



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



BÜTÜNLEŞİK ÜRETİM ve DAĞITIM
ÇİZELGELEME PROBLEMLERİ İÇİN
ÇÖZÜM YAKLAŞIMLARI

Ece ÇETİN YAĞMUR

DOKTORA TEZİ

Endüstri Mühendisliği Anabilim Dalı

Mayıs-2021
KONYA
Her Hakkı Saklıdır

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Ece ÇETİN YAĞMUR
28.05.2021

ÖZET

DOKTORA TEZİ

BÜTÜNLEŞİK ÜRETİM ve DAĞITIM ÇİZELGELEME PROBLEMLERİ İÇİN ÇÖZÜM YAKLAŞIMLARI

Ece ÇETİN YAĞMUR

Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Endüstri Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Saadettin Erhan KESEN

2021, 130 Sayfa

Jüri

Prof. Dr. Saadettin Erhan KESEN

Prof. Dr. İsmail KARAOĞLAN

Prof. Dr. Selçuk Kürşat İŞLEYEN

Prof. Dr. Yakup KARA

Doç. Dr. Çağrı KOÇ

Günümüzde rekabetin işletmelerden çok işletmelerin tedarik zincirleri arasında gerçekleştiği düşünüldüğünde, tedarik zincirinde yer alan fonksiyonların etkin bir şekilde yönetilmesi işletmeler açısından oldukça önemsenmektedir. Üretim ve dağıtım, tedarik zincirinde yer alan iki temel fonksiyondur. Geleneksel yaklaşımda bu fonksiyonlar bireysel olarak değerlendirilmekte ve hiyerarşik olarak ele alınmaktadır. Hiyerarşik bir şekilde alınan bu kararlar sistemin bütünü açısından en iyiye garanti edememektedir. Bu nedenle tedarik zinciri açısından bütünsel en iyiye ulaşmak için üretim ve dağıtım operasyonlarının bütünleşik bir şekilde ele alınması gerekmektedir. Ekonomik avantajlarının yanı sıra gerçek hayatta üretim ve dağıtım operasyonlarının bütünleşik olarak ele alınmasını zorunlu kılan bazı durumlarla karşılaşılmaktadır. Çabuk bozulan ve zamana duyarlı ürünlerin üretimi ve dağıtımı bu uygulama alanlarından birisidir.

Yapılan tez çalışmasında operasyonel düzeyde üretim ve dağıtım operasyonlarının bütünleştirilmesi incelenmiştir. Üretim aşamasında üretim çizelgeleme problemini dağıtım aşamasında ise araç rotalama problemini bünyesinde barındıran bu problem literatürde “Bütünleşik Üretim ve Dağıtım Çizelgeleme (BÜDÇ) Problemi” olarak adlandırılmaktadır. Tez çalışmasında BÜDÇ kapsamında üç farklı problem ele alınmıştır. Permütasyon Akış Tipi Üretim Ortamı ile Tek Araçlı Dağıtım Ortamını içeren BÜDÇ (PAT_TA) Problemi olarak ifade edilen ilk problemde siparişlerin seri olarak yerleştirilmiş tüm makineleri aynı sırada ziyaret ettiği permütasyon akış tipi üretim ortamı ve sınırlı kapasiteye sahip tek araçla teslimatlarının yapıldığı bir dağıtım ortamı incelenmiştir. Permütasyon Akış Tipi Üretim Ortamı ile Çok Araçlı Dağıtım Ortamını içeren BÜDÇ (PAT_ÇA) Problemi olarak ifade edilen ikinci problemde ilk problemin daha genel bir hali dikkate alınarak aynı üretim ortamı ile heterojen kapasiteli birden fazla aracın olduğu dağıtım ortamı incelenmiştir. Son olarak; Atölye Tipi Üretim Ortamı ile Çok Araçlı Dağıtım Ortamını içeren BÜDÇ (AT_ÇA) Problemi olarak ifade edilen üçüncü problemde ise siparişlerin makinelerdeki operasyon sıralarının birbirinden farklı olduğu atölye tipi üretim ortamı ve heterojen kapasiteli birden fazla aracın bulunduğu dağıtım ortamı incelenmiştir. Her üç problemde de araç sayısının sınırlı olması nedeniyle bazı müşteri siparişleri belirtilen teslim tarihinden sonra teslim edilebilmekte, dolayısıyla bu siparişler için gecikmeler oluşmaktadır. PAT_TA ve PAT_ÇA Problemlerinde müşterilerin teslim tarihlerinin yeterince büyük olduğu durumda taşıma sürecinin mümkün olan en kısa sürede tamamlanması istenmektedir. Dolayısıyla PAT_TA ve PAT_ÇA Problemleri için amaç, toplam tur zamanı ile toplam gecikme toplamının en küçüklenmesi olarak ele alınmıştır. AT_ÇA Problemi ise çevresel

faktörler de probleme dahil edilerek araçlar tarafından havaya salınan toplam CO₂ emisyonu ve maksimum gecikmenin en küçüklenmesi şeklinde çok amaçlı bir yapıda incelenmiştir. Her üç problemde de sistemdeki araçlar planlama periyodu boyunca birden fazla kez kullanılabilir. Tez kapsamında incelenen problemler literatürde daha önce çalışılmamıştır ve ilk kez bu tez çalışmasında incelenmiştir.

PAT_TA Problemi için öncelikle problemin varsayımları ve kısıtları tanımlanarak Karma Tamsayılı Doğrusal Programlama (KTDP) modeli geliştirilmiştir. Ardından problemin NP-zor yapısı nedeniyle büyük boyutlu problemler için makul sürelerde optimal ya da optime yakın çözümlerin elde edilmesi amacıyla Memetik Algoritma (MA) yaklaşımı önerilmiştir. PAT_TA Problemi için önerilen çözüm yöntemlerinin performanslarını değerlendirmek amacıyla 1080 adet test problemi müşteri sayılarına göre küçük, orta ve büyük boyutlu olmak üzere oluşturulmuştur. İncelenen 1080 örneğin sadece 256 tanesinde CPLEX ile üç saat süre sınırı içerisinde optimal çözüm elde edilebilmiştir. Bu 256 örneğin 243 tanesinde MA ile optimal çözümlere saniyelerle ifade edilen oldukça kısa sürelerde ulaşılmıştır. Orta boyutlu problemlerde CPLEX ve MA'nın göreceli performansları birbirlerine oldukça yakinken çözüm süreleri değerlendirildiğinde MA'nın çözüm süresi açısından oldukça etkin olduğu görülmüştür. Büyük boyutlu problemlerde ise CPLEX hiçbir örnekte optimal çözüm elde edememiş bununla birlikte üç saatlik süre sınırı içerisinde CPLEX'in göreceli performansı MA'ya göre oldukça düşük bulunmuştur.

PAT_ÇA Problemi için geliştirilen Karma Tamsayılı Programlama (KTP) modeli doğrusal olmadığından model önce doğrusallaştırma kısıtları ile doğrusal hale getirilmiş ardından PAT_TA Problemi için geliştirilen MA yaklaşımı sistemde birden fazla araç olduğu için PAT_ÇA Problemine göre yeniden uyarlanmıştır. PAT_ÇA Problemi için geliştirilen 1980 test problemi üzerinde CPLEX ve MA'nın performansları değerlendirilmiştir. CPLEX, PAT_TA Problemi için tüm örneklerde üst sınır elde edebilirken PAT_ÇA Problemi için üretilen 720 adet büyük boyutlu örneğin sadece 388 tanesinde verilen süre sınırları içerisinde bir üst sınır elde edebilmiştir. Bununla birlikte PAT_TA Problemi için küçük boyutlu problemlerde CPLEX tüm örnekler için optimal çözümü elde edebilirken PAT_ÇA Problemi için 180 adet küçük boyutlu problemin 145 tanesinde optimal çözüm elde edebilmiştir. Sonuç olarak sistemdeki araç sayısının birden fazla olmasının problem karmaşıklığını artırdığı gözlenmiştir. PAT_ÇA Problemi için önerilen MA ile optime bilinen 145 örneğin hepsinde optimal değerler saniyelerle ifade edilen oldukça kısa sürelerde bulunmuştur. MA ile CPLEX'in göreceli performansları değerlendirildiğinde orta boyutlu problemlerde PAT_TA Problemi için yakın sonuçlar elde edilirken PAT_ÇA Problemi için önerilen MA 10 müşterinin olduğu durumda bile CPLEX'e göre göreceli sapma değeri açısından daha başarılı bulunmuş ve müşteri sayısının artmasıyla MA'nın CPLEX'e göre performansı büyük oranda artış göstermiştir. Geliştirilen MA en büyük boyutlu problemlerde bile bir dakikadan daha kısa bir sürede CPLEX'e göre büyük üstünlük sağlamıştır.

AT_ÇA Problemi çok amaçlı yapıda Karma Tamsayılı Doğrusal Programlama modeli (KTDP) ile formüle edilmiş ve problemin çözümü için kesin yöntem olarak Artırılmış Epsilon Kısıt Yöntemi (AEKY) uygulanmıştır. AEKY'nin problem boyutu arttıkça yetersiz gelmesi üzerine çok amaçlı optimizasyon problemleri için geliştirilen iki farklı sezgisel yöntem, Baskın Olmayan Sıralama Genetik Algoritma-II (BSGA-II) ve Pareto Yerel Arama (PYA), AT_ÇA Problemine uygulanmıştır. PYA yaklaşımı içerisinde, problemin üretim ortamında değişken komşu arama algoritması kullanılırken dağıtım aşamasında yerel arama yaklaşımına başvurulmuştur. AT_ÇA Problemi için rassal olarak üretilen 624 adet test problemi üzerinde önerilen algoritmaların hem kesin yöntemle hem de birbirleriyle karşılaştırılması yapılmıştır. Çok amaçlı optimizasyon performans kriterleri olarak pareto çözüm sayısı, hipervolum değeri ve ortalama çözüm süreleri kullanılmıştır. Sonuçlar değerlendirildiğinde küçük boyutlu örneklerde hem BSGA-II hem de PYA ile elde edilen Pareto yüzeyler AEKY ile elde edilen gerçek Pareto yüzeye oldukça yakın bulunmuştur. Orta ve büyük boyutlu problemlerde ise AEKY ile verilen süre sınırı içerisinde ödünleşim matrisi oluşturulmadığından yalnızca BSGA-II ve PYA'nın birbirine göre performansı değerlendirilmiştir. Sonuçlara göre her üç performans kriteri açısından da on dakikalık süre sınırı içerisinde elde edilen Pareto çözümlere göre BSGA-II, PYA'ya göre daha başarılı bulunmuştur. PYA'nın müşteri sayısındaki artışa oldukça duyarlı olduğu gözlenerek PYA bir saat çalıştırılmıştır. Elde edilen sonuçlar incelendiğinde çözüm süresinin artırılmasının PYA performansını ciddi oranda artırdığı hatta bazı örneklerde PYA'nın BSGA-II'yi geçtiği görülmüştür.

