEX Yüzde Fark= ((Sezgisel - En İyi CPLEX)/(En İyi CPLEX))·100

Tablo 5.15: Problem 2: 20 İş İçin Sezgisellerin Alt Sınıra Yüzde Uzaklığı

	Min_NEH	Min_Gupta	Min_CDS
Ortalama	0.91	12.13	11.51
Maksimum	10.85	48.51	47.76

Tablo 5.16: Problem 2: 20 İş İçin Sezgisellerin En iyi Sonuca Yüzde Uzaklığı

	Min_NEH	Min_Gupta	Min_CDS
Ortalama	0.58	11.76	11.14
Maksimum	4.61	48.52	47.76

Tablo 5.15 ve 5.16’da sonuçlar 20 iş için ortalama alınarak verilmiştir. Modifiye NEH algoritmalarının en küçük değerinden elde edilen sonuç CDS ve Gupta modifiyelerine göre çok daha iyi sonuçlar vermiştir. "Min_NEH" 900 saniyelik zaman limiti içerisinde bulunan sonuçlara maksimum %4,61 uzaklıkta iken ortalama uzaklığı sadece %0,58’dir. Makinelerdeki sabit ve esnek işlem süreleri değişimine bağlı olarak oluşturulmuş kombinasyonların 5 tekrar için CPLEX alt sınır ve CPLEX en iyi sonuçlarına yüzde uzaklık ortalamalarının sonuçları ilerleyen sayfalarda Tablolar 5.21 ve 5.22’de gösterilmektedir.

Optimal çözüme uzaklık açısından değerlendirildiğinde en iyi performansı "Min_NEH" sezgiselinin verdiği görülmektedir. Bu sebeple çalışmanın bundan sonraki kısmında sadece bu sezgisel değerlendirilmiştir. Tablo 5.17 "Min_NEH" ile optimal çözüme ulaşılan problem sayısını iş sayısına ve işlem sürelerinin seviyelerine göre göstermektedir. Tablo 5.18 ise CPLEX çözücüsü ile 320 problemden zaman limitine takılarak optimal sonuca ulaşamayan problem sayılarını vermektedir.

Tablo 5.17’ye göre 20 iş için 320 problemden 74 tanesinde geliştirilen sezgisel ile CPLEX alt sınırının ulaştığı değere ulaşılmıştır. Aynı şekilde 50 iş için en az 65 durumda optimal sonuca ulaşıldığı görülmüştür. Alt sınır değeri zaman limitine takılan problemler için optimal çözümden farklı olabilir. Bu nedenle sezgisel sonucunun alt

Tablo 5.17: Problem 2: Min_NEH Optimal Çözüm Sayısı

Problem Sayısı	İşlem Süresi Taban Değeri Seviyesi		
	Düşük (10)	Yüksek (100)	Toplam
20 İş (320)	40	34	74
50 İş (320)	34	31	65

Tablo 5.18: Problem 2: CPLEX Zaman Limitine Takılan Problem Sayısı

Problem Sayısı	İşlem Süresi Taban Değeri Seviyesi		
	Düşük (10)	Yüksek (100)	Toplam
20 İş (320)	10	37	47
50 İş (320)	50	85	135

sınırla karşılaştırılması zaman limitine takılan problemler için sezgiselin performansını gereksiz yere düşürmektedir. Bu sebeple karşılaştırmalarda hem alt sınır hem de CPLEX'in bulunduğu en iyi tamsayı değer ayrı ayrı kullanılmıştır.

Makineler üzerindeki işlem sürelerinin sahip olduğu taban değerlerin seviyesine göre yapılan analizde "Min_NEH" ile edinilen sonuçların alt sınıra göre kaliteli sonuçlar olduğu Tablo 5.19'da gözlemlenmektedir. Bazı problemlerde verilen zaman limiti içerisinde optimal değere ulaşamadığı için alt sınıra uzaklık optimal değere uzaklıktan çok daha fazla olabilir.

Tablo 5.19: Problem 2: İş Sayısına Göre Min_NEH Alt Sınır Yüzde Fark

		İşlem Süresi Taban Değeri Seviyesi	
		Düşük (10)	Yüksek (100)
20 iş	Ortalama	0.71	1.12
	Maksimum	6.63	10.85
50 iş	Ortalama	3.42	2.40
	Maksimum	28.32	21.53

Tablo 5.20'de ise CPLEX'in bulunduğu en iyi çözüme olan uzaklıklar farklı işlem süresi seviyelerine göre gruplandırılarak verilmiştir.

Tablo 5.20: Problem 2: İş Sayısına Göre Min_NEH En İyi CPLEX Yüzde Fark

Min_NEH-En iyi CPLEX % Fark		İşlem Süresi Taban Değeri Seviyesi	
		Düşük(10)	Yüksek (100)
20 iş	Ortalama	0.57	0.59
	Maksimum	1.79	2.58
50 iş	Ortalama	0.35	0.31
	Maksimum	2.03	2.16

CPLEX çözümlerine göre hata oranları Tablo 5.19 ve 5.20’de verilmiştir. CPLEX’in bulunduğu en iyi sonuçlara maksimum uzaklık %2.58’dir.

Tablo 5.21: Problem 2: 20 İş Min_NEH Sezgiseli Ve CPLEX Ortalama Çözüm Süreleri

	20 İş İşlem Süresi Taban Değeri Seviyesi	
	Düşük (10)	Yüksek (100)
Min_NEH	0.039	0.040
CPLEX	65.640	240.099

Tablo 5.22: Problem 2: 50 İş Min_NEH Sezgiseli Ve CPLEX Ortalama Çözüm Süreleri

	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (10)	Yüksek (100)
Min_NEH	0.040	0.041
CPLEX	378.434	541.046

Geliştirilmiş bu sezgiselin çözüm hızı zaman kısıtına uğramamış olan problemler için ortalama %300 kat daha hızlıdır. "Min_NEH" için çözüm süreleri Tablo 5.21 ve 5.22’de özetlenmiştir.

Üretilecek parça sayısının, işlem sürelerini taban değerlere göre büyüklüğünün ve işlem sürelerinin alındığı aralığın çözüm süresi ve yüzde farklar üzerindeki etkisinin istatistiksel olarak anlamlı olup olmadığını test etmek adına 2^4 faktoriyel tasarım oluşturulmuştur. Tasarımda, sabit ve esnek işlem sürelerinin aralıklarının etkisinin görülebilmesi için sabit ve esnek işlem sürelerinin hepsi ya düşük ya da yüksek aralık

seviyesine sahiptir. Tablo 5.23'de yer alan tasarımda işlem süreleri için belirlenen aralık seviyeleri tüm işlemler için sabitlenerek 2^4 faktöriyel tasarıma dönüştürülmüştür ve tabloda bulunan her kombinasyon için oluşturulan 5 koşturumun 5'i de analiz için kullanılmıştır. Faktör tasarımı son halinde parça sayısı, taban değeri ve aralık olarak 4 faktöre sahip olmuştur. Tasarımda yer alan 0 değerleri düşük seviyeyi 1 değerleri ise yüksek seviyeyi göstermektedir. Düşük ve yüksek seviyeler sırasıyla parça sayısı için 20 ve 50; taban değer için 10 ve 100; aralık için (taban değer/2) ve (taban değer ·4) olarak kullanılmıştır.

Tablo 5.23: Problem 2: Faktör Tasarımı

Parça Sayısı	İşlem Süresi	Taban Değeri	Aralık Seviyeleri				
			f^1	f^2	f^3	s^2	s^3
0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1
0	0	0	1	1	1	0	0
0	0	0	1	1	1	1	1
0	1	1	0	0	0	0	0
0	1	1	0	0	0	1	1
0	1	1	1	1	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0
1	0	0	0	0	0	1	1
1	0	0	1	1	1	0	0
1	0	0	1	1	1	1	1
1	1	1	0	0	0	0	0
1	1	1	0	0	0	1	1
1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1

Analiz Minitab 16.0 paket program ile gerçekleştirilmiştir. Bu programa ait çıktıların değerlendirilmesi sırasında tablolarda yer alan p-değerleri kullanılmaktadır. Yapılan testler sırasında kullanılan hipotezler şu şekildedir.

H_0 = Faktör istatistiksel olarak %95 güven düzeyinde anlamlı değildir.

$H_{alternatif}$ = Faktör istatistiksel olarak %95 güven düzeyinde anlamlıdır.

Tablolarda faktörlere karşılık gelen p-değerleri 0.05'ten küçükse boş (null) hipotez reddedilmekte ve alternatif hipotez doğrulanmaktadır. Aksi takdirde boş hipotez reddedilemeyerek, faktörün etkisiz olduğu anlaşılmaktadır. Analizin geri kalan kısmı

bu şekilde değerlendirilmiştir.

CPLEX çözüm süresine ait istatistiksel değerlerin Minitab çıktısı Tablo 5.24'te verilmiştir. Bu tabloya göre p-değeri 0.05'ten küçük olan parça sayısı ve işlem süreleri için verilen taban değer seviyelerinin anlamlı etkiye sahip olduğu gözlemlenmektedir. Parça sayısı arttıkça CPLEX çözüm süresi artış göstermektedir. Daha büyük işlem sürelerine sahip problemler de CPLEX çözüm süresinin artışına neden olmaktadır. Bununla birlikte, taban değerle birlikte esnek işlem süreleri aralık seviyelerinin etkileşimi de çözüm süresi üzerinde etkili bir faktördür. Analiz sonucu elde edilen katsayı değerlerine göre (Tablo 5.24) işlem sürelerinin taban değeri 100 iken ve esnek işlem süreleri 100 – 500 değerleri arasında değişim göstermekteyken CPLEX çözüm süresi azalmaktadır.

Tablo 5.24: Problem 2: CPLEX Çözüm Süresi Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		2,6	110,5	0,02	0,981
Parça Sayısı	727,1	363,5	156,3	2,33	0,023
Taban Değer	853,5	426,7	156,3	2,73	0,008
Aralık f	372,1	186,1	156,3	1,19	0,238
Aralık s	91,8	45,9	156,3	0,29	0,770
Parça Sayısı-Taban Değer	-179,1	-89,6	221,1	-0,41	0,687
Parça Sayısı-Aralık f	-483,5	-241,8	221,1	-1,09	0,278
Parça Sayısı-Aralık s	439,0	219,5	221,1	0,99	0,324
Taban Değer-Aralık f	-196,6	-98,3	221,1	-0,44	0,658
Taban Değer-Aralık s	-929,9	-465,0	221,1	-2,10	0,039
Aralık f-Aralık s	-458,2	-229,1	221,1	-1,04	0,304
Parça Sayısı-Taban Değer-Aralık f	488,2	244,1	312,7	0,78	0,438
Parça Sayısı-Taban Değer-Aralık s	799,1	399,6	312,7	1,28	0,206
Parça Sayısı-Aralık f-Aralık s	76,5	38,3	312,7	0,12	0,903
Taban Değer-Aralık f-Aralık s	290,6	145,3	312,7	0,46	0,644
Parça Sayısı-Taban Değer-Aralık f- Aralık s	-92,9	-46,5	442,2	-0,11	0,917

Benzer şekilde, CPLEX Alt Sınır ve CPLEX En İyi çözüm arasındaki yüzde fark üzerinde parça sayısının, aralık ve taban değer seviyelerinin anlamlı etkiye sahip olup olmadığı incelenmiştir. Tablo 5.25 bu analize dair Minitab çıktısı değerlerini göstermektedir. Tabloda yer alan p-değerlerine göre 0.05'ten küçük olan hiç bir faktör

bulunmamaktadır. Dolayısıyla Problem 2 için, %95 anlamlılık düzeyinde alt sınır ile CPLEX çözümü arasındaki boşluk işlem sürelerinin değişiminden, büyüklüğünden ve problemdeki parça sayısından etkilenmemektedir.

Tablo 5.25: Problem 2: CPLEX Alt Sınır CPLEX En İyi Yüzde Fark Faktörler

Term	Effect	Coef	SE Coef	T	P
Constant		-0,000	0,5508	-0,00	1,000
Parça Sayısı	0,000	0,000	0,7790	0,00	1,000
Taban Değer	0,020	0,010	0,7790	0,01	0,990
Aralık f	0,908	0,454	0,7790	0,58	0,562
Aralık s	-0,000	-0,000	0,7790	-0,00	1,000
Parça Sayısı-Taban Değer	-0,012	-0,006	1,1016	-0,01	0,996
Parça Sayısı-Aralık f	-0,908	-0,454	1,1016	-0,41	0,682
Parça Sayısı-Aralık s	1,155	0,578	1,1016	0,52	0,602
Taban Değer-Aralık f	0,454	0,227	1,1016	0,21	0,838
Taban Değer-Aralık s	-0,020	-0,010	1,1016	-0,01	0,993
Aralık f-Aralık s	-0,908	-0,454	1,1016	-0,41	0,682
Parça Sayısı-Taban Değer-Aralık f	3,804	1,902	1,5580	1,22	0,227
Parça Sayısı-Taban Değer-Aralık s	-0,504	-0,252	1,5580	-0,16	0,872
Parça Sayısı-Aralık f-Aralık s	-0,212	-0,106	1,5580	-0,07	0,946
Taban Değer-Aralık f-Aralık s	-0,454	-0,227	1,5580	-0,15	0,885
Parça Sayısı-Taban Değer-Aralık f-Aralık s	-2,418	-1,209	2,2033	-0,55	0,585

Çalışmada ele alınan faktörlerin sezgiselin alt sınıra olan uzaklığı üzerindeki etkisi incelendiğinde istatistiksel olarak anlamlı bir faktör bulunamamıştır. Tablo 5.26 analize ait istatistiksel verileri barındırmaktadır. Faktörlerin p-değerlerinin 0.05'ten büyük olduğu gözlemlenmiştir.

CPLEX ile edinilen çözümlerden biri alt sınır bir diğeri "CPLEX En İyi" olarak adlandırılan verilen süre limiti içerisinde ulaşılan çözümdür. CPLEX En İyi çözümün sezgisel değere olan uzaklığı üzerinde parça sayısı, aralık ve taban değer faktörlerinden hiç biri istatistiksel olarak anlamlı etkiye sahip değildir. Fakat oluşturulan deney tasarımına göre kararlılık katsayısı sadece %29 olduğu için bu modelin açıklayıcılığı çok yüksek değildir. Ele aldığımız faktörler dışında farklı faktörlerin bu duruma etkisi olabilir. Tablo 5.27'de faktörlerin hepsini p-değerlerinin 0.05'ten büyük olduğu görülmektedir. Bu durumda deney tasarımının başında kurulan

Tablo 5.26: Problem 2: CPLEX Alt Sınır Sezgisel Yüzde Fark Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		0,645	0,5714	1,13	0,263
Parça Sayısı	-0,673	-0,337	0,8080	-0,42	0,678
Taban Değer	-0,422	-0,211	0,8080	-0,26	0,795
Aralık f	1,298	0,649	0,8080	0,80	0,425
Aralık s	0,735	0,367	0,8080	0,45	0,651
Parça Sayısı-Taban Değer	0,142	0,071	1,1427	0,06	0,951
Parça Sayısı-Aralık f	-0,652	-0,326	1,1427	-0,29	0,776
Parça Sayısı-Aralık s	0,537	0,269	1,1427	0,24	0,815
Taban Değer-Aralık f	1,165	0,583	1,1427	0,51	0,612
Taban Değer-Aralık s	-0,144	-0,072	1,1427	-0,06	0,950
Aralık f-Aralık s	-2,109	-1,054	1,1427	-0,92	0,360
Parça Sayısı-Taban Değer-Aralık f	2,710	1,355	1,6161	0,84	0,405
Parça Sayısı-Taban Değer-Aralık s	-0,284	-0,142	1,6161	-0,09	0,930
Parça Sayısı-Aralık f-Aralık s	0,366	0,183	1,6161	0,11	0,910
Taban Değer-Aralık f-Aralık s	-0,254	-0,127	1,6161	-0,08	0,938
Parça Sayısı-Taban Değer-Aralık f-Aralık s	-2,245	-1,123	2,2855	-0,49	0,625

boş hipotez reddedilememektedir.

Tablo 5.27: Problem 2: CPLEX En İyi Sezgisel Yüzde Fark Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		0,6446	0,1638	3,94	0,000
Parça Sayısı	-0,6732	-0,3366	0,2316	-1,45	0,151
Taban Değer	-0,4426	-0,2213	0,2316	-0,96	0,343
Aralık f	0,3737	0,1868	0,2316	0,81	0,423
Aralık s	0,7349	0,3675	0,2316	1,59	0,118
Parça Sayısı-Taban Değer	0,1542	0,0771	0,3276	0,24	0,815
Parça Sayısı-Aralık f	0,2722	0,1361	0,3276	0,42	0,679
Parça Sayısı-Aralık s	-0,6217	-0,3109	0,3276	-0,95	0,346
Taban Değer-Aralık f	0,7230	0,3615	0,3276	1,10	0,274
Taban Değer-Aralık s	-0,1235	-0,0617	0,3276	-0,19	0,851
Aralık f-Aralık s	-1,1845	-0,5922	0,3276	-1,81	0,075
Parça Sayısı-Taban Değer-Aralık f	-1,1148	-0,5574	0,4633	-1,20	0,233
Parça Sayısı-Taban Değer-Aralık s	0,2224	0,1112	0,4633	0,24	0,811
Parça Sayısı-Aralık f-Aralık s	0,5651	0,2826	0,4633	0,61	0,544
Taban Değer-Aralık f-Aralık s	0,1878	0,0939	0,4633	0,20	0,840
Parça Sayısı-Taban Değer-Aralık f- Aralık s	0,1900	0,0950	0,6552	0,14	0,885

5.1.3 Sonuçlar

Esnek işlemlerle değişik tip parça akış tipi üretim sistemi problem için toplamda 11 sezgisel geliştirilmiştir. Sezgiseller 3 temel sezgiselin varyasyonlarından oluşmuştur. Oluşturulan sezgiseller 20 iş ve 50 iş için çözülmüştür. Değerlendirmelere göre NEH için sıralama kriteri olarak $f_i^1 + f_i^2 + f_i^3 + \min(s_i^2, s_i^3)$ ve $f_i^1 + f_i^2 + f_i^3 + \max(s_i^2, s_i^3)$ kullanıldığında daha iyi sonuçlar elde edildiği görülmüştür. Bu algoritmalar arasında en verimli sonucun NEH tabanlı algoritmalarından elde edildiği gözlenmiştir. "Min_NEH" ile CPLEX alt sınır sonuçlarına 20 iş için işlem süresi taban değeri düşük seviyede iken %0.71, yüksek seviyede iken %1.12; 50 iş için işlem süresi taban değeri düşük seviyede iken %3.42, yüksek seviyede iken %2.40 yaklaşmaktadır ki 900 saniye zaman limitiyle CPLEX çözücüsünün elde ettiği sonuçlarda (En iyi CPLEX) alt sınıra yaklaşık olarak aynı oranda uzaklık göstermektedir (Tablo 5.19,

5.20). Rastgele türetilmiş veriler için NEH algoritmasının; üçüncü makinedeki sabit işlem süresi büyük olduğu ve 3. makinedeki esnek işlem süresinin 2. Makinedeki esnek işlem süresinden biraz daha büyük olduğu işler için optimal ve CPLEX çözücüsüne göre 50 parçalık işlerde ortalama 6 saniye daha hızlı çözdüğü görülmüştür.

Parça sayısı, işlem süresi taban değeri ve aralık seviyelerinin çözüm süresi ve yüzde farklar üzerindeki etkisini ölçmek için 2^4 tam faktöriyel deney tasarımı uygulanmıştır. Sonuç olarak yüzde farklar üzerinde bu faktörlerin istatistiksel olarak anlamlı etkisi olmadığı gösterilmiştir. Bununla birlikte bazı faktörler tek başına CPLEX çözüm süresinin artışına neden olurken, faktörlerle olan etkileşimleri bu süreyi azaltmaktadır. CPLEX çözüm süresinin parça sayısı ve işlem sürelerine ait taban değer arttıkça arttığı gözlemlenmiştir. Fakat taban süresinin 100, esnek işlem sürelerinin de geniş aralıktan geldiği durumda bu sürenin azaldığı, çözüm süresine ait faktörlerin katsayılarını gösteren Tablo 5.24'den anlaşılmaktadır. Çalışmanın ilerleyen kısımlarında esnek makine ve esnek işlem sayısının artırıldığı durumlar incelenmektedir. Bu bölümde ulaşılan sonuçlar dolayısıyla NEH [42] algoritmasına "Min_NEH" ile verimli sonuçlar edinildiği, çalışmada yer alan problemler için bu sezgiselden yararlanmanın avantaj sağlayacağı düşünülmüştür.

5.2 Esnek İşlemin Üç Makine Üzerinde İşlenebildiği Durum

Bu problemde sistemdeki tüm makinelerin esnek makine olduğu varsayılmıştır. Problem 2'den farklı olarak esnek işlem 3 makinede de işlenebilmektedir. Probleme yönelik genel matematiksel model 3. bölümde yer almaktadır. Problemin NP-Zor sınıfına ait olması nedeniyle, geliştirilen matematiksel model ile optimal sonuca ulaşmak için çok büyük çözüm sürelerine ihtiyaç duyulmaktadır. Bu nedenle çözüm yöntemi olarak problemi verimli şekilde çözebilecek NEH [42] algoritmasını temel alan polinom zamanda çözüm veren 3 sezgisel geliştirilmiştir. Problem 3 için geliştirilen çözüm yöntemleri ilerleyen alt bölümde yer almaktadır.

5.2.1 Sezgisel Çözüm Yöntemleri

3. Bölümde Problem 3 olarak tanımlanan, bu problemde 3 özdeş olmayan makine ve her üç makinede de işlenebilecek tek bir esnek işlem yer almakta. Sabit ve esnek işlemler makineye ve parçaya göre farklılık göstermektedir. Tek esnek işlemin 2 – 3 (1 – 2) makineleri arasında olduğu problem için CDS, Gupta ve NEH algoritmalarını modifiye etmiştik. Bu algoritmalar arasında, geliştirilen NEH tabanlı algoritmaların diğer sezgisellere kıyasla çok üstün bir performans gösterdiği gözlemlenmiştir. Bu nedenle, bu bölümde geliştirilen sezgiseller NEH algoritmasına dayanmaktadır. Sistem girdilerinin 2. Probleme göre farklı olduğu göz önünde bulundurularak, parçaların ilk sıralanışını belirleyen NEH kriteri değiştirilmiştir. Bu bölümdeki problem için üç farklı NEH ölçütüne sahip sezgiseller önerilmiştir.

Tablo 5.28: Problem 3: Alternatif NEH Algoritmaları

Alternatifler	Sıralama Kriteri
NEH 1	$f_i^1 + f_i^2 + f_i^3 + s_i^1 + s_i^2 + s_i^3$
NEH 2	$f_i^1 + f_i^2 + f_i^3 + \min(s_i^1, s_i^2, s_i^3)$
NEH 3	$f_i^1 + f_i^2 + f_i^3 + \max(s_i^1, s_i^2, s_i^3)$

İlk modifiyede bir parçanın makineler üzerinde sahip olduğu tüm işlem sürelerinin toplamı sıralama ölçütü olarak kullanılmıştır. Algoritmaya dair sözde kod Algoritma 13 olarak verilmiştir.

Algoritma 14 ve 15’te sözde kodları verilen çözüm yöntemlerinde ise sıralama ölçütü parçaların sabit işlem zamanlarına, sahip oldukları esnek işlem sürelerinden en büyük veya en küçük olanının eklenmesiyle elde edilen değer olarak belirlenmiştir. Parçalar bu ölçüte göre sıralandıktan sonra, sıralamadaki ilk iki iş için olası tüm çizelgeler denenerak optimal çizelge sabitlenir. Bu sabitlenen çizelge üzerine her yeni gelen k . parça için üç olası esnek işlem ataması k olası sıralama ile değerlendirilerek en küçük yayılma zamanını veren çizelgenin sabitlenmesi ile devam edilir. Çizelgelenecek olan işler bitene kadar bu adımlar tekrarlanır ve olurlu bir çözüm elde edilir.

Algoritma 13, 14 ve 15’e yönelik performans analizi, oluşturulan matematiksel model ile elde edilen çözümlerle kıyaslanarak gerçekleştirilmiştir. Analiz için gerçekleştirilen

Algoritma 13 Problem 3: Modifiye NEH Algoritması 1

Girdi: $f_i^1, f_i^2, f_i^3, s_i^1, s_i^2, s_i^3$.

Çıktı: C_{max}

$i = 1; i \leq n; i++$ (bütün parçalar için)

$$NEH_i = f_i^1 + f_i^2 + f_i^3 + s_i^1 + s_i^2 + s_i^3$$

NEH_i değerlerini büyükten küçüğe sırala

bitiş döngü{i}

Sıralamaya göre en baştaki iş ve onu takip eden iş için olası tüm esnek işlem atamaları ve sıralamayı uygula

En küçük yayılma zamanını veren atamasını ve sıralamayı sabitle

Çizelgelenmeyen iş kalmayana kadar Tekrarla

Yeni eklenen iş için esnek işlem atamasını sırasıyla 1., 2. ve 3. makineye atayarak ve sıralamayı bir önceki adımda elde edilen sıralamayı bozmadan yayılma zamanını hesapla

Son tekrar

Algoritma 14 Problem 3: Modifiye NEH Algoritması 2

Girdi: $f_i^1, f_i^2, f_i^3, s_i^1, s_i^2, s_i^3$.

Çıktı: C_{max}

$i = 1; i \leq n; i++$ (bütün parçalar için)

$$NEH_i = f_i^1 + f_i^2 + f_i^3 + \min(s_i^1, s_i^2, s_i^3)$$

NEH_i değerlerini büyükten küçüğe sırala

bitiş döngü{i}

Sıralamaya göre en baştaki iş ve onu takip eden iş için olası tüm esnek işlem atamaları ve sıralamayı uygula

En küçük yayılma zamanını veren atamasını ve sıralamayı sabitle

Çizelgelenmeyen iş kalmayana kadar Tekrarla

Yeni eklenen iş için esnek işlem atamasını sırasıyla 1., 2. ve 3. makineye atayarak ve sıralamayı bir önceki adımda elde edilen sıralamayı bozmadan yayılma zamanını hesapla

Son tekrar

Algoritma 15 Problem 3: Modifiye NEH Algoritması 3

Girdi: $f_i^1, f_i^2, f_i^3, s_i^1, s_i^2, s_i^3$.