Anahtar Kelimeler: Artırılmış Epsilon Kısıt, Baskın Olmayan Sıralama Genetik Algoritma-II, Bütünleşik Üretim ve Dağıtım Çizelgeleme, Çok Amaçlı Optimizasyon, Karma Tamsayılı Programlama, Memetik Algoritma, Pareto Yerel Arama, Teslim Tarihi.

ABSTRACT

PhD THESIS

SOLUTION APPROACHES FOR INTEGRATED PRODUCTION AND DISTRIBUTION SCHEDULING PROBLEMS

Ece ÇETİN YAĞMUR

**Konya Technical University
Institute of Graduate Studies
Department of Industrial Engineering**

Advisor: Prof. Dr. Saadettin Erhan KESEN

2021, 130 Pages

Jury

Prof. Dr. Saadettin Erhan KESEN

Prof. Dr. İsmail KARAOĞLAN

Prof. Dr. Selçuk Kürşat İŞLEYEN

Prof. Dr. Yakup KARA

Assoc. Prof. Dr. Çağrı KOÇ

Considering that competition takes place between the supply chains of businesses rather than businesses today, effective management of functions in the supply chain is highly important for businesses. Production and distribution are two main functions in the supply chain. In the traditional approach, these functions are evaluated individually and considered hierarchically. These hierarchically made decisions cannot guarantee optimal in terms of the system. For this reason, production and distribution operations should be handled in an integrated manner to achieve the global optimal in terms of the supply chain. In addition to its economic advantages, it has been inevitable for some cases to consider production and distribution operations in an integrated manner in the supply chain. There are many application areas where production and distribution operations are handled in an integrated manner in real life. Production and distribution of perishable and time-sensitive products is one of these application areas.

In the thesis, the integration of production and distribution operations at the operational level has been examined. This problem, which includes the production scheduling problem in the production phase, and the vehicle routing problem in the distribution phase, is called as “Integrated Production and Distribution Scheduling (IPDS) Problem” in the literature. In the thesis study, three different problems have been addressed within the scope of IPDS. In the problem expressed as the Permutation Flow Shop Production Environment and the Single Vehicle Distribution Environment (PFS-SV) IPDS Problem, a permutation flow shop production environment where orders visit all machines placed in series in the same order and a distribution environment where orders are delivered with a single vehicle with limited capacity are analyzed. In the second problem expressed as the Permutation Flow Shop Production Environment and the Multi Vehicle Distribution Environment (PFS-MV) IPDS Problem, the same production environment and the distribution environment with more than one vehicle with heterogeneous capacity are examined by taking into account the more general form of the first problem. Finally, in the last problem expressed as the Job Shop Production Environment and the Multi Vehicle Distribution Environment (JS-MV) IPDS Problem, the job shop production environment where the operation sequences of the orders are different from each other and the distribution environment with multi vehicles with heterogeneous capacity are examined. Due to the limited number of vehicles in all three types of problems, some customer orders may be delivered after the predefined due date, thus there is tardiness for these orders. In PFS-SV and PFS-MV Problems, when the due dates of the customers are large enough, the distribution process is requested to be completed as soon as possible. Therefore, the objective for PFS-SV and PFS-MV Problems are considered

minimizing the sum of total tour time and total tardiness. JS-MV Problem, on the other hand, has been investigated in a multi-objective structure by including environmental factors into the problem, as minimizing the total CO₂ emission released into the air by vehicles and minimizing the maximum tardiness. In all three problems, the vehicles in the system can be used more than once during the planning period. All three problems addressed within the scope of the thesis have not been encountered before in the literature and have been examined for the first time in this thesis study.

For PFS-SV Problem, a Mixed Integer Linear Programming (MILP) model has been developed by defining the assumptions and constraints of the problem. Due to the NP-hard structure of the problem, the Memetic Algorithm (MA) approach has been proposed to obtain optimal or near-optimal solutions in a reasonable time for big-sized problems. 1080 test instances have been created in small, medium, and big-sized according to the number of customers to evaluate the performance of the solution methods proposed for PFS-SV Problem. In only 256 of the 1080 instances examined, the optimal solution can be obtained with CPLEX within three hours. 243 of these 256 instances, optimal solutions are reached by MA in a very short time expressed in seconds. While relative performances of CPLEX and MA are very close to each other in medium-sized problems, when the solution times are evaluated, it has been seen that MA is very effective in terms of solution time. However, in big-sized problems, CPLEX cannot obtain an optimal solution in any instance and the relative performance of CPLEX within the three-hour time limit is found to be significantly lower than MA.

Since the Mixed Integer Programming (MIP) model developed for PFS-MV Problem is not linear, the model is first linearized with linearization constraints, then the MA approach developed for PFS-SV Problem is adapted according to PFS-MV Problem because there are multi vehicles in the system. The performance of CPLEX and MA are evaluated on the 1980 test instances developed for PFS-MV Problem. While CPLEX can obtain an upper limit in all instances for PFS-SV Problem, for PFS-MV Problem it can obtain an upper limit for only 388 of 720 big-sized instances within the given time limits. Besides, for PFS-SV Problem, CPLEX was able to obtain the optimal solution for all examples, while for PFS-MV Problem it can obtain the optimal solution for 145 of 180 small-sized problems. As a result, it has been observed that the number of vehicles in the system is more than one increases the complexity of the problem. With the proposed MA for PFS-MV Problem, the optimal values in all 145 instances who's optimal are known were found in a very short time, expressed in seconds. When the relative performances of MA and CPLEX are evaluated, close results have been obtained for PFS-SV in medium-sized problems, while the proposed MA for PFS-MV Problem is found to be more successful in terms of relative performance compared to CPLEX even in the case of 10 customers, and with the increase in the number of customers, MA's relative performance to CPLEX has increased substantially. The developed MA outperformed CPLEX even in the biggest problems in less than a minute.

JS-MV Problem has been formulated with a Multi-Objective Mixed Integer Programming model and the Augmented Epsilon Constraint Method (AUGMECON) is applied as the exact method to solve the problem. As the AUGMECON method has become insufficient as the problem size increased, two different heuristic methods, Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) and Pareto Local Search (PLS) have developed for multi-objective optimization problems are applied to JS-MV Problem. In the PLS algorithm, while the variable neighbor search algorithm is applied in the production phase of the problem, the local search approach is used in the distribution phase. The algorithms proposed for JS-MV Problem are compared on 624 randomly generated test instances both with the exact method and with each other. The number of pareto solutions, hypervolume values, and solution times are used as the multi-objective optimization performance criteria. When the results are evaluated, the Pareto fronts obtained with both NSGA-II and PLS approach in small-sized instances are found to be very close to the real Pareto front obtained by AUGMECON. In medium and big-sized problems, since the trade-off matrix cannot be created within the time limit given by AUGMECON, only the relative performance of NSGA-II and PLS to each other has been evaluated. According to the results, NSGA-II is found to be more successful than the PLS compared to the Pareto solutions obtained within the ten-minute time limit in terms of all three-performance criteria. It has been observed that the PLS is very sensitive to the increase in the number of customers, so the PLS is also run for one hour. When the results obtained are examined, it is seen that increasing the solution time significantly increased the performance of the PLS, even in some cases, the PLS has outperformed NSGA-II.

Keywords: Augmented Epsilon Constraint, Due Date, Integrated Production and Distribution Scheduling, Memetic Algorithm, Mixed Integer Programming, Multi-Objective Optimization, Non-Dominated Sorting Genetic Algorithm-II, Pareto Local Search.

ÖNSÖZ

Doktora sürecim boyunca akademik açıdan bilgi birikimi ve tecrübelerini her fırsatta benimle paylaşan, tezimin her aşamasında aynı motivasyonla çalışmamı sağlayarak gelişmeme sürekli katkıda bulunan değerli hocam ve tez danışmanım Prof. Dr. Saadettin Erhan KESEN' e sonsuz teşekkürlerimi sunarım.

Kıymetli fikir ve önerilerinden dolayı tez izleme komitesi üyesi hocalarım Sayın Prof. Dr. İsmail KARAOĞLAN ve Sayın Prof. Dr. Selçuk Kürşat İŞLEYEN' e, doktora eğitimim boyunca bölümde her türlü fiziksel ve sosyal olanağı sağlayarak bizlere huzurlu bir çalışma ortamı sunan, başta Endüstri Mühendisliği Bölüm Başkanı Sayın Prof. Dr. Ahmet PEKER olmak üzere tüm bölüm hocalarıma ve meslektaşlarıma teşekkürlerimi sunarım.

Bu çalışmayı, beni bugünlere getiren ve üzerimde sonsuz emekleri olan değerli aileme, çalışmalarım boyunca sabır ve anlayışla beni her koşulda destekleyen değerli eşim Sercan YAĞMUR' a ve bensiz geçen zamanlarının özrü olarak canım oğlum Çınar' a ithaf ediyorum.