Çıktı: C_{max}

$i = 1; i \leq n; i++$ (bütün parçalar için)

$$NEH_i = f_i^1 + f_i^2 + f_i^3 + \max(s_i^1, s_i^2, s_i^3)$$

NEH_i değerlerini büyükten küçüğe sırala

bitiş döngü{i}

Sıralamaya göre en baştaki iş ve onu takip eden iş için olası tüm esnek işlem atamaları ve sıralamayı uygula

En küçük yayılma zamanını veren atamasını ve sıralamayı sabitle

Çizelgelenmeyen iş kalmayana kadar Tekrarla

Yeni eklenen iş için esnek işlem atamasını sırasıyla 1., 2. ve 3. makineye atayarak ve sıralamayı bir önceki adımda elde edilen sıralamayı bozmadan yayılma zamanını hesapla

Son tekrar

deneysel çalışma bir sonraki alt bölümde verilmektedir.

5.2.2 Deneysel Çalışma

Bu bölümde Problem 3 için geliştirilen sezgisellerin performansını analiz etmek amacıyla gerçekleştirilen deneysel çalışmadan ve elde edilen sonuçlardan bahsedilecektir. Oluşturulan deneysel çalışmada, f_i^1, f_i^2 ve f_i^3 değerleri için alt sınır ve işlem sürelerinin alabileceği değer aralığı anlamında iki seviye yer almaktadır. Alt sınırlar 10 ve 100 olarak belirlenmiştir. Aralıklar ise alt sınırı 10 olan işlem süreleri için $[5, 40]$ iken alt sınır değeri 100 olanlar için $[50, 400]$ olarak belirlenmiştir. Dolayısıyla işlem sürelerinin her biri için $(f_i^1, f_i^2, f_i^3$ ve $s)$ $[10, 15], [10, 50]$ ve $[100, 150], [100, 500]$ aralıklarında tek düze dağılım gösteren değerlerden rastgele elde edilmiştir. s_i^1, s_i^2 ve s_i^3 değerleri, üretilmiş s değerleri üzerinden $[0.8, 1]$ ve $[1, 1.2]$ yine tek düze dağılıma sahip (α, β, γ) katsayılarıyla bulunmuştur. Aralıkların ve taban değerlerin iki seviyeli olması nedeniyle $(2^7 \cdot 2 = 256)$ farklı durum vardır ve her durum için 5 tekrar yapılmıştır. Sonuç olarak 20 ve 50 iş için 1280 farklı problem seti üretilmiştir.

Taban değerler ve aralık seviyeleri için farklı parametrik değerler verilmesinin nedeni problemin kolaylaştığı veya zorlaştığı durumları tespit ederek bu faktörlerin etkilerini gösterebilmektir. Aynı şekilde 20 ve 50 iş için ayrı ayrı ölçümler yapılmasının nedeni

sezgisel deęer ile CPLEX ile elde edilen deęerler arasındaki fark üzerinde, CPLEX çözüm süresi üzerinde parça sayısının etkisi olup olmadığını anlayabilmektir. Üretilen problemler için optimal çözümler, karma tamsayı programlama ile modellenmiş ve GAMS CPLEX 12.0 çözücüsü ile çözdürülmüştür. Her problemin optimal çözüme ulaşabilmesi için 900 saniyelik bir zaman limiti konulmuştur ve bazı problemler bu süre içerisinde optimal çözüme erişememektedir. Problemin güçlü NP-Zor olması nedeniyle çözüm süresi, problem sayısı ve işlem süreleri arttıkça uzamaktadır. GAMS CPLEX 12.0 ile çözülmüş problemler ile ilgili özet Tablo 5.29’da yer almaktadır.

Tablo 5.29: Problem 3: CPLEX Çözüm Durumları

	Optimal Çözüm			Zaman Limiti		
	Düşük (10)	Yüksek (100)	Toplam	Düşük (10)	Yüksek (100)	Toplam
20 İş (1280)	541	373	914	99	267	366
50 İş (1280)	340	336	376	300	304	604

20 parçalı problemler için işlem süresi alt sınırı düşük seviyede (10) alınarak üretilen durum için CPLEX çözücüsü 900 saniyelik limitle 640 problemde 541’ini optimal olarak bulabilmiştir fakat alt sınırın 100 olarak belirlendiği durumlarda ancak 340 problemde optimal sonuç elde edilmiştir. Parça sayısı 50’ye çıktığında ise zaman limitine takılan problem sayısını 366’dan 604’e çıktığı gözlemlenmektedir. Dolayısıyla Problem 3 için polinom zamanda çözüm üretebilen iyi performanslı bir sezgisel ihtiyaç duyulmaktadır. Performans değerlendirmesi CPLEX ile bulunmuş yayılma zamanı alt sınır değerleri ve önerilen sezgisellerin yayılma zamanı değerleri arasındaki yüzde fark ile yapılmıştır. Yüzde fark, Bölüm 5.1’de verildiği gibi hesaplanmaktadır.

Tablo 5.30: Problem 3: 20 İş İçin Min_NEH Alt Sınır Yüzde Fark

	İşlem Süresi Taban Deęeri Seviyesi	
	Düşük (10)	Yüksek (100)
NEH 1	3.55	7.12
NEH 2	2.95	6.43
NEH 3	7.78	11.23
Min_NEH	2.43	5.83
CPLEX	0.88	4.22

Tablo 5.31: Problem 3: 50 İş İçin Min_NEH Alt Sınır Yüzde Fark

	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (10)	Yüksek (100)
NEH 1	6.03	6.32
NEH 2	5.51	5.66
NEH 3	12.09	12.12
Min_NEH	5.01	5.21
CPLEX	4.47	5.01

Tablo 5.30 ve 5.31’de sezgisellerden gelen sonuçlar her tekrardan gelen sonuçların ortalama yüzde fark uzaklığı ile değerlendirilmektedir. Tablo 5.30 ve 5.31’e göre işlem sürelerindeki artış optimal çözüme olan uzaklığı arttırmaktadır. Bu problem için ortalamalara bakıldığında CPLEX çözücüsünün ulaştığı çözümlere en yakın çözümler NEH ölçütü olarak $f^1 + f^2 + f^3 + \min(s^1, s^2, s^3)$ kullanıldığında elde edilmektedir. Bu problem için de elde edilen sonuçların verimini arttırmak için "Min_NEH" kullanılmıştır. 3 sezgiselden de elde edilen en küçük sonuç kullanılmaktadır. "Min_NEH" ile ulaşılan sonuçların işlem süresi seviyelerine ve iş sayısına göre CPLEX alt sınırından farkı Tablo 5.32’de verilmektedir. Bu tabloda sezgisel ile ulaşılan değerler ile CPLEX çözücüsü ile elde edilmiş alt sınır arasındaki maksimum uzaklığın %58’e ulaştığı gözlenmektedir. Verilen zaman limiti içerisinde alt sınır her zaman için optimal sonuca ulaşamamaktadır. Bu durumda CPLEX’in aynı zamanda ulaştığı değerle birlikte bir değerlendirme yapılması anlamlı olmaktadır. Sezgisel ile CPLEX ile elde edilen en iyi çözüm kıyaslandığında alt sınır ile uzaklığın çok yüksek olduğu durumlarda, CPLEX en iyi ve sezgisel değeri arasındaki yüzde farkın %1’e düştüğü gözlemlenmektedir. Aynı zamanda alt sınıra olan ortalama uzaklığın 50 iş için %5 civarında olması sezgiselin verimli sonuçlara ulaşabildiğini göstermektedir.

Tablo 5.33’e göre CPLEX’in eriştiği sonuçlara çok yakın sonuçlar edinildiği gözlenmektedir. Bu sonuçlara göre yüzde uzaklık maksimum %10.36’dır. İş sayısı arttıkça optimal çözüme ulaşma süresinin artması nedeniyle 20 işten 50 işe çıkıldığında En iyi CPLEX değerlerine olan uzaklık azalmaktadır.

Çalışmada sezgisellerle optimal çözüme ulaşılan problem sayısı, sezgisel ile ulaşılan

Tablo 5.32: Problem 3: İş Sayısına göre Min_NEH Alt Sınır Yüzde Fark

		İşlem Süresi Taban Değeri Seviyesi	
		Düşük (10)	Yüksek (100)
20 iş	Ortalama	2.43	5.83
	Maksimum	53.89	59.84
50 iş	Ortalama	5.01	5.21
	Maksimum	78.08	31.58

Tablo 5.33: Problem 3: İş Sayısına göre Min_NEH En İyi CPLEX Yüzde Fark

Min_NEH- En İyi CPLEX % Fark		İşlem Süresi Taban Değeri Seviyesi	
		Düşük (10)	Yüksek (100)
20 iş	Ortalama	1.54	1.55
	Maksimum	8.85	10.36
50 iş	Ortalama	0.54	0.42
	Maksimum	7.77	6.53

çözüm değerinin CPLEX alt sınır değeri ile aynı değere sahip olup olmadığı kontrol edilerek hesaplanmıştır. Tablo 5.34'e göre 20 iş için 1280 problemde 312'sinde, 50 iş için 1280 problemde 252'sinde optimal çözüme ulaşılmıştır.

Tablo 5.34: Problem 3: İş Sayısına göre Min_NEH Optimal Çözülen Problem Sayısı

	İşlem Süresi Taban Değeri Seviyesi		
	Düşük (10)	Yüksek (100)	Toplam
20 İş	139 (541)	173 (373)	312 (914)
50 İş	141 (340)	111 (336)	252 (676)

Tablo 5.35: Problem 3: 20 İş için Sezgisel ve CPLEX Çözüm Süreleri

	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (sn)	Yüksek (sn)
NEH 1	0.02	0.02
NEH 2	0.02	0.02
NEH 3	0.02	0.02
Min_NEH	0.06	0.06
CPLEX	169.09	387.37

Sezgisellerin performans analizi yapılırken dikkate alınması gereken bir diğer nokta, çözüm süreleridir. Tablo 5.35 ve 5.36'de gösterilmektedir. Bu tablolara göre CPLEX, çok büyük çözüm sürelerine ihtiyaç duyarken, geliştirdiğimiz sezgiselin çözüm süresi 1 saniyeyi bile bulmamaktadır.

Ortalama ve maksimum değerler dışında sahip olunan tüm veriler üzerinde bir değerlendirme yapabilmek için deney tasarımı tekniklerinden yararlanılmıştır. Problem 3 için parça sayısının, işlem sürelerini taban değerlerinin ve işlem sürelerinin alındığı aralığın CPLEX çözüm süresi ve sezgisel çözüm ile CPLEX çözümleri arasındaki yüzde farklar üzerindeki etkinin istatistiksel olarak değerlendirilebilmesi için 2^4 faktoriyel tasarım oluşturulmuştur. Tasarımda, sabit ve esnek işlem sürelerinin aralıklarının etkisinin görülebilmesi için sabit ve esnek işlem sürelerinin hepsi ya düşük ya da yüksek aralık seviyesine sahiptir. Tasarım Tablo 5.37'de verilmektedir. Tablo 5.37'de yer alan tasarımda işlem süreleri için belirlenen aralık seviyeleri tüm

Tablo 5.36: Problem 3: 50 İş için Sezgisel ve CPLEX Çözüm Süreleri

	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (sn)	Yüksek (sn)
NEH 1	0.26	0.25
NEH 2	0.24	0.24
NEH 3	0.21	0.20
Min_NEH	0.71	0.69
CPLEX	437.57	445.62

işlemler için sabitlenerek 2^4 faktöriyel tasarıma dönüştürülmüştür ve tabloda bulunan her kombinasyon için 5 tekrar yapılmıştır. Faktör tasarımı son halinde parça sayısı, taban değeri ve aralık olarak 3 faktöre sahip olmuştur. Tasarımda yer alan 0 değerleri düşük seviyeyi 1 değerleri ise yüksek seviyeyi göstermektedir. Düşük ve yüksek seviyeler sırasıyla parça sayısı için 20 ve 50; taban değer için 10 ve 100; aralık için (taban değer/2) ve (taban değer ·4) olarak kullanılmıştır.

Tablo 5.37: Problem 3: Faktör Tasarımı

Parça Sayısı	İşlem Süresi Taban Değeri	Aralık Seviyeleri							
		f^1	f^2	f^3	s	s^1	s^2	s^3	
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1
0	0	1	1	1	0	0	0	0	0
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	0	0	0	0
0	1	0	0	0	1	1	1	1	1
0	1	1	1	1	0	0	0	0	0
0	1	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	1	1	1	1	1
1	0	1	1	1	0	0	0	0	0
1	0	1	1	1	1	1	1	1	1
1	1	0	0	0	0	0	0	0	0
1	1	0	0	0	1	1	1	1	1
1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1

Analiz Minitab 16.0 paket program ile gerçekleştirilmiştir. Bu programa ait çıktıların değerlendirilmesi sırasında tablolarda yer alan p-değerleri kullanılmaktadır. Yapılan

testler sırasında kullanılan hipotezler Problem 2’de kullanıldığı gibidir.

H_0 = Faktör istatistiksel olarak %95 güven düzeyinde anlamlı değildir.

$H_{alternatif}$ = Faktör istatistiksel olarak %95 güven düzeyinde anlamlıdır.

Öncelikle, CPLEX çözüm süresini etkileyen faktörler incelendiğinde Tablo 5.38’e göre temel faktörler parça sayısı, işlem süresi taban değeri ve esnek işlem sürelerinin aralığıdır. Anlamlı etkiye sahip temel faktörlerin tek başlarına CPLEX çözüm süresini uzattığı katsayıların pozitif olmasından anlaşılmaktadır (Tablo 5.38). Diğer yandan Tablo 5.38’de p değerleri 0.05’ten küçük olan parça sayısı ve taban değer ; parça sayısı ve esnek işlem süresi aralığının etkileşimleri ile oluşan faktörlerin ise çözüm süresini azalttığı yine bu tablodaki faktörlerin katsayısını veren "Coeff" sütunundan görülmektedir.

Tablo 5.38: Problem 3: CPLEX Çözüm Süresi Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		730,7	21,80	33,52	0,000
Parça Sayısı	338,8	169,4	21,80	7,77	0,000
Taban Değer	303,7	151,8	21,80	6,97	0,000
Aralık f	41,3	20,6	21,80	0,95	0,348
Aralık s	92,6	46,3	21,80	2,12	0,037
Parça Sayısı-Taban Değer	-303,7	-151,8	21,80	-6,97	0,000
Parça Sayısı-Aralık f	-41,3	-20,6	21,80	-0,95	0,348
Parça Sayısı-Aralık s	-92,6	-46,3	21,80	-2,12	0,038
Taban Değer-Aralık f	-76,4	-38,2	21,80	-1,75	0,084
Taban Değer-Aralık s	-57,5	-28,7	21,80	-1,32	0,192
Aralık f-Aralık s	32,5	16,3	21,80	0,75	0,459
Parça Sayısı-Taban Değer-Aralık f	76,4	38,2	21,80	1,75	0,085
Parça Sayısı-Taban Değer-Aralık s	57,5	28,7	21,80	1,32	0,192
Parça Sayısı-Aralık f-Aralık s	-32,5	-16,2	21,80	-0,74	0,459
Taban Değer-Aralık f-Aralık s	2,6	1,3	21,80	0,06	0,952
Parça Sayısı-Taban Değer-Aralık f-Aralık s	-2,7	-1,3	21,80	-0,06	0,951

Problem 3 için CPLEX Alt Sınır ve CPLEX En İyi çözüm arasındaki yüzde boşluk üzerinde taban değer seviyelerinin ve esnek işlem aralık seviyesinin etkilerinin

olduğu gözlemlenmiştir. Bu analizin Minitab çıktısına ait değerler Tablo 5.39’da gösterilmektedir. Bu tabloda anlamlı bulunan bu faktörlerin aldığı katsayı değerlerine göre bu yüzde boşluk taban değer 10’dan 100’e çıkarıldığında artmaktadır ve benzer şekilde esnek işlem sürelerinin üretildiği aralık daha geniş olduğunda da bu boşluk artmaktadır. %95 anlamlılık düzeyinde sabit işlemlerin aralık seviyesinin etkisi gözlemlenmemektedir.

Tablo 5.39: Problem 3: CPLEX Alt Sınır CPLEX En İyi Yüzde Fark Faktörler

Term	Effect	Coef	SE Coef	T	P
CConstant		10,067	1,373	7,33	0,000
Parça Sayısı	0,055	0,028	1,373	0,02	0,984
Taban Değer	6,273	3,136	1,373	2,29	0,026
Aralık f	-1,700	-0,850	1,373	-0,62	0,538
Aralık s	7,933	3,967	1,373	2,89	0,005
Parça Sayısı-Taban Değer	-4,539	-2,270	1,373	-1,65	0,103
Parça Sayısı-Aralık f	1,557	0,778	1,373	0,57	0,573
Parça Sayısı-Aralık s	-3,416	-1,708	1,373	-1,24	0,218
Taban Değer-Aralık f	-0,364	-0,182	1,373	-0,13	0,895
Taban Değer-Aralık s	1,837	0,918	1,373	0,67	0,506
Aralık f-Aralık s	1,933	0,966	1,373	0,70	0,484
Parça Sayısı-Taban Değer-Aralık f	2,063	1,032	1,373	0,75	0,455
Parça Sayısı-Taban Değer-Aralık s	-0,273	-0,137	1,373	-0,10	0,921
Parça Sayısı-Aralık f-Aralık s	0,252	0,126	1,373	0,09	0,927
Taban Değer-Aralık f-Aralık s	0,676	0,338	1,373	0,25	0,806
Parça Sayısı-Taban Değer-Aralık f-Aralık s	-1,861	-0,930	1,373	-0,68	0,500

Çalışmada sezgiselin performansını test ederken en çok kullanılan değerlerden biri CPLEX alt sınırı ile arasındaki yüzde farktır. Problem 3’te bu artışı etkileyen temel faktör esnek işlem süresinin aralığı ve taban değer olmuştur. Bu problem de esnek işlemin 3 makineye de atanabilir olması bir çok değişkende esnek işlem süresinin aralık seviyesinin etkili bir faktör olarak çıkmasına neden olarak yorumlanabilir.

Alt sınırla birlikte CPLEX çözücüsünden elde edilen, verilen süre içerisinde ulaşılan CPLEX çözümü ile sezgisel arasındaki yüzde fark da önemli bir değişkendir. Bu değişkeni etkileyen faktörler ise Problem 3 için üretilecek parça sayısı ve işlem sürelerinin taban değeri olmuştur. Tablo 5.41 katsayı değerlerine göre parça sayısı

Tablo 5.40: Problem 3: CPLEX Alt Sınır Sezgisel Yüzde Fark Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		10,708	1,376	7,78	0,000
Parça Sayısı	-1,947	-0,973	1,376	-0,71	0,482
Taban Değer	7,002	3,501	1,376	2,54	0,013
Aralık f	-0,753	-0,376	1,376	-0,27	0,785
Aralık s	7,870	3,935	1,376	2,86	0,006
Parça Sayısı-Taban Değer	-3,986	-1,993	1,376	-1,45	0,153
Parça Sayısı-Aralık f	1,516	0,758	1,376	0,55	0,584
Parça Sayısı-Aralık s	-3,774	-1,887	1,376	-1,37	0,175
Taban Değer-Aralık f	-0,407	-0,204	1,376	-0,15	0,883
Taban Değer-Aralık s	1,952	0,976	1,376	0,71	0,481
Aralık f-Aralık s	1,434	0,717	1,376	0,52	0,604
Parça Sayısı-Taban Değer-Aralık f	1,894	0,947	1,376	0,69	0,494
Parça Sayısı-Taban Değer-Aralık s	-0,279	-0,140	1,376	-0,10	0,920
Parça Sayısı-Aralık f-Aralık s	-0,369	-0,185	1,376	-0,13	0,894
Taban Değer-Aralık f-Aralık s	1,382	0,691	1,376	0,50	0,617
Parça Sayısı-Taban Değer-Aralık f-Aralık s	-1,386	-0,693	1,376	-0,50	0,616

arttıkça yüzde fark azalmaktadır, diğer yandan sabit işlem süreleri geniş aralıktan seçildikçe sezgisel ile CPLEX En iyi çözüm değerleri uzaklaşmaktadır.

Tablo 5.41: Problem 3: CPLEX En İyi Sezgisel Yüzde Fark Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		0,6074	0,1931	3,15	0,003
Parça Sayısı	-1,8165	-0,9082	0,1931	-4,70	0,000
Taban Değer	0,5905	0,2952	0,1931	1,53	0,131
Aralık f	0,8578	0,4289	0,1931	2,22	0,030
Aralık s	-0,1583	-0,0791	0,1931	-0,41	0,683
Parça Sayısı-Taban Değer	0,5532	0,2766	0,1931	1,43	0,157
Parça Sayısı-Aralık f	-0,0814	-0,0407	0,1931	-0,21	0,834
Parça Sayısı-Aralık s	-0,2893	-0,1446	0,1931	-0,75	0,457
Taban Değer-Aralık f	-0,0326	-0,0163	0,1931	-0,08	0,933
Taban Değer-Aralık s	0,1312	0,0656	0,1931	0,34	0,735
Aralık f-Aralık s	-0,4954	-0,2477	0,1931	-1,28	0,204
Parça Sayısı-Taban Değer-Aralık f	-0,1167	-0,0584	0,1931	-0,30	0,763
Parça Sayısı-Taban Değer-Aralık s	0,0327	0,0164	0,1931	0,08	0,933
Parça Sayısı-Aralık f-Aralık s	-0,5146	-0,2573	0,1931	-1,33	0,187
Taban Değer-Aralık f-Aralık s	0,5960	0,2980	0,1931	1,54	0,128
Parça Sayısı-Taban Değer-Aralık f- Aralık s	0,4351	0,2176	0,1931	1,13	0,264

5.2.3 Sonuçlar

Bu bölümde tek esnek işlemlili ve tüm makinelerin esnek olduğu 3 makineli sistemde yayılma zamanını en küçüklemeyi hedefleyen çözüm yöntemlerinin performans analizi yapılmıştır. 3 farklı sezgisel önerilmiştir. Sezgiseller oluşturulurken, NEH sezgiseli temel alınmıştır. Sezgiselin performansı toplamda 2560 problem üzerinde incelenmiştir. CPLEX çözücüsünün ulaştığı sonuçlar ve sezgiseller arasındaki en küçük değer alınarak elde edilen "Min_NEH" değeri arasındaki CPLEX alt sınır değerine uzaklıklar birbirine çok yakındır. İş sayısı ve parçaların makineler üzerindeki işlem süresi arttıkça problemin karmaşıklığının arttığı gözlemlenmiştir. Önerilen sezgiseller zaman ve yüzde fark açısından incelendiğinde verimli sonuçlara ulaştırmaktadır. Problem 3 için uygulanan deney tasarımı sonucu parça sayısı, taban

değer ve esnek işlem sürelerinin aralık seviyeleri CPLEX çözüm süresi üzerinde oldukça etkili olduğu sonucuna varılmıştır. Faktörler arasında yüzde farklar ve çözüm süreleri üzerinde işlem sürelerinin taban değer seviyelerindense, esnek işlem sürelerinin sahip olduğu aralık seviyeleri istatistiksel anlamda daha etkilidir.

6. ÜÇ MAKİNE İKİ ESNEK İŞLEME SAHİP SİSTEMLER

Değişik tip parça üretimi sağlayan 3-makinelik akış tipi sistemlerde ele alınan bir diğer problem ise iki esnek işlem olduğu durumdur. İki esnek işlemden ilki 1 veya 2'inci makinede ikincisi ise 2 veya 3'üncü makinede işlenebilmektedir. Parçaların ve makinelerin özdeş olmaması işlem sürelerinin hem makinelere hem de parçalara göre değişim göstermesine neden olmaktadır. Her parçanın işlem görmesi gereken beş işlem bulunmaktadır. İşlemlerden üç tanesi sabit işlemlerdir ve iki tane de esnek işlem bulunmaktadır.

Bu bölümde ele alınan problem (Problem 4) için geçerli olan matematiksel model, Bölüm 3'te verilen genel matematiksel modelden farklı kısıtlara ve notasyona ihtiyaç duymaktadır. Hem 1. hem de 2. esnek işlem 2. makinede işlenebildiği için, makinelerde bir iş tamamlanmadan yeni bir işin başlayamayacağını anlatan kısıtlar 3.2 ve 3.5 değişmelidir. Aynı zamanda esnek işlemlerin esnek makinelerden birine atanmasına yönelik 3.10 numaralı kısıttan iki esnek işlem için de bulunması gerekmektedir. Bunlara bağlı olarak doğrusal olmayan kısıtları doğrusallaştırmak için kullandığımız 3.8 ve 3.9 numaralı kısıtlar da değişikliğe ihtiyaç duymaktadır. Problem 4'e ait kısıtlar bu bölümde yer almaktadır.

Min C_{max}

Öyle ki

$$C_{max} \geq T_n^3 + \sum_{i=1}^n f_i^3 . x_{in} + s_i^{23} . b_{i,n}^{23} \quad \forall i \in N \quad (6.1)$$

$$T_j^2 \geq T_j^1 + \sum_{i=1}^n f_i^1 . x_{ij} + s_i^{11} . b_{i,j}^{11} \quad \forall j \in N \quad (6.2)$$

$$T_j^3 \geq T_j^2 + \sum_{i=1}^n f_i^2 . x_{ij} + s_i^{12} . b_{i,j}^{12} + s_i^{22} . b_{i,j}^{22} \quad \forall j \in N \quad (6.3)$$

$$T_{j+1}^1 \geq T_j^1 + \sum_{i=1}^n f_i^1 . x_{ij} + s_i^{11} . b_{i,j}^{11} \quad \forall j \in N - \{n\} \quad (6.4)$$

$$T_{j+1}^2 \geq T_j^2 + \sum_{i=1}^n f_i^2 . x_{ij} + s_i^{12} . b_{i,j}^{12} + s_i^{22} . b_{i,j}^{22} \quad \forall j \in N - \{n\} \quad (6.5)$$

$$T_{j+1}^3 \geq T_j^3 + \sum_{i=1}^n f_i^3 . x_{ij} + s_i^{23} . b_{i,j}^{23} \quad \forall j \in N - \{n\} \quad (6.6)$$

$$\sum_{i=1}^n x_{ij} = 1 \quad \forall j \in N \quad (6.7)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i \in N \quad (6.8)$$

$$y_j^{1k} \geq \sum_{i=1}^n b_{ij}^{1k} \quad \forall i \in N, \forall j \in N, \forall k = \{1, 2\} \quad (6.9)$$

$$y_j^{2k} \geq \sum_{i=1}^n b_{ij}^{2k} \quad \forall i \in N, \forall j \in N, \forall k = \{2, 3\} \quad (6.10)$$

$$y_j^{11} + y_j^{12} = 1 \quad \forall j \in N \quad (6.11)$$

$$y_j^{22} + y_j^{23} = 1 \quad \forall j \in N \quad (6.12)$$

$$b_{ij}^{1k} + 1 \geq x_{ij} + y_j^{1k} \quad \forall i \in N, \forall j \in N, \forall k = \{1, 2\} \quad (6.13)$$

$$b_{ij}^{2k} + 1 \geq x_{ij} + y_j^{2k} \quad \forall i, \forall j, \forall k = \{2, 3\} \quad (6.14)$$

$$b_{ij}^{lk}, y_j^{lk}, x_{ij} \in \{0, 1\} \quad \forall i \in N, j \in N, k \in M, l = \{1, 2\} \quad (6.15)$$

$$T_j^k \geq 0, C_{max} \geq 0 \quad \forall i \in N, j \in N, k \in M \quad (6.16)$$

Yayılma zamanı (makespan) sonuncu (n .) parçanın 3. makedeki tamamlanma zamanına eşittir. Son parçanın ikinci esnek işlemleri 3. makineye atanmışsa, yayılma

zamanı n . parçanın makine 3'te başlama zamanı ve makine 3'teki sabit ve esnek işlem süresinin toplamına eşittir. Dolayısıyla, parça sıralaması değişik tip parça problemlerinin diğerlerinde olduğu gibi bu problem için de önem arz etmektedir. Problemden çıktı maksimizasyonunu sağlayan, esnek işlem atamasına ve parça sıralamasına karar verilmesi önemlidir. Bu problem Problem 2 ve 3'e benzer niteliklere sahiptir. Bu problemde fazladan bir esnek işlem daha bulunmaktadır. Dolayısıyla Problem 4 güçlü NP-Zor'dur. Bu problem için geliştirilmiş matematiksel model ile optimal sonuçlar elde etmek için çok büyük çözüm sürelerine ihtiyaç duyulmaktadır. Problemden esnek işlemlerin atamalarına ve parçaların sıralamalarına karar verebilmek için NEH [42] tabanlı, kaliteli sonuçlar veren algoritmalar geliştirilmiştir. Bir sonraki bölümde bu algoritmalarından bahsedilmektedir.