Ece ÇETİN YAĞMUR
KONYA-2021

İÇİNDEKİLER

ÖZET	v
ABSTRACT.....	vii
ÖNSÖZ	ix
İÇİNDEKİLER	x
SİMGELER VE KISALTMALAR	xii
1. GİRİŞ	1
2. KAYNAK ARAŞTIRMASI	7
2.1. Tek Makineli Üretim Ortamı için BÜDÇ Literatürünün Değerlendirilmesi	11
2.2. Paralel Makineli Üretim Ortamı için BÜDÇ Literatürünün Değerlendirilmesi ..	14
2.2.1. Özdeş paralel makineli üretim ortamı için BÜDÇ literatürünün değerlendirilmesi	14
2.2.2. Düzgün paralel makineli üretim ortamı için BÜDÇ literatürünün değerlendirilmesi	15
2.2.3. İlişkisiz paralel makineli üretim ortamı için BÜDÇ literatürünün değerlendirilmesi	15
2.3. Akış Tipi Üretim Ortamı için BÜDÇ Literatürünün Değerlendirilmesi.....	16
2.4. Atölye Tipi Üretim Ortamı için BÜDÇ Literatürünün Değerlendirilmesi	16
2.5. BÜDÇ Literatürünün Genel Değerlendirilmesi	16
3. PERMÜTASYON AKIŞ TİPİ ÜRETİM ORTAMI İLE TEK ARAÇLI DAĞITIM ORTAMINI İÇEREN BÜDÇ (PAT_TA) PROBLEMİ.....	19
3.1. PAT_TA Probleminin Tanımı	19
3.1.1. PAT_TA Problemi için önerilen KTDP modeli	20
3.1.2. PAT_TA Problemi için açıklayıcı örnek	23
3.2. PAT_TA Problemi için Materyal ve Yöntem.....	25
3.2.1. Memetik Algoritmalar	25
3.2.2. PAT_TA Problemi için önerilen MA	27
3.2.3. Algoritma adımlarının örnek test verisine uygulanması.....	33
3.2.4. PAT_TA Problemi için test problemlerinin üretilmesi.....	39
3.3. PAT_TA Problemi için Karşılaştırmalı Sonuçlar	40
3.3.1. Önerilen KTDP modelinin sonuçları	41
3.3.2. Önerilen MA'nın performansı	43
4. PERMÜTASYON AKIŞ TİPİ ÜRETİM ORTAMI İLE ÇOK ARAÇLI DAĞITIM ORTAMINI İÇEREN BÜDÇ (PAT_ÇA) PROBLEMİ.....	47
4.1. PAT_ÇA Probleminin Tanımı	47
4.1.1. PAT_ÇA Problemi için önerilen KTP modeli	48
4.1.2. PAT_ÇA Problemi için açıklayıcı örnek	52
4.2. PAT_ÇA Problemi için Materyal ve Yöntem.....	55

4.2.1. PAT_ÇA Problemi için önerilen MA	55
4.2.2. Algoritma adımlarının örnek test verisine uygulanması.....	57
4.2.3. PAT_ÇA Problemi için test problemlerinin üretilmesi	62
4.3. PAT_ÇA Problemi için Karşılaştırmalı Sonuçlar.....	63
4.3.1. Önerilen KTP modelinin sonuçları	63
4.3.2. Önerilen MA'nın performansı	65
5. ATÖLYE TİPİ ÜRETİM ORTAMI İLE ÇOK ARAÇLI DAĞITIM ORTAMINI İÇEREN BÜDÇ (AT_ÇA) PROBLEMİ	69
5.1. AT_ÇA Probleminin Tanımı	69
5.1.1. AT_ÇA Problemi için önerilen çok amaçlı KTDP modeli.....	70
5.1.2. AT_ÇA Problemi için açıklayıcı örnek	74
5.2. AT_ÇA Problemi için Materyal ve Yöntem.....	77
5.2.1. Artırılmış epsilon kısıt yöntemi	77
5.2.1. AEKY'nin örnek test verisine uygulanması	79
5.2.3. Pareto yerel arama	81
5.2.4. PYA yaklaşımının örnek test verisine uygulanması	86
5.2.3. Baskın olmayan sıralama genetik algoritma-II.....	88
5.2.4. BSGA-II yaklaşımının örnek test verisine uygulanması	92
5.2.5. AT_ÇA Problemi için test problemlerinin üretilmesi.....	98
5.3. AT_ÇA Problemi için Karşılaştırmalı Sonuçlar.....	100
5.3.1. Önerilen çok amaçlı KTDP modelinin sonuçları.....	101
5.3.2. Sezgisel algoritmaların değerlendirilmesi	102
6. SONUÇLAR VE TARTIŞMA.....	108
6.1 Sonuçlar	108
6.2 Tartışma	112
KAYNAKLAR	114

SİMGELER VE KISALTMALAR

Kısaltmalar

AEKY	: Artırılmış Epsilon Kısıt Yöntemi
AT_ÇA	: Atölye Tipi Üretim Ortamı ile Çok Araçlı Dağıtım Ortamını içeren BÜDÇ
BSGA-II	: Baskın Olmayan Sıralama Genetik Algoritma-II
BÜDÇ	: Bütünleşik Üretim ve Dağıtım Çizelgeleme
GPÇ	: Genelleştirilmiş Parça Çaprazlama
GSC	: Genelleştirilmiş Sıra Çaprazlama
GSD	: Görelî Sapma Değeri
HV	: Hipervolüm
KTDP	: Karma Tamsayılı Doğrusal Programlama
KTP	: Karma Tamsayılı Programlama
MA	: Memetik Algoritma
OÇS	: Ortalama Çözüm Süresi
OÇUPS	: Optimal Çözüme Ulaşılan Problem Sayısı
ÖKÇ	: Öncelik Koruyucu Çaprazlama
PAT_TA	: Permütasyon Akış Tipi Üretim Ortamı ile Tek Araçlı Dağıtım Ortamını içeren BÜDÇ
PAT_ÇA	: Permütasyon Akış Tipi Üretim Ortamı ile Çok Araçlı Dağıtım Ortamını içeren BÜDÇ
PÇS	: Pareto Çözüm Sayısı
PYA	: Pareto Yerel Arama
TZÜ	: Tam Zamanında Üretim
YSD	: Yüzde Sapma Değeri

1. GİRİŞ

Günümüzde işletmeler, artan rekabet ortamının etkisiyle içerisinde bulundukları ekonomik ve sosyal çevrede mevcut konumlarından daha iyi konumlara ulaşma arzusu içerisindeyler. Bu hedefe ulaşmak için, kaynakların etkin bir şekilde kullanılması, sürece değer katmayan faaliyetlerin en aza indirgenmesi ve süreç çıktılarının en verimli şekilde talep merkezlerine ulaştırılması gerekmektedir. Bu stratejilerin eş zamanlı olarak uygulanmasıyla işletmeler hedeflerine giden yolda üstel bir başarı elde edebilmekte ancak bu durum güçlü bir koordinasyonun gerekliliğini de beraberinde getirmektedir. Özellikle rekabetin işletmelerden çok işletmelerin tedarik zincirleri arasında gerçekleştiği düşünüldüğünde, tedarikçiden nihai tüketiciye kadar uzanan bu ağda etkin bir koordinasyonun gerekliliği yadsınamaz bir gerçek olarak karşımıza çıkmaktadır.

Thomas ve Griffin (1996) tedarik zincirinde yer alan temel fonksiyonları “tedarik”, “üretim” ve “dağıtım” olarak tanımlamışlar ve bu fonksiyonların koordinasyonu adına yapılan çalışmaları stratejik ve operasyonel düzeyde incelemişlerdir. Stratejik düzeyde alınan kararlar uzun vadeli olup tedarik zincirinin genel yapılandırılması ve tasarımı ile ilgilirken, operasyonel düzey kararlar kısa vadeli olup günlük faaliyetlere odaklanmaktadır. Tedarik zincirinde yer alan üretim ve dağıtım fonksiyonlarının bütünleştirilmesi adına yapılan çalışmaların büyük çoğunluğunun stratejik ve taktik düzey kararlar üzerinde yoğunlaştığı görülmektedir (Z.-L. Chen, 2010). Buna karşın operasyonel düzeyde üretim ve dağıtım operasyonları genellikle birbirinden bağımsız olarak ele alınmaktadır.

Operasyonel düzeyde, üretim aşamasında, taktik düzeyde hazırlanan üretim planları dikkate alınarak üretimi yapılacak olan ürünlerin hangi kaynaklarda, hangi sırada işlem göreceği kararları verilmektedir. Dağıtım aşamasında ise müşterilere hangi araçlar tarafından hangi turlarda, hangi sırada hizmet verileceği belirlenmektedir. Bu kararlar alınırken bütünleşik olmayan geleneksel yaklaşımda öncelikle üretim çizelgeleri belirlenmekte ardından bu çizelgelere bağlı kalınarak ürünlerin üretim tamamlanma zamanlarına göre dağıtım planları oluşturulmaktadır. Tam tersi şekilde, önce dağıtım planlarının oluşturulup ardından bu planlardaki dağıtıma başlama zamanlarına göre üretim çizelgelerinin belirlenmesi de mümkündür. Ancak bu şekilde tek bir problemin en iyilenmesiyle diğer problemin kısıtlamaları ve gereksinimleri göz ardı edilmektedir. Sonuç olarak geleneksel yaklaşımda verilen kararlar bireysel fonksiyonlar açısından

tatmin edici olsa da sistemin bütünü açısından en iyi garanti edememektedir. Buna karşın üretim ve dağıtım operasyonlarının eş zamanlı olarak ele alındığı bütünleşik bir yaklaşım ile yüksek performans gösteren genel bir sistem elde edilebilir. Operasyonel düzeyde üretim ve dağıtım fonksiyonlarını bütünleştirmenin bütünleşik olmayan yaklaşıma göre ortalama %5 ile %20 oranında bir iyileşme ile sonuçlandığı çeşitli kaynaklarda belirtilmiştir (Z.-L. Chen ve Vairaktarakis, 2005; Park ve Hong, 2009; Ullrich, 2013, Yağmur ve Kesen, 2020).

Üretim ve dağıtım operasyonlarının bütünleşik olarak ele alınmasının işletmelere sağlayacağı bir diğer avantaj bütünleşik yaklaşımın, bu fonksiyonlar arasında stok tutma zorunluluğunu ortadan kaldırmasıdır. Özellikle siparişe göre üretim yapan işletmelerde ürünler müşteri ihtiyaç ve beklentilerine göre düzenlenebildiğinden üretim ancak talep geldiği anda başlayabilmektedir. Bu tarz işletmelerde, üretim ve dağıtım fonksiyonları arasında nihai ürün stoğu bulunmadığından, firmaların üretim ve dağıtım sürecinde geçen zamanı etkin bir şekilde yönetmeleri gerekmektedir. Tam Zamanında Üretim (TZÜ) felsefesine göre çalışan işletmelerde bütünleşik yaklaşımın uygulanmasıyla üretim ve dağıtım fonksiyonları etkin bir şekilde yönetilebilmekte, böylece hem müşteri hizmet düzeyi artırılmakta hem de stoksuz çalışmanın avantajlarından faydalanılmaktadır.