6.1 Sezgisel Çözüm Yöntemleri

Problem 4 diğer problemlerden farklı olarak tek esnek işlem yerine 2 esnek işleme sahiptir. Bu durum esnek işlem süresine ait notasyonda farklılık yaratmaktadır. $s_i^{kl} = i$ işinin l 'inci esnek işleminin k makinesindeki işlem süresini göstermektedir. Problemin çözümü için geliştirilmiş sezgiseller Bölüm 5'de anlatılan sezgisellerle benzerlik göstermektedir. Fakat bu kez parçaların en baştaki sıralanışını belirleyen NEH ölçütü daha farklıdır. Bu bölümdeki problem için beş farklı NEH ölçütüne sahip sezgisel önerilmiştir.

Tablo 6.1: Problem 4: Alternatif NEH Algoritmaları

Alternatifler	Sıralama Kriteri
NEH 1	$f_i^1 + f_i^2 + f_i^3 + s_i^{11} + s_i^{12} + s_i^{22} + s_i^{23}$
NEH 2	$f_i^1 + f_i^2 + f_i^3 + \min(s_i^{11}, s_i^{12}) + \min(s_i^{22}, s_i^{23})$
NEH 3	$f_i^1 + f_i^2 + f_i^3 + \max(s_i^{11}, s_i^{12}) + \max(s_i^{22}, s_i^{23})$
NEH 4	$f_i^1 + f_i^2 + f_i^3 + \min(s_i^{11}, s_i^{12}) + \max(s_i^{22}, s_i^{23})$
NEH 5	$f_i^1 + f_i^2 + f_i^3 + \max(s_i^{11}, s_i^{12}) + \min(s_i^{22}, s_i^{23})$

Sözde kodları Algoritma 16 – 20 olarak verilen sezgisellerde amaç yayılma zamanını küçükleyecek esnek işlem atamasını ve parça sıralamasını belirlemeye yardımcı

olmaktır. Tablo 6.1’de verilen kriterlere göre sıralama yaptıktan sonra ilk iki iş için olası tüm atamalar ve sıralamalar gerçekleştirilerek en küçük yayılma zamanına sahip çizelge sabitlenir. Çizelgelenen bu işlerin kendi içlerindeki sıralaması ve sahip oldukları esnek işlem atamaları değiştirilmeden yeni gelen k . parça için k olası sıralamadan ve dört olası esnek işlem atamasından en küçük yayılma zamanını sağlayan çizelge sabitlenerek tüm parçalar çizelgelenir. Olası esnek işlem atamaları şu şekilde listelenebilir;

1. **Alternatif İşlem Ataması 1:** İlk esnek işlem 1. makineye atanırken, ikinci esnek işlem 2. makineye atanır. Atama sonucu oluşan işlem süreleri şu şekilde

$$p_i^1 = f_i^1 + s_i^{11}$$

$$p_i^2 = f_i^2 + s_i^{22}$$

$$p_i^3 = f_i^3$$

2. **Alternatif İşlem Ataması 2:** İlk esnek işlem 1. makineye atanırken, ikinci esnek işlem 3. makineye atanır. Atama sonucu oluşan işlem süreleri şu şekilde

$$p_i^1 = f_i^1 + s_i^{11}$$

$$p_i^2 = f_i^2$$

$$p_i^3 = f_i^3 + s_i^{23}$$

3. **Alternatif İşlem Ataması 3:** İlk esnek işlem 2. makineye atanırken, ikinci esnek işlem de 2. makineye atanır. Atama sonucu oluşan işlem süreleri şu şekilde

$$p_i^1 = f_i^1$$

$$p_i^2 = f_i^2 + s_i^{12} + s_i^{22}$$

$$p_i^3 = f_i^3$$

4. **Alternatif İşlem Ataması 4:** İlk esnek işlem 2. makineye atanırken, ikinci esnek işlem 3. makineye atanır. Atama sonucu oluşan işlem süreleri şu şekilde

$$p_i^1 = f_i^1$$

$$p_i^2 = f_i^2 + s_i^{12}$$

$$p_i^3 = f_i^3 + s_i^{23}$$

İlk modifiyede (NEH 1) sıralama ölçütü olarak parçaların makineler üzerindeki tüm sabit ve esnek işlemlerinin toplamı alınmıştır. NEH 2’de iki esnek işlem için de işlem süresinin minimum olduğu makinelerdeki işlem süreleri ve sabit işlem sürelerinin toplamından yararlanılmıştır. Benzer şekilde 3. alternatifte de esnek işlemler için minimum olan işlem süresi yerine maksimum süreler üzerinden gidilmiştir. NEH 4 ve NEH 5 için de esnek işlemlerden birinde o esnek işlemin sahip olduğu en büyük işlem süresi alınırken diğer esnek işlem için en küçük olanı alınarak sıralama yapılmaktadır. Bu algoritmaların performansını analiz eden deneysel çalışma bir sonraki alt bölümde yer almaktadır.

Algoritma 16 Problem 4: Modifiye NEH Algoritması 1

Girdi: $f_i^1, f_i^2, f_i^3, s_i^{11}, s_i^{12}, s_i^{22}, s_i^{23}$.

Çıktı: C_{max}

- 1: $i = 1$; $i \leq n$; $i++$ (bütün parçalar için)
 - 2: $NEH_i = f_i^1 + f_i^2 + f_i^3 + s_i^{11} + s_i^{12} + s_i^{22} + s_i^{23}$
 - 3: NEH_i değerlerini büyükten küçüğe sırala
 - 4: **bitiş döngü**{i}
 - 5: Sıralamaya göre öncelikle en baştaki iki iş için olası esnek işlem atamalarını ve sıralamayı dene
 - 6: En küçük yayılma zamanını veren sıralamayı ve atamayı sabitle
 - 7: Çizelgelenmeyen iş kalmayana kadar Tekrarla
 - 8: Yeni eklenen iş için esnek işlemlerin olası 4 atamasını ve bir önceki adımda elde edilen sıralamayı bozmayan olası sıralamalar ile yayılma zamanını hesapla.
 - 9: **Son** tekrar
-

6.2 Deneysel Çalışma

Oluşturulan deneysel çalışmada, f_i^1, f_i^2 ve f_i^3 değerleri için alt sınır ve işlem sürelerinin alabileceği değer aralığı anlamında iki seviye yer almaktadır. Alt sınırlar 10 ve 100 olarak belirlenmiştir. Aralıklar ise alt sınırı 10 olan işlem süreleri için [5, 40] iken alt sınır değeri 100 olanlar için [50, 400] olarak belirlenmiştir. Dolayısıyla işlem süreleri [10, 15], [10, 50] ve [100, 150], [100, 500] aralıklarında tek düze dağılım gösteren değerlerden rastgele elde edilmiştir. s_i^{12}, s_i^{11} e ve s_i^{23}, s_i^{22} ye bağlı olarak, [0.8, 1] ve

Algoritma 17 Problem 4: Modifiye NEH Algoritması 2

Girdi: $f_i^1, f_i^2, f_i^3, s_i^{11}, s_i^{12}, s_i^{22}, s_i^{23}$.

Çıktı: C_{max}, NEH_i

- 1: $i = 1; i \leq n; i ++$ (bütün parçalar için)
 - 2: $NEH_i = f_i^1 + f_i^2 + f_i^3 + \min(s_i^{11}, s_i^{12}) + \min(s_i^{22}, s_i^{23})$
 - 3: NEH_i değerlerini büyükten küçüğe sırala
 - 4: **bitiş döngü**{i}
 - 5: Sıralamaya göre öncelikle en baştaki iki iş için olası esnek işlem atamalarını ve sıralamayı dene
 - 6: En küçük yayılma zamanını veren sıralamayı ve atamayı sabitle
 - 7: Çizelgelenmeyen iş kalmayana kadar Tekrarla
 - 8: Yeni eklenen iş için esnek işlemlerin olası 4 atamasını ve bir önceki adımda elde edilen sıralamayı bozmayan olası sıralamalar ile yayılma zamanını hesapla.
 - 9: **Son** tekrar
-

Algoritma 18 Problem 4: Modifiye NEH Algoritması 3

Girdi: $f_i^1, f_i^2, f_i^3, s_i^{11}, s_i^{12}, s_i^{22}, s_i^{23}$.

Çıktı: C_{max}, NEH_i

- 1: $i = 1; i \leq n; i ++$ (bütün parçalar için)
 - 2: $NEH_i = f_i^1 + f_i^2 + f_i^3 + \max(s_i^{11}, s_i^{12}) + \max(s_i^{22}, s_i^{23})$
 - 3: NEH_i değerlerini büyükten küçüğe sırala
 - 4: **bitiş döngü**{i}
 - 5: Sıralamaya göre öncelikle en baştaki iki iş için olası esnek işlem atamalarını ve sıralamayı dene
 - 6: En küçük yayılma zamanını veren sıralamayı ve atamayı sabitle
 - 7: Çizelgelenmeyen iş kalmayana kadar Tekrarla
 - 8: Yeni eklenen iş için esnek işlemlerin olası 4 atamasını ve bir önceki adımda elde edilen sıralamayı bozmayan olası sıralamalar ile yayılma zamanını hesapla.
 - 9: **Son** tekrar
-

Algoritma 19 Problem 4: Modifiye NEH Algoritması 4

Girdi: $f_i^1, f_i^2, f_i^3, s_i^{11}, s_i^{12}, s_i^{22}, s_i^{23}$.

Çıktı: C_{max}, NEH_i

- 1: $i = 1; i \leq n; i ++$ (bütün parçalar için)
 - 2: $NEH_i = f_i^1 + f_i^2 + f_i^3 + \min(s_i^{11}, s_i^{12}) + \max(s_i^{22}, s_i^{23})$
 - 3: NEH_i değerlerini büyükten küçüğe sırala
 - 4: **bitiş döngü**{i}
 - 5: Sıralamaya göre öncelikle en baştaki iki iş için olası esnek işlem atamalarını ve sıralamayı dene
 - 6: En küçük yayılma zamanını veren sıralamayı ve atamayı sabitle
 - 7: Çizelgelenmeyen iş kalmayana kadar Tekrarla
 - 8: Yeni eklenen iş için esnek işlemlerin olası 4 atamasını ve bir önceki adımda elde edilen sıralamayı bozmayan olası sıralamalar ile yayılma zamanını hesapla.
 - 9: **Son** tekrar
-

Algoritma 20 Problem 4: Modifiye NEH Algoritması 5

Girdi: $f_i^1, f_i^2, f_i^3, s_i^{11}, s_i^{12}, s_i^{22}, s_i^{23}$.

Çıktı: C_{max}, NEH_i

- 1: $i = 1; i \leq n; i ++$ (bütün parçalar için)
 - 2: $NEH_i = f_i^1 + f_i^2 + f_i^3 + \max(s_i^{11}, s_i^{12}) + \min(s_i^{22}, s_i^{23})$
 - 3: NEH_i değerlerini büyükten küçüğe sırala
 - 4: **bitiş döngü**{i}
 - 5: Sıralamaya göre öncelikle en baştaki iki iş için olası esnek işlem atamalarını ve sıralamayı dene
 - 6: En küçük yayılma zamanını veren sıralamayı ve atamayı sabitle
 - 7: Çizelgelenmeyen iş kalmayana kadar Tekrarla
 - 8: Yeni eklenen iş için esnek işlemlerin olası 4 atamasını ve bir önceki adımda elde edilen sıralamayı bozmayan olası sıralamalar ile yayılma zamanını hesapla.
 - 9: **Son** tekrar
-

[1, 1.2] yine tek düze dağılıma sahip (α, β) katsayılarıyla türetilmiştir. Bir önceki bölümdeki deneysel çalışmanın benzeri bu çalışma için de uygulanmıştır ve her bir iş sayısı için $(2^7) \cdot 2 \cdot 5 = 1280$ farklı problem seti üretilmiştir. Bahsi geçen problemler için yayılma zamanı en küçüklenmesi, hem karmaşık tamsayılı programlama ile modellenmiş ve GAMS CPLEX 12.0 çözücüsü ile çözdürülmüş hem de geliştirilen sezgiseller ile sağlanmaya çalışılmıştır. Her problem için kaynak kısıtı olarak GAMS çözücüsünde, süre limiti olarak 900 saniye kullanılmıştır. Problemin güçlü NP-Zor olmasından dolayı Tablo 6.2’de de gösterildiği gibi problem sayısı ve işlem süreleri arttıkça zaman limitine takılan problem sayısı da artmaktadır.

Tablo 6.2: Problem 4: CPLEX Çözüm Durumları

	Optimal Çözüm			Zaman Limiti		
	Düşük	Yüksek	Toplam	Düşük	Yüksek	Toplam
20 İş	536	346	882	104	294	398
50 İş	112	78	190	528	562	1090

NEH algoritmasının çalışması için gereken sıralama ölçütleri Tablo 6.1’de gösterilmektedir. Geliştirilen sezgisellerin performansları, önceki bölümlerde olduğu gibi, CPLEX çözücüsünün verdiği sonuçlar dikkate alınarak ölçülmektedir. Bölüm 5’te Problem 2 ve 3 için yapılan yüzde fark tanımı Problem 4 için de geçerli olmakla beraber performans ölçütü olarak da bu tanım kullanılmaktadır.

Tablo 6.3: Problem 4: 20 İş İçin Sezgiseller - Alt Sınır Yüzde Fark

	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (10)	Yüksek (100)
NEH 1	3.76	3.95
NEH 2	3.46	3.76
NEH 3	3.54	3.78
NEH 4	3.55	3.82
NEH 5	3.5	3.81
Min NEH	2.68	2.96
CPLEX	0.47	0.83

Tablo 6.4: Problem 4: 50 İş için Sezgiseller - Alt Sınır Yüzde Fark

	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (10)	Yüksek (100)
NEH 1	10.53	6.82
NEH 2	10.52	6.75
NEH 3	10.49	6.67
NEH 4	10.49	6.78
NEH 5	10.54	6.81
Min NEH	10.03	6.21
CPLEX	9.3	5.45

Tablo 6.3 ve 6.4'de sezgisellerden elde edilen sonuçlar alt sınıra ortalama yüzde fark uzaklığı göstermektedir. Tabloların son satırında bulunan "CPLEX" verilen zaman limitinde matematiksel model ile ulaşılan sonucun aynı süre içerisinde elde edilen alt sınıra olan ortalama uzaklığı göstermektedir. Problem 3 ile benzer şekilde Problem 4 için de 2560 problem üzerinde yapılan çalışmaya göre üretilen parça sayısındaki artış alt sınıra olan uzaklığı arttırmaktadır. Bu problem için ortalamalara bakıldığında farklı NEH ölçütlerinin ulaştığı çözümler arasında belirgin bir fark gözlemlenmemektedir. Problem 4 için de Problem 2 ve 3'e benzer olarak daha verimli çözümlere ulaşabilmek için "Min NEH" kullanılmıştır. 5 sezgiselden de elde edilen en küçük sonuç orjinal problemin çözümü olarak kullanılmaktadır. "Min NEH" ile ulaşılan sonuçların işlem süresi seviyelerine ve iş sayısına göre CPLEX alt sınırından farkı Tablo 6.5'te verilmektedir. Bu tabloda 50 iş için maksimum uzaklığın %149'a ulaştığı gözlenmektedir fakat verilen zaman limiti içerisinde ulaşılan alt sınır optimal sonuçtan çok uzak olabilmektedir. Bu kadar yüksek fark içeren problemin en iyi CPLEX değeri incelendiğinde sezgiselin bu değerden %2 oranında daha küçük bir sonuca ulaştığı tespit edilmiştir. Aynı zamanda alt sınıra olan ortalama uzaklığın 50 iş için %8 civarında olması sezgiselin etkin sonuçlara ulaşabildiğini göstermektedir

Benzer şekilde Tablo 6.6 incelendiğinde, CPLEX tarafından verilen zaman limitinde ulaşılan en iyi çözümlere en büyük uzaklığın %8,46 olduğu görülmektedir. Bununla birlikte özellikle 50 parçalı ve işlem sürelerinin 100'den fazla olduğu problemlerde

Tablo 6.5: Problem 4: İş Sayısına göre Min NEH Alt Sınır Yüzde Fark

		İşlem Süresi Taban Değeri Seviyesi	
		Düşük (10)	Yüksek (100)
20 iş	Ortalama	2.68	2.96
	Maksimum	25.56	18.61
50 iş	Ortalama	10.03	6.21
	Maksimum	149.33	46.62

sezgisel ile elde edilen sonuçların %33'e kadar daha verimli sonuçlar verdiği gözlemlenmektedir. Parça sayısı arttıkça problem zorlaşmaktadır ve CPLEX alt sınır ile en iyi CPLEX değeri arasındaki boşluk artmaktadır.

Tablo 6.6: Problem 4: İş Sayısına göre Min NEH En İyi CPLEX Yüzde Fark

		İşlem Süresi Taban Değeri Seviyesi	
		Düşük (10)	Yüksek (100)
20 iş	Ortalama	2.21	2.12
	Maksimum	8.46	7.96
	Minimum	0	-0.03
50 iş	Ortalama	0.84	0.42
	Maksimum	0.88	7.67
	Minimum	-19.98	-33.32

Çalışmada sezgisellerle optimal çözüme ulaşılan problem sayısı, sezgisel ile ulaşılan çözüm değerinin CPLEX alt sınır değeri ile aynı değere sahip olup olmadığı kontrol edilerek hesaplanmıştır. Tablo 6.7'ye göre 20 iş için optimal çözüme ulaşıldığından emin olduğumuz 882 problemde 34'ünde, 50 iş için kesin olarak optimal çözüme ulaşılmış 190 problemde 25'inde optimal çözüme ulaşılmıştır. Fakat özellikle Problem 4 için zaman limitine takılan problem sayısı çok fazladır. 50 iş için bu sayı 1090'dır (Tablo 6.2). Bu nedenle bu problem için Tablo 6.7'de yer alan sonuçlar sezgiselin performansı açısından yanıltıcı olabilir.

Sezgiselin etkinliği zaman açısından incelendiğinde ise, sezgiselin çok kısa sürede

Tablo 6.7: Problem 4: İş Sayısına göre Min NEH Optimal Çözülen Problem Sayısı

	İşlem Süresi Taban Değeri Seviyesi		
	Düşük	Yüksek	Toplam
20 İş	18 (536)	16 (346)	34 (882)
50 İş	14 (112)	11 (78)	25 (190)

çözüm verirken CPLEX çözücüsünün çok uzun sürede olurlu çözüm verdiği hatta 50 parçalı problemlerde olurlu çözüme ulaşamadığı gözlemlenmiştir. Sezgiselin çözüm süresi 1 saniyeden daha kısa bir süre gerektirirken, 900 saniye zaman limit ile çalışan CPLEX 20 iş için ortalama 225, 50 iş için 813 saniyeye ihtiyaç duymaktadır.

Tablo 6.8: Problem 3 20 İş için Sezgisel ve CPLEX Çözüm Süreleri

20 İş	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (sn)	Yüksek (sn)
NEH 1	0.04	0.04
NEH 2	0.04	0.04
NEH 3	0.04	0.04
NEH 4	0.05	0.05
NEH 5	0.04	0.04
Min NEH	0.19	0.19
CPLEX	185.35	464.69

Bu bölümde ele alınan Problem 4 için üretilecek parça sayısının, işlem süreleri taban değeri ve işlem süreleri aralık seviyelerinin çözüm süresi ve yüzde farklar üzerindeki etkisini test etmek amacıyla Problem 2 ve 3'e benzer şekilde 2^4 faktoriyel tasarım oluşturulmuştur. Faktörler, parça sayısı, taban değerler, sabit işlem sürelerinin aralık seviyesi ve esnek işlem sürelerinin aralık seviyelerinden oluşmaktadır. Tablo 6.10'da yer alan tasarım sahip olunan 9 değişkenin bahsedilen 4 faktörün etkisini ölçebilmek için nasıl sabitlendiğini göstermektedir. Tüm sabit ve esnek işlemler kendi aralarında aynı seviyeye sahip iken elde edilen koşullar tasarımda değerlendirilmiştir. Tasarımda yer alan 0 değerleri düşük seviyeyi 1 değerleri ise yüksek seviyeyi göstermektedir. Düşük ve yüksek seviyeler sırasıyla parça sayısı için 20 ve 50;

Tablo 6.9: Problem 3 50 İş için Sezgisel ve CPLEX Çözüm Süreleri

20 İş	İşlem Süresi Taban Değeri Seviyesi	
	Düşük (sn)	Yüksek (sn)
NEH 1	0.4	0.41
NEH 2	0.46	0.47
NEH 3	0.43	0.43
NEH 4	0.48	0.48
NEH 5	0.41	0.41
Min NEH	2.15	2.16
CPLEX	805.85	822.82

taban değer için 10 ve 100; aralık için (taban değer/2) ve (taban değer ·4) olarak kullanılmıştır.

Analiz Minitab 16.0 paket program ile gerçekleştirilmiştir. Bu programa ait çıktıların değerlendirilmesi sırasında tablolarda yer alan p-değerleri kullanılmaktadır. Yapılan testler sırasında kullanılan hipotezler Problem 4 için de problem 2 ve 3' te olduğu şekliyle yer almaktadır.

H_0 = Faktör istatistiksel olarak %95 güven düzeyinde anlamlı değildir.

$H_{alternatif}$ = Faktör istatistiksel olarak %95 güven düzeyinde anlamlıdır.

Tablolarda faktörlere karşılık gelen p-değerleri 0.05'ten küçükse boş (null) hipotez reddedilmekte ve alternatif hipotez doğrulanmaktadır. Aksi takdirde boş hipotez reddedilemeyerek, faktörün etkisiz olduğu anlaşılmaktadır. Analizin geri kalan kısmı bu şekilde değerlendirilmiştir.

CPLEX çözüm süresine ait istatistiksel değerlerin Minitab çıktısı Tablo 6.11'de verilmiştir. Bu tabloya göre p-değeri 0.05'ten küçük olan temel faktör sadece parça sayısıdır. Fakat bazı faktörlerin etkileşimi de CPLEX çözüm süresini etkilemektedir. Taban değerleri sabit ve esnek işlem süreleri ile 2'li ve 3'lü interaksyonu çözüm süresi için %95 güvenilirlik düzeyinde istatistiksel olarak anlamlı etkiye sahiptir. Taban değerinin 100 olduğu ve aynı zamanda işlem süreleri aralık seviyesi taban değerinin 4 katı

Tablo 6.10: Problem 4: Faktör Tasarımı

Parça Sayısı	İşlem Süresi Taban	Aralık Seviyeleri							
		f^1	f^2	f^3	s^{11}	s^{12}	s^{22}	s^{23}	
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1
0	0	1	1	1	0	0	0	0	0
0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	0	0	0	0
0	1	0	0	0	1	1	1	1	1
0	1	1	1	1	0	0	0	0	0
0	1	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	1	1	1	1	1
1	0	1	1	1	0	0	0	0	0
1	0	1	1	1	1	1	1	1	1
1	1	0	0	0	0	0	0	0	0
1	1	0	0	0	1	1	1	1	1
1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1

olduğu durumda CPLEX çözüm süresi artmaktadır. Parça sayısının 20 parçadan 50 parçaya çıkarılması da problemin çözülmesi için gereken zamanı hızla arttırmaktadır. Bu durum Tablo 6.2’de gözlemlenen değerlerle örtüşmektedir.

CPLEX Alt Sınır ve CPLEX En İyi çözüm arasındaki yüzde boşluk üzerinde parça sayısının, aralık ve taban değer seviyelerinin anlamlı etkiye sahip olmadığı gözlemlenmiştir. Tablo 6.12 bu analize dair Minitab çıktısı değerlerini göstermektedir. Tabloda yer alan p-değerlerine göre 0.05’ten küçük olan hiç bir faktör bulunmamaktadır. Problem 4 için, %95 anlamlılık düzeyinde alt sınır ile CPLEX çözümü arasındaki boşluk belirlediğimiz parametre değerlerinin değişiminden etkilenmemektedir.

Sezgisel sonucu ve CPLEX alt sınır arasındaki yüzde fark üzerinde çalışmada yer alan faktörlerin istatistiksel olarak anlamlı olmadığı Tablo 6.13’deki p-değerlerinden anlaşılmaktadır. Tüm temel faktörlerin ve faktörlerin interaksyonlarının p-değerlerinin 0.05’ten büyük olduğu gözlemlenmiştir.