Ekonomik açıdan sağladığı avantajların yanı sıra bütünleşik yaklaşımın kullanılmasını zorunlu kılan durumlarla da karşılaşmaktadır. Bu uygulama alanlarından biri zamana duyarlı ürünlerin üretimi ve dağıtımıdır. Çabuk bozulan gıda ürünleri (H.-K. Chen ve ark., 2009; Farahani ve ark., 2012), hazır beton karışımları (Garcia ve ark., 2004; Naso ve ark., 2007), endüstriyel yapıştırıcı malzemeler (Geismar ve ark., 2008; Viergutz ve Knust, 2014) ve bazı ilaçlar (Lee ve ark., 2014) bu tarz ürünlere örnek olarak verilebilir. Örneğin, beton karışımı için gereken hammaddeler karıştırıldıktan bir saat gibi kısa bir süre sonra katılaşır ve kullanılamaz hale gelir. Benzer şekilde bazı yapıştırıcı malzemeler yapıştırıcı özelliğini bir hafta içerisinde kaybeder. Bu nedenle bu tarz ürünler üretildikten hemen sonra tüketicilere ulaştırılmak zorundadır. Bir diğer zamana duyarlı ürün grubu ise gazetelerin basılması ve dağıtılmasıdır (Chiang ve ark., 2009; Hurter ve Van Buer, 1996; Russell ve ark., 2008; Van Buer ve ark., 1999). Benzer şekilde, genellikle gece yarısından sonra basılmaya başlanan günlük gazeteler en kısa sürede araçlara yüklenip ertesi gün gazete bayilerine teslim edilmek üzere yola çıkmak zorundadır.

Bu örnekler göz önüne alındığında üretim ve dağıtım çizelgelerinin bütünleşik bir şekilde hazırlanması birçok sektörde kaçınılmaz derecede önemli hale gelmiştir. Böylece

literatürde Bütünleşik Üretim ve Dağıtım Çizelgeleme Problemi (BÜDÇ) olarak adlandırılan yeni bir problem ortaya çıkmıştır. Z.-L. Chen (2010) operasyonel düzeyde üretim ve dağıtım fonksiyonlarının bütünleştirilmesi adına literatürde yapılan çalışmaları dağıtım yöntemlerine göre sınıflandırmıştır. Bu çalışmaların büyük bir kısmının görece olarak basit teslimat yöntemlerini dikkate aldığı görülmektedir. Bu teslimat yöntemleri arasında müşterilere doğrudan teslimatın yapıldığı durumlar (Hall ve Potts, 2003; Potts, 1980), sabit müşteri rotasının kullanıldığı durumlar (Armstrong ve ark., 2008; Gupta ve ark., 2012) ve üçüncü parti lojistik şirketlerinde sıklıkla rastlanan sabit ayrılış zamanlı teslimatlar (C. L. Li ve Ou, 2005; Stecké ve ark., 2007; J. Zhang ve ark., 2016) örnek olarak verilebilir.

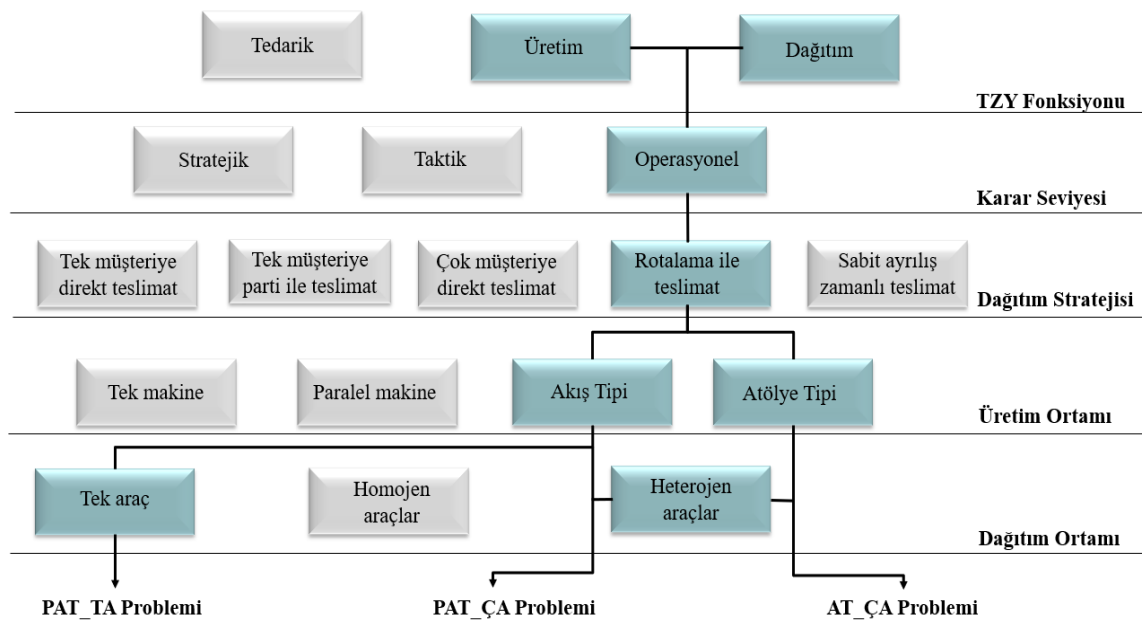
Yapılan tez çalışmasının odak noktası ise rotalama kararlarının verildiği operasyonel düzey BÜDÇ problemleridir. Bu sınıfta yer alan problemlerde teslimat turlarında birden fazla müşterinin ziyaret edilmesine izin verilmektedir. Bu nedenle müşterilerin hangi sırada ziyaret edileceğini de içeren her bir tur için bir rota bulunmasına gereksinim duyulmaktadır. Böyle bir rotalama problemi NP-zor yapıda olan gezgin satıcı probleminin özel bir durumu olarak çözümü zor problemler sınıfındadır. Bu nedenle birden fazla müşteriye rotalama yöntemiyle teslimatın yapıldığı problemler BÜDÇ problemleri içerisinde çözümü en zor olan problemlerdir.

Yapılan tez çalışmasında üretim çizelgeleme ve araç rotalama problemlerinin kısıtları ve gereksinimleri bütünleşik sisteme ilişkin en iyi çözümün elde edilmesi açısından eş zamanlı olarak dikkate alınmıştır. Tez kapsamında literatürde daha önce çalışılmamış BÜDÇ problemleri incelenmiştir.

Permütasyon Akış Tipi Üretim Ortamı ile Tek Araçlı Dağıtım Ortamını içeren BÜDÇ (PAT_TA) Problemi olarak ifade edilen ilk problemde siparişlerin seri olarak yerleştirilmiş tüm makineleri aynı sırada ziyaret ettiği akış tipi üretim ortamı ve sınırlı kapasiteye sahip tek araçla siparişlerin teslimatlarının yapıldığı bir dağıtım ortamı incelenmiştir. Permütasyon Akış Tipi Üretim Ortamı ile Çok Araçlı Dağıtım Ortamını içeren BÜDÇ (PAT_ÇA) Problemi olarak ifade edilen ikinci problemde ilk problemin daha genel bir hali dikkate alınarak aynı üretim ortamı ile heterojen kapasiteli birden fazla aracın olduğu dağıtım ortamı incelenmiştir. Son olarak, Atölye Tipi Üretim Ortamı ile Çok Araçlı Dağıtım Ortamını içeren BÜDÇ (AT_ÇA) Problemi olarak ifade edilen son problemde ise siparişlerin makinelerdeki operasyon sıralarının birbirinden farklı olduğu atölye tipi üretim ortamı ve heterojen kapasiteli birden fazla aracın bulunduğu dağıtım ortamı incelenmiştir. Her üç problemde de araç sayısının sınırlı olması nedeniyle bazı

müşteri siparişleri belirtilen teslim tarihinden sonra teslim edilebilmekte, dolayısıyla bu siparişler için gecikmeler oluşmaktadır. PAT_TA ve PAT_ÇA Problemlerinde müşterilerin teslim tarihlerinin yeterince büyük olduğu durumda taşıma sürecinin mümkün olan en kısa sürede tamamlanması istenmektedir. Dolayısıyla PAT_TA ve PAT_ÇA Problemleri için amaç, toplam tur zamanı ile toplam gecikme toplamının en küçüklenmesi olarak ele alınmıştır. AT_ÇA Problemi ise çevresel faktörler de probleme dahil edilerek araçlar tarafından havaya salınan toplam CO₂ emisyonu ve maksimum gecikmenin en küçüklenmesi şeklinde çok amaçlı bir yapıda incelenmiştir.

İncelenen problemlerin özelliklerine ilişkin şematik gösterim Şekil 1.1’de verilmiştir. Buna göre Tedarik Zinciri Yönetimi (TZY) fonksiyonu olarak üretim ve dağıtım ortamlarının koordinasyonu ele alınmış, bu fonksiyonların karar seviyesi açısından operasyonel düzeyde bütünleştirilmesi incelenmiştir. İncelenen problemler dağıtım stratejisi olarak rotalama kararlarının alındığı BÜDÇ problemleri sınıfına girmektedir. Tez kapsamında incelenen problemlerin üretim ve dağıtım ortamları da yine Şekil 1.1’de görülmektedir.



Şekil 1.1. İncelenen problemlerin şematik gösterimi

Z.-L. Chen (2010) yapmış olduğu çalışmada, makine çizelgeleme problemlerinden esinlenerek, BÜDÇ problemlerinin gösterimini standartlaştırmak amacıyla $(\alpha | \beta | \pi | \delta | \gamma)$ şeklindeki beşli gösterimi önermiştir. Bu gösterimde, α , β ve γ , çizelgeleme problemleri için geliştirilen klasik gösterimle aynı olup α üretim

sistemlerindeki makine ortamını, β siparişlerle ilgili özel durum ve kısıtları, γ modelde kullanılan amaç fonksiyonunu ifade etmektedir. π , teslimat süreciyle ilgili karakteristikleri ifade ederken, δ ise müşteri sayısını temsil etmektedir. Tez kapsamında incelenen problemlerin gösteriminde kullanılan simgeler ve açıklamaları Çizelge 1.1’de verilmiştir.