CPLEX ile edinilen çözümlerden biri alt sınır bir diğeri "CPLEX En İyi" olarak adlandırılan verilen süre limiti içerisinde ulaşılan çözümdür. CPLEX En İyi çözümün sezgisel değere olan uzaklığı üzerinde parça sayısı, işlem süresi taban değerleri

Tablo 6.11: Problem 4: CPLEX Çözüm Süresi Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		144,5	96,50	1,50	0,139
Parça Sayısı	1259,5	629,7	136,47	4,61	0,000
Taban Değer	385,2	192,6	136,47	1,41	0,163
Aralık f	-233,7	-116,8	136,47	-0,86	0,395
Aralık s	203,7	101,8	136,47	0,75	0,458
Parça Sayısı-Taban Değer	-133,5	-66,8	192,99	-0,35	0,731
Parça Sayısı-Aralık f	303,9	152,0	192,99	0,79	0,434
Parça Sayısı-Aralık s	-58,5	-29,2	192,99	-0,15	0,880
Taban Değer-Aralık f	850,5	425,2	192,99	2,20	0,031
Taban Değer-Aralık s	922,2	461,1	192,99	2,39	0,020
Aralık f-Aralık s	580,6	290,3	192,99	1,50	0,137
Parça Sayısı-Taban Değer-Aralık f	-920,7	-460,3	272,94	-1,69	0,097
Parça Sayısı-Taban Değer-Aralık s	-1067,4	-533,7	272,94	-1,96	0,055
Parça Sayısı-Aralık f-Aralık s	-655,9	-327,9	272,94	-1,20	0,234
Taban Değer-Aralık f-Aralık s	-1197,4	-598,7	272,94	-2,19	0,032
Parça Sayısı-Taban Değer-Aralık f-Aralık s	1272,6	636,3	385,99	1,65	0,104

Tablo 6.12: Problem 4: CPLEX Alt Sınır CPLEX En İyi Yüzde Fark Faktörler

Term	Effect	Coef	SE Coef	T	P
Constant		-0,000	1,682	-0,00	1,000
Parça Sayısı	7,076	3,538	2,379	1,49	0,142
Taban Değer	0,009	0,005	2,379	0,00	0,998
Aralık f	0,000	0,000	2,379	0,00	1,000
Aralık s	0,056	0,028	2,379	0,01	0,991
Parça Sayısı-Taban Değer	-6,880	-3,440	3,365	-1,02	0,310
Parça Sayısı-Aralık f	10,473	5,237	3,365	1,56	0,125
Parça Sayısı-Aralık s	-4,723	-2,362	3,365	-0,70	0,485
Taban Değer-Aralık f	0,041	0,021	3,365	0,01	0,995
Taban Değer-Aralık s	0,020	0,010	3,365	0,00	0,998
Aralık f-Aralık s	0,022	0,011	3,365	0,00	0,997
Parça Sayısı-Taban Değer-Aralık f	-8,242	-4,121	4,759	-0,87	0,390
Parça Sayısı-Taban Değer-Aralık s	4,954	2,477	4,759	0,52	0,604
Parça Sayısı-Aralık f-Aralık s	-5,574	-2,787	4,759	-0,59	0,560
Taban Değer-Aralık f-Aralık s	-0,122	-0,061	4,759	-0,01	0,990
Parça Sayısı-Taban Değer-Aralık f-Aralık s	3,161	1,581	6,730	0,23	0,815

Tablo 6.13: Problem 4: CPLEX Alt Sınır Sezgisel Yüzde Fark Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		0,645	0,5714	1,13	0,263
Parça Sayısı	-0,673	-0,337	0,8080	-0,42	0,678
Taban Değer	-0,422	-0,211	0,8080	-0,26	0,795
Aralık f	1,298	0,649	0,8080	0,80	0,425
Aralık s	0,735	0,367	0,8080	0,45	0,651
Parça Sayısı-Taban Değer	0,142	0,071	1,1427	0,06	0,951
Parça Sayısı-Aralık f	-0,652	-0,326	1,1427	-0,29	0,776
Parça Sayısı-Aralık s	0,537	0,269	1,1427	0,24	0,815
Taban Değer-Aralık f	1,165	0,583	1,1427	0,51	0,612
Taban Değer-Aralık s	-0,144	-0,072	1,1427	-0,06	0,950
Aralık f-Aralık s	-2,109	-1,054	1,1427	-0,92	0,360
Parça Sayısı-Taban Değer-Aralık f	2,710	1,355	1,6161	0,84	0,405
Parça Sayısı-Taban Değer-Aralık s	-0,284	-0,142	1,6161	-0,09	0,930
Parça Sayısı-Aralık f-Aralık s	0,366	0,183	1,6161	0,11	0,910
Taban Değer-Aralık f-Aralık s	-0,254	-0,127	1,6161	-0,08	0,938
Parça Sayısı-Taban Değer-Aralık f- Aralık s	-2,245	-1,123	2,2855	-0,49	0,625

ve işlem sürelerinin aralıklarının istatistiksel olarak anlamlı etkileri yoktur. Tablo 6.14'deki temel faktörlerin ve faktör etkileşimlerinin p-değerlerinin 0.05'ten büyük olması nedeniyle çalışmadaki boş hipotez reddedilememektedir.

Tablo 6.14: Problem 4: CPLEX En İyi Sezgisel Yüzde Fark Faktörleri

Term	Effect	Coef	SE Coef	T	P
Constant		0,826	0,6025	1,37	0,175
Parça Sayısı	-1,468	-0,734	0,8521	-0,86	0,392
Taban Değer	-0,624	-0,312	0,8521	-0,37	0,716
Aralık f	2,125	1,062	0,8521	1,25	0,217
Aralık s	-0,017	-0,009	0,8521	-0,01	0,992
Parça Sayısı-Taban Değer	1,079	0,539	1,2050	0,45	0,656
Parça Sayısı-Aralık f	-5,612	-2,806	1,2050	-2,33	0,023
Parça Sayısı-Aralık s	0,620	0,310	1,2050	0,26	0,798
Taban Değer-Aralık f	0,236	0,118	1,2050	0,10	0,922
Taban Değer-Aralık s	1,409	0,705	1,2050	0,58	0,561
Aralık f-Aralık s	-0,202	-0,101	1,2050	-0,08	0,933
Parça Sayısı-Taban Değer-Aralık f	3,436	1,718	1,7041	1,01	0,317
Parça Sayısı-Taban Değer-Aralık s	-2,237	-1,118	1,7041	-0,66	0,514
Parça Sayısı-Aralık f-Aralık s	2,062	1,031	1,7041	0,61	0,547
Taban Değer-Aralık f-Aralık s	-2,593	-1,297	1,7041	-0,76	0,450
Parça Sayısı-Taban Değer-Aralık f-Aralık s	0,948	0,474	2,4100	0,20	0,845

6.3 Sonuçlar

İki esnek işleme sahip, farklı tip parça üretimi gerçekleştiren 3 makineli akış atölyesinde üretim çıktısını en büyükmek için esnek işlemlerin atamasına ve parçaların sıralamasına karar verebilmek için NEH temel alınarak 5 farklı sıralama ölçütüne sahip sezgiseller geliştirilmiştir. Geliştirilen sezgiseller, bir deneysel çalışma oluşturularak 2560 farklı problem üzerinde uygulanarak elde edilen sonuçlar, CPLEX çözücüsü ile elde edilen sonuçlarla karşılaştırılmıştır. Karşılaştırma sonucu geliştirilen sezgisellerin CPLEX sonuçlarına olan uzaklıkları 20 iş için ortalama %2,16 ve 50 iş için %0,63 tür. Altısınıra göre bakılan değerlerde yüzde uzaklıkların çok yüksek olduğu durumlarla karşılaşılmaktadır fakat bu değerler en iyi CPLEX değerleri

ile kıyaslandığında sezgiselin çok daha iyi sonuçlara ulaşabildiği gözlemlenmiştir. CPLEX çözücüsü ile ulaşılan çözümler ve sezgisel değeri arasındaki uzaklık parça sayısı ve işlem sürelerinden etkilenmemektedir. Sezgiselin zaman açısından performansı çok yüksektir, bir problemin çözüm için 1 saniyeden kısa süreler söz konusudur. Bunun yanında CPLEX çözümlerinde verilen zaman limitinde 50 parçalık problemlerin büyük çoğunluğunun optimal çözüme ulaşamadığı gözlemlenmiştir. CPLEX çözüm süresi parça sayısından ve işlem sürelerinin taban değerleriyle aralık seviyelerinden etkilenmektedir.

7. SONUÇ ve DEĞERLENDİRME

Bu çalışmada 2 ve 3 makineli akış tipi üretim sisteminde n farklı tip parça çizelgeleme problemi ele alınmıştır. Daha önceden belirlenmiş makineler arasında esnek işlem bulunmaktadır ve bu işlemler birden fazla makine üzerinde işlem görebilmektedir. Sistemlerdeki esnek işlem sayısı, esnek makine sayısı ve makinelerin özdeşliğine göre değişim gösteren 4 farklı problem ele alınmıştır. Problemlerde göz önünde bulundurulan amaç fonksiyonu yayılma zamanının enküçüklenmesidir. Bu amaç fonksiyonu doğrultusunda çalışmada ele alınan problemlerde parça sıralaması ve esnek işlemlerin atmasına karar verilmesi gerekmektedir.

Çalışmada öncelikle ele alınan problemlere yönelik kapsamlı bir literatür taraması yapılmıştır. Ardından 4 problemde detaylı olarak tanımlanarak bu problemler için genel bir karma tamsayılı matematiksel model geliştirilmiştir. Problem 2, 3 ve 4'ün NP-Zor olması nedeniyle parça sayısı arttıkça problemlerin çözümü için gereken süre hızla artmıştır. Problem 1'in ise NP-Zor olup olmadığı tam olarak bilinmemektedir. Bu nedenlerden dolayı problemlerin çözümü için sezgisel çözüm yöntemleri geliştirilmiştir. Problem 1 için bazı önteoremler geliştirilmiş ve bu önteoremlere dayanarak Johnson algoritmasını temel alan optimal sonuçlara çok yakın çözümler veren kaliteli bir sezgisel geliştirilmiştir. Problem 2'ye çözüm önerisi olarak ise CDS, Gupta ve NEH tabanlı 11 farklı sezgisel geliştirilmiştir. Sonrasında ise CDS ve Gupta temel alınan sezgisellerin performansının çok düşük olduğu gözlemlenmiştir. Bu gözleme dayalı olarak Problem 3 ve 4 için farklı sıralama kriterlerine sahip NEH tabanlı sezgiseller önerilmiştir. Geliştirilen sezgisellerin hepsi makineler arası stok alanının sonsuz olduğunu varsaymaktadır. Polinom zamanda çözüm üreten bu sezgiseller optimal sonucu garanti edememektedir. Bu nedenle sezgisellerin performansını test etmek amacıyla her biri için ayrı ayrı deneysel çalışmalar uygulanmıştır.

Problemi zorlaştıracakı düşünülen çeşitli faktörler belirlenerek oluşturulan problem setleri GAMS CPLEX 12.0 çözücüsü ile ve C++ ortamında kodlanmış sezgiseller

ile çözdürülerek elde edilen sonuçlar kıyaslanmıştır. Parametre değerlerini etkileyen değerler makineler üzerindeki sabit ve esnek işlem süreleri ile doğrudan alakalı olduğu için Problem 1 için 480, Problem 2 için 640, Problem 3 ve 4 için 2560'şar adet farklı problem çözdürülmüştür.

Problem 1 için geliştirilen sezgisel çözdürdüğümüz problemlerde optimal değerlere ulaşmıştır fakat her zaman optimalliği garanti etmemektedir. Aynı zamanda çok kısa sürede bu kadar iyi sonuçlar vermesi bu sezgiselin yüksek performansa sahip olduğunu göstermektedir. Problem 2 için geliştirilen CDS ve Gupta tabanlı sezgiseller çözüm süresi açısından iyi bir performansa sahip olsa da CPLEX ile elde edilen çözümlere olan uzaklıkları (Tablo 5.15) NEH ile elde edilen uzaklıklara göre %10 civarında daha fazladır. Oluşturulan NEH sezgiselleri arasından minimum sonucu veren sezgiselin sonuçları çözüm yöntemi olarak kullanılmıştır. Bu çözüm yöntemi CPLEX ile 900 saniyelik süre limitinde ulaşılan sonuçlara ortalama %2.91 yaklaşmıştır. CPLEX çözüm süresinin parça sayısı ve işlem sürelerine ait taban değer arttıkça arttığı gözlemlenmiştir. Kaliteli sonuçlara ulaşılan bu sezgisel sıralama ölçütleri değiştirilerek Problem 3 ve 4 için de çözüm yöntemi olmuştur ve CPLEX ile ulaşılan çözümlere çok yakın değerler verdiği gözlemlenmiştir. Hatta bazı problemler için sezgiselin daha iyi sonuçlara ulaştığı farkedilmiştir. Bununla birlikte, Problem 3 için yapılan deneysel çalışmada sezgisellerin CPLEX çözümlerine olan yüzde farkları üzerinde esnek işlem sürelerinin aralığının istatistiksel olarak anlamlı etkiye sahip olduğu saptanmıştır. Tüm problemler için CPLEX çözüm süresi parça sayısındaki artıştan etkilenmektedir. Bununla birlikte taban değerler ve işlem seviyelerinin aralık seviyelerinin etkileşimi de CPLEX çözüm süresi üzerinde etkilidir.

Bu çalışmayı geliştirecek olan çeşitli çalışmalar yapılabilir. Makineler arasındaki ara stok alanının bulunmadığı veya sınırlı olduğu problemler ele alınabilir. Matematiksel modelde ara stok alanı ile kısıt eklenmesi halinde problem için olurlu alan daralacak, problem zorlaşacaktır. Makinelerin ve parçaların özdeş olmadığı bu 3-makineli akış tipi sistemlerde çizelgeleme problemleri NP-Zor problemlerdir. Dolayısıyla makine sayısının daha fazla olduğu durumlarda yapılacak çalışmalarda sezgisel çözüm yöntemlerinden yararlanılmalıdır. Bu çalışmada kullanılan sezgiseller m -makineli sistemler için de çözüm yöntemi olarak temel alınabilir. Fakat ulaşılan sonuçların kalitesini arttıracak ve çözüm süresinin çok fazla artmayacağı sezgiseller geliştirilebilir veya varolan sezgiseller üzerinden iyileştirmeler yapılabilir. Diğer yandan Problem

1 olarak tanıtılan 2-makinele tek esnek işleme sahip olan makinelerin özdeş olduğu sistemin NP-Zor olup olmadığı açık bir konudur. Bu konuyla ilgili çalışmalar geliştirilebilir.