Çizelge 1.1. İncelenen problemlerin gösteriminde kullanılan simgeler

Simge	Açıklama
F_m	: Akış tipi üretim ortamı
J_m	: Atölye tipi üretim ortamı
$prmu$	: Permütasyon akış tipi üretim ortamı
dd_i	: Her bir i siparişine ait en son teslim tarihinin olduğu durum
$V(1, Q)$	: Sistemde Q kapasiteye sahip tek aracın olduğu durum
$V(v, Q_k)$	: Sistemde heterojen kapasiteli (Q_k) v adet aracın olduğu durum
$routing$	: Farklı müşterilere ait siparişlerin aynı turda teslim edilebildiği durum
n	: Her siparişin farklı bir müşteriye ait olduğu durum
$\sum T_i + \sum \sum t_{ij}$	: Toplam gecikme ile toplam tur zamanı toplamının en küçüklenmesi
$\sum CO_2, T_{max}$	: Toplam CO ₂ emisyonu ve maksimum gecikmenin en küçüklenmesi

Buna göre incelenen problemlerin gösterimi şu şekildedir:

- (i) PAT_TA Problemi: $F_m | prmu, dd_i | V(1, Q), routing | n | \sum T_i + \sum \sum t_{ij}$
- (ii) PAT_ÇA Problemi: $F_m | prmu, dd_i | V(v, Q_k), routing | n | \sum T_i + \sum \sum t_{ij}$
- (iii) AT_ÇA Problemi: $J_m | dd_i | V(v, Q_k), routing | n | \sum CO_2, T_{max}$

Yapılan tez çalışmasında incelenen problemler için öncelikle problemlerin kısıt ve varsayımları ifade edilerek matematiksel modeller geliştirilmiştir. Hem makine çizelgeleme hem de araç rotalama problemi NP-zor problem sınıfında yer aldığından bütünleşik problemin de NP-zor yapıda olduğu söylenebilir. Bu nedenle büyük boyutlu problemlerde makul sürelerde optimal ya da optimale yakın çözümler elde edilebilmesi amacıyla PAT_TA ve PAT_ÇA Problemleri için Memetik Algoritma (MA) yaklaşımına başvurulmuştur. AT_ÇA Problemi ise birbiriyle çelişen iki amaçlı bir yapıya sahip olduğundan küçük boyutlu örnekler için Artırılmış Epsilon Kısıt Yöntemi (AEKY) kullanılmış, büyük boyutlu örneklerde ise makul sürelerde Pareto çözümlerin elde edilebilmesi amacıyla Pareto Yerel Arama (PYA) ve Baskın Olmayan Sıralama Genetik Algoritma-II (BSGA-II) yaklaşımlarına başvurulmuştur.

Tez çalışmasının diğer bölümleri şu şekilde düzenlenmiştir. Çalışmanın ikinci bölümünde BÜDÇ problemi ile ilgili literatürde yer alan çalışmalar detaylı olarak incelenmiş ve literatürdeki boşlukların genel bir değerlendirilmesi yapılmıştır. Üçüncü dördüncü ve beşinci bölümde sırasıyla tez çalışması kapsamında incelenen PAT_TA,

PAT_ÇA ve AT_ÇA Problemleri ele alınmıştır. Bu bölümlerde öncelikle problem tanımları verilmiş, problemler matematiksel olarak ifade edildikten sonra her bir problemin çözümüne ilişkin önerilen yöntemler ve bu yöntemlerin performanslarını değerlendirmek amacıyla karşılaştırmalı sonuçlar verilmiştir. Her üç problem de incelendikten sonra son bölümde ise tez çalışması kapsamında elde edilen bulguların genel bir değerlendirmesi yapılarak, gelecekte yapılması planlanan çalışmalara ışık tutabilecek öneriler yer almaktadır.



2. KAYNAK ARAŞTIRMASI

Bu bölümde rotalama kararlarının verildiği operasyonel düzey BÜDÇ problemleri ile ilgili literatürde yapılan çalışmalara yer verilmiştir. Öncelikle literatürdeki BÜDÇ problemleri Çizelge 2.1’de verilen üretim ve dağıtım özellikleri ile amaç fonksiyonlarına göre kronolojik bir şekilde incelenmiştir. Çizelge 2.1’in ilk sütununda görüldüğü gibi incelenen kaynaklarda ele alınan problemlerin özellikleri üretim ortamı, dağıtım ortamı ve amaç fonksiyonu olmak üzere üç başlıkta incelenmiştir. İkinci sütunda ilk sütunda yer alan temel özelliklere ait olası durumlar verilirken son sütunda bu durumlara ilişkin açıklamalar yer almaktadır. Buna göre Çizelge 2.1’de verilen durumlara göre ilgili literatür kronolojik bir şekilde Çizelge 2.2’de incelenmiştir. Bu şekilde literatürdeki boşlukların net bir şekilde görülmesi sağlanmıştır. Ardından her bir üretim ortamına (tek makine, paralel makine, akış tipi ve atölye tipi) ilişkin literatür ayrıntılı bir şekilde incelenmiş ve son olarak BÜDÇ literatürüne ilişkin genel bir değerlendirme yapılmıştır.

Çizelge 2.1. Kaynak araştırmasında dikkate alınan özellikler

Özellik	Durum	Açıklama
Üretim Ortamı	İşletme Sayısı	: Üretim ve dağıtım fonksiyonlarının gerçekleştirildiği işletme sayısı
	Makine Ortamı	: Üretim aşamasında kullanılan makine ortamı
Dağıtım Ortamı	Araç Filosu	: Dağıtım aşamasında kullanılan araç sayısı ve özellikleri
	Çoklu Turlar	: Planlama döneminde araçların birden fazla kez kullanım durumu
Amaç	Maliyet	: Üretim ve dağıtım süreçlerinde oluşan maliyetlerin en küçüklenmesi
	Kar	: Gelir ve maliyet bazlı fonksiyonların birlikte yer aldığı problemlerde karın en büyüklenmesi
	Hizmet	: Zamana bağlı amaç fonksiyonları ile ölçülen müşteri hizmet düzeyinin en büyüklenmesi
	Karşılanan Talep Miktarı	: Kaynak kapasitesinden dolayı tüm siparişlerin karşılanamadığı durumlarda karşılanan talep miktarının en büyüklenmesi
	Kullanılan Araç Sayısı	: Araç kullanımına ilişkin sabit maliyetlerin olduğu durumlarda kullanılan araç sayısının en küçüklenmesi
	Kat edilen mesafe	: Araçların dağıtım aşamasında kat ettikleri toplam mesafenin en küçüklenmesi
	Kalite	: Özellikle çabuk bozulan ürün gruplarında teslimat zamanına göre üründe meydana gelen deformasyonun en küçüklenmesi
	Çevre	: Çevresel faktörler açısından araçların yakıt tüketimi veya emisyon miktarının en küçüklenmesi

Çizelge 2.2. BÜDÇ problemlerine ilişkin literatür özeti

Kaynak	Üretim Ortamı						Dağıtım Ortamı						Amaç							
	Tek depo	Çok depo	Tek makine	Paralel makine	Akış Tipi	Atölye Tipi	Tek araç	Homojen araçlar	Heterojen araçlar	Sınırsız sayıda araç	Sınırlı sayıda araç	Çoklu turlar	Maliyet	Kar	Hizmet	Karşılanan talep miktarı	Kullanılan araç sayısı	Kat edilen mesafe	Kalite	Çevre
Hurter ve Van Buer (1996)	•		•					•			•		•							
Van Buer ve ark. (1999)	•		•					•			•	•	•							
Chang ve Lee (2004)	•		•				•					•			•					
Garcia ve ark. (2004)		•	•					•			•			•						
Z.-L. Chen ve Vairaktarakis (2005)	•		•	•				•		•			•		•					
C.-L. Li ve ark. (2005)	•		•				•					•			•					
Naso ve ark. (2007)		•	•					•		•			•							
Geismar ve ark. (2008)	•		•				•					•			•					
Russell ve ark. (2008)	•			•					•		•						•	•		
H.-K. Chen ve ark. (2009)	•		•					•			•			•						
Chiang ve ark. (2009)	•			•					•		•						•	•		
Park ve Hong (2009)	•		•					•			•		•							
Geismar ve ark. (2011)	•		•					•			•	•	•		•					
W. Li ve Ferrell (2011)	•		•						•		•	•	•							
Farahani ve ark. (2012)	•			•				•			•		•						•	
Amorim ve ark. (2013)	•			•				•		•			•							
Condotta ve ark. (2013)	•		•					•			•				•					
Low ve ark. (2013)	•		•					•		•					•					

Çizelge 2.2. BÜDÇ problemlerine ilişkin literatür özeti (devam)

Kaynak	Üretim Ortamı						Dağıtım Ortamı						Amaç							
	Tek depo	Çok depo	Tek makine	Paralel makine	Akış Tipi	Atölye Tipi	Tek araç	Homojen araçlar	Heterojen araçlar	Sınırsız sayıda araç	Sınırlı sayıda araç	Çoklu turlar	Maliyet	Kar	Hizmet	Karşılana n talep miktarı	Kullanıla n araç sayısı	Kat edilen mesafe	Kalite	Çevre
Ullrich (2013)	•			•					•		•	•			•					
Chang ve ark. (2014)	•			•				•			•		•		•					
Hajiaghaei-Keshteli ve ark. (2014)	•		•						•		•		•		•					
Lee ve ark. (2014)	•			•					•		•	•	•							
Low ve ark. (2014)	•		•						•	•			•							
Viergutz ve Knust (2014)	•		•				•									•				
Zu ve ark. (2014)	•		•					•			•	•	•							
Belo-Filho ve ark. (2015)	•			•				•		•			•							
W. Li ve Ling (2015)	•		•					•			•	•	•							
Low ve ark. (2017)	•		•						•	•			•							
Cheref ve ark. (2016)	•		•				•					•			•					
Devapriya ve ark. (2017)	•		•					•		•		•	•							
Jamili ve ark. (2016)	•		•					•		•			•		•					
Johar ve ark. (2016)	•		•					•			•		•							
Kang ve ark. (2016)		•						•			•		•							
K. Li ve ark. (2016)	•		•					•			•		•		•					
Karaoğlu ve Kesen (2017)	•		•				•					•			•					
Kergosien ve ark. (2017)	•			•			•					•			•					

Çizelge 2.2. BÜDÇ problemlerine ilişkin literatür özeti (devam)

Kaynak	Üretim Ortamı						Dağıtım Ortamı						Amaç							
	Tek depo	Çok depo	Tek makine	Paralel makine	Akış Tipi	Atölye Tipi	Tek araç	Homojen araçlar	Heterojen araçlar	Sınırsız sayıda araç	Sınırlı sayıda araç	Çoklu turlar	Maliyet	Kar	Hizmet	Karşılanan talep miktarı	Kullanılan araç sayısı	Kat edilen mesafe	Kalite	Çevre
Fu ve ark. (2017)	•			•					•		•		•							
F Marandi ve Zegordi (2017)	•				•				•		•		•							
Ramezani ve ark. (2017)	•				•				•		•		•							
Zou ve ark. (2018)	•		•					•			•				•					
Gharaei ve Jolai (2018)		•		•				•		•			•		•					
Lacomme ve ark. (2018)	•		•					•			•	•			•					
Fateme Marandi ve Fatemi Ghomi (2019)		•	•						•		•		•							
Kesen ve Bektaş (2019)	•			•				•			•				•					
Mohammadi ve ark. (2020)	•					•			•		•		•		•					
J. Wang ve ark. (2019)	•		•						•		•									•
S. Wang ve ark. (2019)	•				•		•					•			•					
Liu ve ark. (2020)	•		•					•			•				•					
Ganji ve ark. (2020)	•		•						•				•		•					•
Yağmur ve Kesen (2020)	•				•		•				•	•			•					
He ve ark. (2021)	•		•					•			•		•		•					
PAT_TA Problemi	•				•		•				•	•			•					
PAT_ÇA Problemi	•				•				•		•	•			•					
AT_ÇA Problemi	•					•			•		•	•			•					•

2.1. Tek Makineli Üretim Ortamı için BÜDÇ Literatürünün Değerlendirilmesi

BÜDÇ problemleri ile ilgili yapılan ilk çalışmalar gazete basım ve dağıtımı ile ilgilidir. Hurter ve Van Buer (1996), bir basım evinde gazete basım ve dağıtım çizelgeleme problemini incelemişlerdir. İncelenen problemde dağıtımda kullanılan araç sayısı, temel dağıtım maliyetini oluşturduğu varsayılarak en küçüklenmiştir. Aynı problem daha sonra Van Buer ve ark. (1999), tarafından araçların birden fazla kez kullanılabilirdiği durum için incelenmiştir. Chang ve Lee (2004), dağıtım sürecinde tek aracın olduğu ve her bir siparişin araç üzerinde farklı depolama alanına ihtiyaç duyduğu problemi incelemişler, ardından problem için makine ve müşteri sayılarına göre üç farklı senaryo tanımlayarak bu senaryolar için sezgisel yöntemler geliştirmişlerdir. Garcia ve ark. (2004) hazır beton karışımlarının taşınma sürecini incelemişlerdir. Birden fazla üretim işletmesinin olduğu durum için geliştirilen doğrusal tamsayılı programlama modeli ile dağıtım maliyetlerini göz önünde bulundurarak karı en büyüleyecek siparişler belirlenmektedir. Z.-L. Chen ve Vairaktarakis (2005), problemi hazır yemek ve bilgisayar endüstrisine uyarlamışlardır. Çalışmada tek ve paralel makineli üretim ortamları için kesin ve sezgisel yöntemlere başvurulmuştur. C.-L. Li ve ark. (2005), müşteri sayısının tek olduğu durum için polinomsal zamanlı bir çözüm yaklaşımı önerirken, birden fazla müşterinin olduğu daha genel bir durum için dinamik programlama yaklaşımına başvurmuşlardır. Naso ve ark. (2007)'nin yapmış olduğu çalışmada üretim çizelgeleme problemi ile çok tesisli zaman pencereli araç rotalama problemi bütünleştirilmiştir. İncelenen problemde bazı tesisler diğer tesislerin kamyon filolarına ihtiyaç duymakta ve dış şirketlerden araç kiralaması yapılabilmektedir. Problemin amacı kamyonların yükleme ve boşaltma yapmak için bekleme sürelerine bağlı ceza maliyetleri ile ulaşım, dış kaynaklı üretim, ek olarak kiralanan kamyonlar ve sürücüler için fazla mesai maliyetlerinin toplamının en küçüklenmesidir. Geismar ve ark. (2008) çalışmalarında ürün ömrü olan siparişleri incelemişler ve bu ürünlerin üretim ve dağıtımı için gereken sürenin en küçüklenmesi üzerinde çalışmışlardır. H.-K. Chen ve ark. (2009) ürün kalitesinin zamanla belirli bir oranda azaldığı bozulabilen ürünler için doğrusal olmayan tamsayılı bir programlama modeli önermişlerdir. Stokastik müşteri taleplerinin olduğu problemde üretim, dağıtım ve prestij kaybı ile ilgili maliyetler dikkate alınarak tedarikçiden beklenen karı en büyülemek amaçlanmıştır. Park ve Hong (2009) tarafından incelenen problemde müşteriler sıkı ve gevşek teslim tarihlerine sahiptir. Geliştirilen karma tamsayılı doğrusal programlama modelinin amacı üretim, dağıtım ve gecikmelerden oluşan maliyetlerin en küçüklenmesidir. Geismar ve ark. (2011) sıfır

stokla çalışılan bir sistemde aktarma merkezlerinin avantajını vurgulayarak her bir merkez çevresinde yer alan müşteri siparişlerinin 3. parti lojistik firmalarınca sağlandığı durumu incelemişlerdir. W. Li ve Ferrell (2011), tarafından ele alınan problemde belirli bir ürün ömrü olan siparişlerin kapasite ve maliyet açısından farklılık gösteren araçlarla dağıtımını incelenmiştir. Condotta ve ark. (2013) her bir işin kendine ait üretime başlama ve en son teslim tarihinin olduğu durumu incelemişlerdir. Çalışmada en büyük gecikmenin en küçüklenmesi amaçlanırken ilgili problem için karma tamsayılı programlama yaklaşımıyla alt sınır geliştirilmiştir. Low ve ark. (2013) müşterilere ait zaman pencerelerini dikkate alarak tüm siparişlerin üretilip dağıtılması için gereken zamanın en küçüklenmesi için matematiksel bir model geliştirmişler, büyük boyutlu problemlerde ise genetik algoritmanın varyasyonlarına başvurmuşlardır. Hajiaghahi-Keshteli ve ark. (2014), problemi demiryolu endüstrisi için ele almışlardır. Amaç en düşük maliyetle müşteri hizmet düzeyinin en büyüklenmesidir. Low ve ark. (2014) bir dağıtım şirketinde siparişlerin heterojen araçlarla sistemdeki perakendecilere atanması ve perakendecilerin belirlediği bir zaman penceresi içerisinde her tur için müşteri ziyaret sıralarının belirlenmesi problemi ile ilgilenmişlerdir. Amaç zaman penceresinden sapma olması durumunda oluşan ceza maliyetleri, dağıtım maliyetleri ve sabit araç maliyetleri toplamını en küçükleme. Viergutz ve Knust (2014), Armstrong ve ark. (2008) tarafından ele alınan problemi üretime başlarken oluşan gecikmeleri göz önünde bulundurarak genişletmişlerdir. Problemde araç kapasitesi ve ürünün sınırlı kullanım ömrü olması nedeniyle bazı müşteri talepleri gerekli zaman pencereleri içinde karşılanamamaktadır. Geliştirilen karma tamsayılı doğrusal programlama modelinin amacı toplam karşılanan talebin en büyüklenmesidir. Zu ve ark. (2014), W. Li ve Ferrell (2011)'in yapmış oldukları çalışmayı aynı turda toplama ve dağıtım işlemlerine izin vererek genişletmişlerdir. Çalışmada tedarik zinciri, tedarikçi, firma ve müşteriler olmak üzere üç seviyeden oluşmaktadır. Geliştirilen karma tamsayılı programlama modelinin amacı toplam dağıtım maliyetinin en küçüklenmesidir. W. Li ve Ling (2015), birden fazla tedarikçi, tek üretim tesisi ve birden fazla müşterinin bulunduğu üç aşamalı bir tedarik zincirinde bozulabilir ürünlerin üretim ve dağıtımını incelemişlerdir. Çalışmada aracın aynı turda yükleme ve boşaltma yapmasına izin verilmiştir. Low ve ark. (2017), iki kademeli bir tedarik zincirinde dağıtımda heterojen araçların kullanıldığı ve müşterilere ait zaman pencerelerinin dikkate alındığı problem için genetik algoritmanın iki varyantının performansını karşılaştırmışlardır. Cheref ve ark. (2016) problemi belirsizlik altında incelemişlerdir. İncelenen problemde üretime başlama zamanları, üretim

zamanları, dağıtım süreleri ve sipariş teslim zamanları stokastik yapıdadır. Çözüm için geliştirilen karma tamsayılı doğrusal programlama modelinin amacı en büyük gecikmenin en küçüklenmesidir. Devapriya ve ark. (2017), Geismar ve ark. (2008)'nin yapmış oldukları çalışmayı filo büyüklüğü de bir karar değişkeni olacak şekilde genişletmişlerdir. Geliştirilen karma tamsayılı programlama modelinin amacı değişken ve sabit ulaşım maliyetleri toplamının en küçüklenmesidir. Jamili ve ark. (2016), geliştirdikleri çok amaçlı modelde dağıtım maliyeti ile ortalama teslim süresini ağırlıklı toplam yöntemiyle tek bir amaca dönüştürmüştür. Johar ve ark. (2016), her bir siparişin üretime başlama ve bitiş zamanının olduğu durumu incelemişlerdir. Önerilen sezgisel yöntem literatürde bilinen test problemleri üzerinde denenmiş ve başarısı kanıtlanmıştır. K. Li ve ark. (2016), sabit ve değişken dağıtım maliyeti toplamı ile müşterilerin toplam bekleme zamanını en küçükleyen çok amaçlı bir modeli incelemişlerdir. Karaoğlu ve Kesen (2017) yapmış oldukları çalışmada belirli bir ürün ömrü olan siparişlerin üretim ve dağıtım için gereken sürenin en küçüklenmesi ile ilgilenmişlerdir. Yazarlar, problem için dal-kesme algoritmasından yararlanırken, alt sınırların iyileştirilmesinde geçerli eşitsizlikler geliştirmişler, üst sınırların iyileştirilmesi amacıyla ise tavlama benzetimine dayanan yerel arama stratejisini kullanmışlardır. Zou ve ark. (2018), çalışmalarında en büyük teslim zamanını en küçüklemek için genetik algoritmaya dayalı bir yaklaşım önermişlerdir. Lacomme ve ark. (2018), Geismar ve ark. (2008) tarafından geliştirilen test problemleri için sistemde tek aracın olduğu durumda önermiş oldukları sezgisel yöntemle yeni alt sınırlar elde etmişlerdir. Fateme Marandi ve Fatemi Ghomi (2019), sistemde birden fazla üretim tesisinin olduğu durumda ulaşım ve gecikme maliyetleri toplamının en küçüklenmesi için geliştirdikleri algoritmanın performansını genetik algoritma ve parçacık sürüsü optimizasyonunun da içinde olduğu dört farklı sezgisel yaklaşımla kıyaslamışlardır. BÜDÇ problemlerini ilk kez çevresel açıdan ele alan J. Wang ve ark. (2019)'nın yapmış olduğu çalışmada havaya salınan CO₂ emisyonunun en küçüklenmesi için tabu arama algoritmasına başvurulmuştur. Bunun yanı sıra üretim ortamında ardışık iş parçaları arasında makinelerin kapatılmasına izin verilerek makinelerin elektrik tüketimi de incelenmiştir. Liu ve ark. (2020)'nin yapmış olduğu çalışmada siparişlerin teslim zamanlarının toplamını en küçüklemek için değişken komşu arama sezgiseline başvurulmuştur. Ganji ve ark. (2020), dağıtım ortamında heterojen kapasiteli bir filonun görev aldığı problemi çok amaçlı bir yapıda incelemiştir. Maliyet ve hizmet amaçlarının yanı sıra yakıt tüketimi ve havaya salınan CO₂ emisyonunun dikkate alındığı çalışmada parçacık sürüsü optimizasyonu, karınca kolonisi