KAYNAKLAR

- [1] M.S. Aktürk, H. Gültekin, O.E. Karasan, Robotic cell scheduling with operational flexibility, *Discrete Applied Mathematics*, 145(3):334-348, 2005.
- [2] A. Allahverdi, J. N. D. Gupta, T.Aldowaisan, A Review of Scheduling Research Involving Setup Considerations, *Omega*, 27:219-239, 1999.
- [3] R. Anuar, Y. Bukchin, Design and Operation of Dynamic Assembly Lines Using Work Sharing, *International Journal of Production Research*, 44:4043-4065, 2006.
- [4] R. Askin, J. Chen, Dynamic Task Assingment for Throughput Maximization with Worksharing, *European Journal of Operational Research*, 168:853-869, 2006.
- [5] A. Babayan, D. He, Solving the n -job 3-stage Flexible Flowshop Scheduling Problem Using an Agent-Based Approach, *International Journal of Production Research*, 42(4):777-799,2004.
- [6] P. Baptiste, L.K. Hguny, A branch-and-bound algorithm for the F/no-idle/Cmax, *Proceedings of the International Conference on Industrial Engineering and Production Management*, 1:429-438, 1997.
- [7] I. Baybars, A Survey of Exact Algorithms for The Simple Assembly Line Balancing Problem. *Management Science*, 32:909-932, 1986.
- [8] R. Beach, A.P. Muhlemann, D.H.R. Price, A. Paterson, J.A. Sharp, A Review of Manufacturing Flexibility. *European Journal of Operational Research*, 122:41-57, 2000.

- [9] C. Becker, A. Scholl, A Survey on Problems and Methods in Generalized Assembly Line Balancing. *European Journal of Operational Research*, 168:694-715, 2006.
- [10] J.J. Bernardo, Z. Mohamed, The Measurement and Use of Operational Flexibility in the Loading of Flexible Manufacturing Systems. *European Journal of Operational Research*, 60: 144-155, 1992.
- [11] F. F. Boctor, A Multiple-Rule Heuristic for Assembly Line Balancing. *Journal of the Operational Research Society*, 46:62-69, 1995.
- [12] N. Boysen, M. Fliedner, A. Scholl, A Classification of Assembly Line Balancing Problems. *European Journal of Operational Research*, 183:674-693, 2006.
- [13] J. Browne, D. Dubois, K. Rathmill, S. Sethi, K. Stecke, Types of Flexibilities and Classification of Flexible Manufacturing Systems. *Division of Research Graduate School of Business Administration The University of Michigan*, 367, 1984.
- [14] G. Campbell, A. Dudek and L. Smith, A Heuristic Algorithm For The n Job, m Machine Sequencing Problem, *Institute for Operations Research and the Management Science*, 10:630-637, 1970.
- [15] Y. Cho, S. Sahni, Preemptive Scheduling of Independent Jobs with Release and Due Times on Open, Flow and Job Shops. *Operations Research*, 29:511-522, 1981.
- [16] C. Chung, I. Chen, Managing the Flexibility of Flexible Manufacturing Systems for Competitive Edge. *In: Liberatore, M.J. (Ed.), Selection and Evaluation of Advanced Manufacturing Technologies. Springer, New York*, 280-303, 1990.
- [17] Y. Crama, J. Van de Klundertb , F. C.R. Spieksma, Production Planning Problems in Printed Circuit Board Assembly. *Discrete Applied Mathematics*, 123:339-361, 2002.
- [18] Y. Crama, H. Gultekin, Throughput Optimization in 2-Machine Flowshops with Flexible Operations. *Journal of Scheduling*, 13(3): 227-243, 2010.
- [19] R. Daniels, J. Mazzola, Flow Shop Scheduling with Resource Flexibility. *Operations Research*, 42(3):504-522, 1994.

- [20] R. Daniels, J. Mazzola, S. Dailun, Flow Shop Scheduling with Partial Resource Flexibility. *Management Science*, 50(5):658-669, 2004.
- [21] W. Jordan, C. Graves, Principles on the Benefits of Manufacturing Process Flexibility. *Management Science*, 41(4): 577-594, 1995.
- [22] M. R. Garey, D. S. Johnson, R. Sethi, The Complexity of Flowshop and Jobshop Scheduling. *Journal Mathematics of Operations Research*, 1:117-129, 1976.
- [23] D. Gerwin, Manufacturing Flexibility: A Strategic Perspective. *Management Science*, 39:395-410, 1993.
- [24] P. C. Gilmore, R. E. Gomory, Sequencing a One State-Variable Machine: A Solvable Case of the Traveling Salesman Problem. *Operations Research*, 12:655-679, 1964.
- [25] J. Gong, L. Wang, S. Zhang, A New Workforce Cross-Training Policy for a U-shaped Assembly Line, *Information and Management Engineering*, 235:529-536, 2011.
- [26] T. Gonzalez, S. Sahni, Flowshop and Jobshop Schedules: Complexity and Approximation. *Operations Research*, 26:36-52, 1978.
- [27] R. L. Graham, E. L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, Optimization and Approximation in Deterministic Sequencing and Scheduling: a Survey. *Proceedings of the Advanced Research Institute on Discrete Optimization and Systems Applications of the Systems Science Panel of NATO and of the Discrete Optimization Symposium*, 5:287-326, 1979.
- [28] H. Gultekin, M.S. Aktürk, O.E. Karasan, Cyclic Scheduling of a robotic cell with tooling constraints. *European Journal Operational Research*, 174:777-796, 2006.
- [29] H. Gultekin, M.S. Aktürk, and O.E. Karasan, Bi-criteria Robotic Cell Scheduling. *Journal of Scheduling*, 11(6):457-473, 2008.
- [30] H. Gultekin, Scheduling in Flowshops with Flexible Operations: Throughput Optimization and Benefits of Flexibility, *International Journal of Production Economics*, 140:900-9111, 2012.

- [31] Z. X. Guo, W. K. Wong, S. Y. S. Leung, J. T. Fan, S. F. Chan, A Genetic-Algorithm-Based Optimization Model for Solving the Flexible Assembly Line Balancing Problem with Work Sharing and Workstation Revisiting, *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions*, 38:218-228, 2008.
- [32] N. D. Gupta, A Functional Heuristic Algorithm for the Flowshop Scheduling Problem, *Operational Research Society*, 22:39-47, 1971.
- [33] Y. P. Gupta, S. Goyal, Flexibility in Manufacturing Systems. *European Journal of operational Research*, 43:119-135, 1989.
- [34] N. D. J. Gupta, C. P. Koulamas, G.J. Kyparisis, C.N. Potts, V.A. Strusevich, Scheduling Three-Operation Jobs in a Two Machine Flow Shop to Minimize Makespan. *Annals of Operations Research*, 129:171-185, 2004.
- [35] J. N. D. Gupta, E. F. Jr. Stafford, Flowshop Scheduling Research After Five Decades. *European Journal of Operational Research*, 169:699-711, 2006.
- [36] S. T. Hackman, M. J. Magazine, T. S. Wee, Fast, Effective Algorithms for Simple Assembly Line Balancing Problems. *Operations Research*, 37: 916-924, 1989.
- [37] N. G. Hall, C. Sriskandarajah, A Survey of Machine Scheduling Problems with Blocking and No-wait in Process. *Operational Research*, 44:510-525, 1996.
- [38] S. Hejazi, S. Saghafian, Flowshop-scheduling Problems with Makespan Criterion: A review. *International Journal of Production Research*, 14:2895-2929, 2005.
- [39] S. M. Johnson, Optimal Two and Three-Stage Production Schedules with Setup Times Included, *Naval Research Logistics Quarterly*, 1:61-68, 1954.
- [40] R. Klein, A. Scholl, Maximizing the Production Rate in Simple Assembly Line Balancing- A Branch and Bound Procedure. *European Journal of Operational Research*, 91:367-385, 1996.
- [41] S.B. Liu, H.L. Ong, H.C. Huang, A Bi-directional Heuristic for Stochastic Assembly Line Balancing Problems. *International Journal of Advanced Manufacturing Technology*, 25:71-77, 2005.

- [42] M. Nawaz, Jr, E. Enscore, and I. Ham, A Heuristic Algorithm for the m-Machine, n-Job Flowshop Sequencing Problem, *OMEGA*, 1:91-95, 1983.
- [43] J. O. McClain, J. Thomas, C. Sox, 'On-the-fly' line balancing with very little WIP. *International Journal of Production Economics*, 27:283-289, 1992.
- [44] D. S. Palmer, Sequencing Jobs Through a Multi-Stage Process in the Minimum Total Time – A Quick Method of Obtaining a Near Optimum, *Operational Research Society*, 1:101-107, 1965.
- [45] J. C. H. Pan, J. S. Chen, and C. M. Chao, Minimizing Tardiness in a Two Machine Flow Shop. *Computers and Operations Research*, 29:869-885, 2002.
- [46] Q. K. Pan, R. Ruiz, A Comprehensive and Evaluation of Permutation Flowshop Heuristics to Minimize Flowtime. *Computers and Operations Research*, 40:117-128, 2013.
- [47] S. Parveen. H. Ullah, Review on Job-Shop and Flow-Shop Scheduling using Multi Criteria Decision Making, *Journal of Mechanical Engineering*, 41:130-145, 2010.
- [48] M. Pinedo, Scheduling Theory, Algorithms and Systems, *Springer Science Business Media, Prentice Hall*, 2008.
- [49] B. Rekiek, P. de Lit, F. Pellichero, T. L'Eglise, E. Falkenauer, A. Delchambre, Applying the Equal Piles to Balance Assembly Lines. *Proceedings of the ISATP*, 399-404, 1999.
- [50] H. Röck, The Three-Machine No-Wait Flow Shop Is NP-Complete. *Journal of the Association for Computing Machinery*
- [51] N.H. Saadani, A. Guinet, M. Moalla, A travelling salesman approach to solve the F/noidle/Cmax Problem. *European Journal of Operational Research*, 161:11-20, 2005.
- [52] A. Ruiz-Torres, J. Ho, H. Ablanedo-Rosas, Makespan and Workstation Utilization Minimization in a Flowshop with Operations Flexibility. *Omega*, 39:273-282, 2011.

- [53] J. H. Saleh , G. M. and N. C. Jordan, Flexibility: a multidisciplinary literature review and a research agenda for designing flexible engineering systems. *Journal of Engineering Design*, 20(3):307-323, 2009.
- [54] M. E. Salveson, The Assembly Line Balancing Problem. *The Journal of Industrial Engineering* 6:18-25, 1955.
- [55] A. Scholl, C. Becker, State-of-the-art Exact and Heuristic Solution Procedures for Simple Assembly Line Balancing. *European Journal of Operational Research*, 168:666-693, 2006
- [56] A. Sethi, S. Sethi, Flexibility in Manufacturing: A Survey. *The International Journal of Flexible Manufacturing Systems*, 2:289-328, 1990.
- [57] N. Shakhlevich, H. Hoogeveen, M. Pinedo, Minimizing Total Weighted Completion Time in a Proportionate Flow Shop. *Journal of Scheduling*, 1:157-168, 1998.
- [58] P.M. Swamidass, Manufacturing Flexibility. *Monograph No. 2, Operations Management Association, Norman and Schneider Group, Waco, TX* , 1988.
- [59] C. Tang, B. Tomlin, The Power of Flexibility for Mitigating Supply Chain Risks. *International Journal of Production Economics*, 116:12-27, 2008.
- [60] H. F. Ugurdag, R. Rachamadugu, C. A. Papachristou, Designing Paced Assembly Lines with Fixed Number of Stations. *European Journal of operational Research*, 102:488-501, 1997.
- [61] G. Vairaktarakis, C. Lee, Analysis of Algorithms for Two-Stage Flowshops with Multi-Processor Task Flexibility. *Naval Research Logistics*, 51:44-59, 2004.
- [62] L. Vincent, S. Ponnambalam, Scheduling Flexible Assembly Lines Using Differential Evolution, *Swarm, Evolutionary, and Memetic Computing*, 7076:43-50, 2011.
- [63] T. S. Wee, M. J. Magazine, Assembly Line Balancing as Generalized Bin Packing. *Operations Research Letters*, 1/2:56-58, 1982.
- [64] D. A. Wismer, Solution of the Flowshop-Scheduling Problem with No Intermediate Queues. *Operations Research*, 20:689-697, 1972.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : SUCU, Seda
Uyruğu : T.C.
Doğum tarihi ve yeri : 25.08.1988 Samsun
Medeni hali : Bekar
Telefon : +905076617265
e-mail : ssucu@.etu.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Y. Lisans	TOBB Ekonomi ve Teknoloji Üniv.	2013
Lisans	Orta Doğu Teknik Üniversitesi	2011

İş Deneyimi

Yıl	Yer	Görev
20011-2013	TOBB Ekonomi ve Teknoloji Üniv.	Burslu Yüksek Lisans Öğr.

Yabancı Dil

İngilizce (Çok iyi)

Yayınlar

Sucu, S., Gültekin, H., "Scheduling in 3-machine Flowshops With Flexible Operations", 26th EURO-INFORMS Conference On Operational Research, Roma,

İtalya, Temmuz, 2013.

Sucu S., Gültekin H., "Part Scheduling And Operation Allocation In a Flowshop with Three Non-identical Machines", International IIE Conference, İstanbul, Türkiye, Haziran, 2013.