optimizasyonu ve BSGA-II algoritmalarının performansı karşılaştırılmıştır. He ve ark. (2021) üç boyutlu yazıcılar ile tam zamanında dağıtım sistemini bütünleştirerek dağıtım maliyetleri ile teslimat zamanları toplamını ağırlıklı toplam şeklinde yeni bir küme kapsama modeli olarak formüle etmişler ve problem için kesin bir yöntem olan dal-ücret algoritmasına başvurmuşlardır.

2.2. Paralel Makineli Üretim Ortamı için BÜDÇ Literatürünün Değerlendirilmesi

Paralel makineli üretim ortamına ilişkin literatür, özdeş (identical, P_m) paralel makineler, düzgün (uniform, Q_m) paralel makineler ve ilişkisiz (unrelated, R_m) paralel makineler olmak üzere üç başlıkta incelenmiştir. İlk grupta aynı işi yapabilen makineler birbiriyle tamamen aynı özelliklere sahip iken ikinci gruptaki makinelerin hızları farklılık göstermektedir. Son gruptaki makinelerin hızları ise işlem görecektir olan işlere göre değişmektedir.

2.2.1. Özdeş paralel makineli üretim ortamı için BÜDÇ literatürünün değerlendirilmesi

Russell ve ark. (2008) gazete basım ve dağıtımını problemini paralel bir üretim hattında incelemişlerdir. Yapılan çalışmada dağıtım aşaması zaman pencereli ve bölge kısıtlı araç rotalama problemi olarak ele alınmış ve çözümde tabu arama algoritmasından yararlanılmıştır. Chiang ve ark. (2009), Russell ve ark. (2008)'nin incelemiş olduğu problemi ek bir gazete baskısı ile incelemiştir. Her iki çalışmada da zaman pencereli araç rotalama problemi karma tamsayılı doğrusal programlama modeli ile formüle edilmiştir. Tek bir araçla teslimat yapılabilen bölgelerin sayısında sınırlama yapılan çalışmada kullanılan araç sayısı ile toplam kat edilen mesafenin en küçüklenmesi amaçlanmıştır. Farahani ve ark. (2012) günümüzde oldukça yaygın olan hazır yemek şirketlerinde üretim ve dağıtım arasında geçen sürenin azaltılması yoluyla besin kalitesinin iyileştirilmesini incelemişlerdir. Karma tamsayılı doğrusal programlama modelinde sıra bağımlı hazırlık süreleri ve maliyetleri dikkate alınmıştır. Amaç hazırlık ve dağıtım maliyetleri ile bozulabilir gıda ürünlerinin kalitesi arasında bir ödünleşme sağlamaktır. Ullrich (2013) siparişlerin paralel makineli bir üretim ortamında üretilip heterojen araçlarla dağıtıldığı bir sistemi incelemiştir. Geliştirilen karma tamsayılı doğrusal programlama modelinin amacı siparişlerin toplam gecikmesinin en küçüklenmesidir. Gharaei ve Jolai (2018) çok tesisli BÜDÇ Problemini incelemişlerdir. Tesislerde tek makine bulunduğundan sistem özdeş paralel makineli bir üretim ortamı olarak incelenmiştir. Kesen ve Bektaş (2019)

özdeş paralel makineli üretim ortamı ile homojen sınırlı sayıda aracın görev aldığı dağıtım ortamını müşterilerin zaman pencerelerini dikkate alarak toplam erken gelme ve gecikme zamanlarını en küçükleme amacı altında incelemişlerdir.

2.2.2. Düzgün paralel makineli üretim ortamı için BÜDÇ literatürünün değerlendirilmesi

Amorim ve ark. (2013), bozulabilen ürünlerin olduğu siparişlerin düzgün paralel makinelerde işlendiği durumu incelemişlerdir. Çalışmada bir müşteri siparişinin farklı makinelerde işlenmek üzere alt birimlere bölünmesine izin verildiği durumla verilmediği durum kıyaslanmış ve hazırlık süresi ile maliyetlerinin sonuçları güçlü bir şekilde etkileyebileceği gösterilmiştir. Lee ve ark. (2014) tıbbi tedavi için kullanılan radyoaktif bileşenlerin üretim ve dağıtımıyla ilgilenmişlerdir. Makine kapasitesi ihlal edilmediği sürece, tek bir üretim koşumunda aynı anda birden çok sipariş üretilebilmektedir. Zaman pencerelerinin üst limiti ilaçların son kullanma süresidir. Geliştirilen karma tamsayılı doğrusal modelde amaç, üretim ve dağıtım maliyetleri toplamının en küçüklemesidir. Belo-Filho ve ark. (2015) çalışmalarında Amorim ve ark. (2013) tarafından geliştirilen modeli parti bölme yaklaşımıyla ele almışlar büyük boyutlu problemlerin çözümü için adaptif geniş komşuluk arama stratejisinden yararlanmışlardır. Kergosien ve ark. (2017), kemoterapi için kullanılan ilaçların birbirinden bağımsız paralel çalışan teknisyenler tarafından hazırlanıp son teslim tarihine kadar bozulmadan hastalara ulaştırılması sürecini incelemişlerdir. İlaçların bozulmaması için üretim ve teslim zamanı arasında geçen süreye ilişkin kısıtlar da dikkate alınmıştır.

2.2.3. İlişkisiz paralel makineli üretim ortamı için BÜDÇ literatürünün değerlendirilmesi

Chang ve ark. (2014)'nın yapmış olduğu çalışmada aynı araç tarafından teslim edilen tüm müşteri siparişleri bir partide sırasıyla üretilmektedir. Geliştirilen doğrusal olmayan matematiksel modelin amacı dağıtım süreleri ve toplam dağıtım maliyetlerinin ağırlıklı kombinasyonunun en küçüklemesidir. Fu ve ark. (2017), bir metal ambalaj endüstrisinde işlerin bölünebildiği durumu incelemişlerdir. İşler ilişkisiz paralel makinelerde sıra bağımlı hazırlık süresi dikkate alınarak işlenmektedir. Problem için geliştirilen matematiksel modelin amacı toplam hazırlık maliyeti ve dağıtım maliyetinin en küçüklemesidir.

2.3. Akış Tipi Üretim Ortamı için BÜDÇ Literatürünün Değerlendirilmesi

F Marandi ve Zegordi (2017), çabuk bozulan gıda ürünleri için üretim ve dağıtım süreçleri arasında geçen sürenin azaltılarak ürün kalitesinin artırılmasını incelemişler ve problemin çözümü için parçacık sürüsü optimizasyonundan yararlanmışlardır. Ramezani ve ark. (2017), akış tipi üretim ortamını incelemiş ve dağıtım sürecinde direkt teslimat ve rotalama yapılarak teslimat olmak üzere iki farklı yöntemi karşılaştırmışlardır. S. Wang ve ark. (2019), üç aşamalı hibrit akış tipi çizelgeleme problemini en büyük dağıtım tamamlanma zamanının en küçüklenmesi için incelemişlerdir. Hibrit akış tipi üretimin ilk aşamasında sıra bağımlı hazırlık süresine sahip özdeş paralel makineler yer alırken ikinci aşamada hazırlık süresi olmayan özdeş paralel makineler yer almaktadır. Son aşamada ise siparişe özel tahsis edilmiş makineler bulunmaktadır. Yazarlar problem için öncelikle karma tamsayılı bir programlama yaklaşımı geliştirmişler ardından değişken komşuluk arama stratejisine dayalı üç farklı sezgisel yöntem önermişlerdir. Yağmur ve Kesen (2020), permütasyon akış tipi üretim ortamı ile sistemde tek bir aracın bulunduğu dağıtım ortamında toplam gecikme ile toplam tur zamanı toplamını en küçüklemek için MA yaklaşımını kullanmışlardır.

2.4. Atölye Tipi Üretim Ortamı için BÜDÇ Literatürünün Değerlendirilmesi

Kang ve ark. (2016) yarı iletken cihaz üretiminde farklı üretim aşamalarının farklı üretim tesislerinde gerçekleştirildiği bir sistemi incelemişlerdir. İşler ürün tipine göre sınıflandırılmış ve partiler halinde dış kaynaklı firmalarda işlenmektedir. Üretim maliyetleri ürün tipi ve kullanılan dış kaynaklı firma türüne göre değişkenlik göstermektedir. Dağıtımda birden fazla araç kullanılmaktadır ve her bir aracın en fazla kat edebileceği mesafe sınırlandırılmıştır. Amaç üretim ve dağıtım maliyetleri toplamının en küçüklenmesidir. Mohammadi ve ark. (2020) mobilya imalatı yapan bir firmada esnek atölye tipi üretim ortamında bütünleşik problemi incelemişlerdir. Amaç üretim ve dağıtım maliyetlerinin toplamı ile teslim tarihlerine göre erken gelme ve gecikme zamanlarının ağırlıklı toplamının en küçüklenmesidir. Yazarlar, çok amaçlı modelin çözümünde epsilon-kısıt yönteminden ve hibrit parçacık sürüsü optimizasyonundan yararlanmışlardır.

2.5. BÜDÇ Literatürünün Genel Değerlendirilmesi

BÜDÇ problemlerinin ilk olarak çalışılmasından (Potts, 1980) bu yana yaklaşık kırk yıla yakın bir süre geçmiştir. Z.-L. Chen (2010) yapmış olduğu kaynak araştırması

çalışmasında operasyonel düzeyde BÜDÇ problemlerinin yeterince çalışılmadığını belirtse de o tarihten günümüze kadar özellikle son 10 yıl içerisinde BÜDÇ problemlerinin literatürde önem kazandığı net bir şekilde görülmektedir. Problemin çalışılmaya başlandığı ilk yıllarda hem üretim hem de dağıtım ortamı açısından daha basit varyantları çalışılmış, problem literatürde yer edinmeye başladığında ise daha önce çalışılmayan alanlar keşfedilerek bu boşluklar doldurulmaya çalışılmıştır. Çizelge 2.3'te tez kapsamında incelenen 51 adet çalışmada ele alınan problemlere ilişkin genel bir değerlendirme yapılmıştır. Çizelgenin ilk sütununda ilgili özellik, ikinci sütunda bu özelliklere ilişkin olası durumlar ve son sütunda ise ikinci sütunda verilen durumu kapsayan kaynakların oranları verilmiştir.

Çizelge 2.3. BÜDÇ Literatürüne ilişkin genel değerlendirme

Özellik	Durum	Kaynak Sayısı
Üretim Ortamı	Tek Depo	46/51
	Çok Depo	5/51
	Tek Makine	33/51
	Paralel Makine	13/51
	Akış Tipi	4/51
	Atölye Tipi	2/51
Dağıtım Ortamı	Tek Araç	9/51
	Homojen Araçlar	27/42
	Heterojen Araçlar	15/42
	Yeterli sayıda araç	10/42
	Sınırlı sayıda araç	32/42
	Çoklu turlar	9/42
Amaç	Maliyet	19/51
	Hizmet	15/51
	Maliyet, Hizmet	9/51
	Kar	2/51
	Kullanılan Araç Sayısı, Kat edilen mesafe	2/51
	Karşılanan Talep Miktarı	1/51
	Kalite	1/51
	Çevre	1/51
	Çevre, Hizmet, Maliyet	1/51

Çizelge 2.3 üretim ortamı açısından değerlendirildiğinde sistemde birden fazla üretim işletmesinin olduğu durumun çok sınırlı sayıda çalışıldığı görülmektedir. Makine ortamlarına bakıldığında ise en fazla çalışılan makine ortamı tek makine olup bunu paralel makineli üretim ortamı izlemektedir. Çok operasyonlu işlere ait akış tipi ve atölye tipi üretim ortamları ise çok sınırlı sayıda çalışılmıştır. Sonuç olarak literatürde yer alan BÜDÇ problemlerinin büyük çoğunluğunun nispeten daha basit üretim ortamlarını dikkate aldığı söylenebilir.

Çizelge 2.3 dağıtım ortamı açısından değerlendirildiğinde çalışmaların büyük çoğunluğunda müşterilere birden fazla araç hizmet vermektedir. Sistemde araç filosunun olduğu durumda araçların homojen oldukları durum heterojen oldukları duruma göre daha fazla çalışılmıştır. Benzer şekilde çok az sayıdaki çalışmada araç sayısı sınırlandırılmazken çalışmaların büyük çoğunluğunda araç sayısı problem parametresi olarak sınırlandırılmıştır. Sistemde tek aracın olduğu problemlerde araçların kapasite sınırlamasından dolayı müşteri taleplerini karşılamak için birden fazla turda görev aldığı görülmektedir. Ancak sistemde birden fazla aracın olduğu problemlerde araçların birden fazla turda görev almasına çok sınırlı sayıdaki çalışmada izin verilmiştir.

Çizelge 2.3 amaç fonksiyonları açısından değerlendirildiğinde en fazla çalışılan amaç fonksiyonları maliyet ve hizmet bileşenlerini içermektedir. Bazı çalışmalarda birden fazla amaç eş zamanlı olarak dikkate alınırken, son yıllarda yapılan çalışmalarda klasik amaçların dışında çevresel faktörlerin de ilgi çektiği görülmektedir.

Sonuç olarak tez kapsamında incelenen problemlerde üretim ortamı açısından literatürde boşlukları fark edilen çok operasyonlu işlerin çizelgelenmesine odaklanılmıştır. Dağıtım ortamında ise sistemde hem tek aracın olduğu hem de heterojen kapasiteli birden fazla aracın olduğu durum incelenmiştir. Araç filosundaki araçların planlama periyodu boyunca birden fazla kez kullanılmasına izin verilerek literatürdeki bir diğer boşluk da doldurulmaya çalışılmıştır. Amaç fonksiyonları açısından ise her üç problemde de ele alınan amaçlar daha önce çalışılmamış olup ilk kez bu tez çalışmasında incelenmiştir.

3. PERMÜTASYON AKIŞ TİPİ ÜRETİM ORTAMI İLE TEK ARAÇLI DAĞITIM ORTAMINI İÇEREN BÜDÇ (PAT_TA) PROBLEMİ

Önceki bölümde belirtildiği gibi yapılan kaynak araştırması sonucunda rotalama kararlarının alındığı BÜDÇ Problemleri içerisinde en fazla çalışılan makine ortamının tek makine ve paralel makineli üretim ortamları olduğu görülmüştür. Nispeten daha karmaşık yapıdaki akış tipi ve atölye tipi üretim ortamları ise literatürde oldukça sınırlı sayıda incelenmiştir. Bununla birlikte çok operasyonlu işleri ele alan problemlerin gerçek hayattaki karşılıkları ve uygulama alanları tek makine ve paralel makineli üretim ortamlarına göre daha fazladır.

Akış Tipi Üretim ortamı, genellikle üretim hacminin yüksek, ürün çeşitliliğinin sınırlı olduğu durumlarda tercih edilmektedir. Bu üretim ortamında makineler üretim hattında birbiri ardına sıralanmakta ve işlem görecekt olan tüm işler nihai ürün haline gelmek için hattaki bu rotayı takip etmektedir. Akış Tipi Üretim ortamının özel bir varyantı olan Permütasyon Akış Tipi Üretim ortamında ise işlerin tüm makinelerdeki işlem sıraları aynıdır. Bu üretim ortamı özellikle seri üretim için uygun bir seçenektir. Buna göre tez kapsamında incelenen ilk problemin gerçek hayattaki karşılıklarına örnek olarak seri üretime göre paketlenmiş gıda üretimi yapan bir işletme verilebilir. Özellikle seri üretime göre üretilen hızlı tüketim ürünlerinin her gün, her an bulunabilirliklerini sağlamak amacıyla bütünleşik yaklaşım tercih edilebilir. Böylece tez kapsamında incelenen ilk problemin üretim ortamı Permütasyon Akış Tipi üretim ortamı olarak belirlenmiştir.

Bu bölüm, PAT_TA Probleminin tanımı, PAT_TA Problemi için önerilen yöntemler ve önerilen yöntemlere ait karşılaştırmalı sonuçlar olmak üzere üç alt başlıkta incelenmiştir.

3.1. PAT_TA Probleminin Tanımı

PAT_TA Probleminin üretim ortamı incelendiğinde sistemde siparişlerin seri şekilde işlendiği makine grubundan (M) oluşan tek bir tesis bulunmaktadır. Bu tesiste, bir grup sipariş (N_C) her bir makineyi aynı sırada ziyaret ederek (permütasyon akış tipi üretim ortamı) işlenmektedir. Herhangi bir siparişin ($i \in N_C$), herhangi bir makinedeki ($m \in M$) işlem zamanı önceden bilinen p_{im} kadardır. İşlerin planlama periyodu başında hazır olduğu ve makinelerde hazırlık süresinin oluşmadığı varsayılmaktadır. Herhangi bir işin herhangi bir makinedeki operasyonu bitmeden diğer bir operasyonu başlatabamaz ve

her bir iş bir makineyi en fazla bir kez ziyaret edebilir. Dağıtım aşamasında ise her sipariş farklı bir müşteriye ait olduğundan sipariş ve müşteri kavramları birbirinin yerine kullanılmıştır. Herhangi bir i müşterisine ait talep miktarı d_i önceden bilinmektedir. Sipariş teslimatlarının partiler halinde yapılacağı varsayılmıştır. Bir parti de farklı müşterilere ait siparişler bulunabilmektedir yani aynı turda birden fazla müşteri ziyaret edilebilir. Modelin sadeleştirilmesi adına üretim ve dağıtım partilerinin sıralamalarının aynı olduğu varsayılmıştır. Rotalama aşamasında '0' depo düğümü ve diğer düğümler de müşterileri ifade etmektedir. Herhangi bir i ve j düğüm çifti arasındaki ulaşım zamanı t_{ij} ile ifade edilmekte ve ulaşım zamanlarının simetrik yapıda olduğu ($t_{ij} = t_{ji}$) ve üçgensel eşitsizliğin sağlandığı ($t_{ik} \leq t_{ij} + t_{jk}$) varsayılmaktadır. Belirli bir partideki siparişlerin üretimi tamamlandıktan sonra bu siparişler, ilgili müşterilere müşterilerin belirlediği teslim tarihi dd_i dikkate alınarak sistemde bulunan tek bir araçla dağıtılmaktadır. Aracın i müşterisine ulaşma zamanı A_i ile gösterilmektedir. Araç müşterilerin belirlemiş oldukları teslim tarihinden sonra geldiğinde T_i kadar bir gecikme oluşmaktadır.

İncelenen problem sonucunda; (i) müşteri siparişlerinin makinelerdeki operasyon sıralarının belirlenmesi, (ii) her bir müşteri siparişinin üretime başlama ve üretim tamamlanma zamanının belirlenmesi, (iii) aracın her bir turda dağıtım yapacağı müşterilerin ziyaret sıralarının belirlenmesi, (iv) her bir müşteri siparişinin teslimat zamanının belirlenmesine ilişkin kararlar alınmaktadır.

PAT_TA Probleminin daha iyi anlaşılabilmesi amacıyla öncelikle ele alınan problem için KTDP modeli geliştirilmiş ardından PAT_TA Problemine ilişkin örnek bir çözüm incelenmiştir.

3.1.1. PAT_TA Problemi için önerilen KTDP modeli

Bu bölümde PAT_TA Problemi için önerilen KTDP modelinde kullanılan simgeler, model ve modelde yer alan kısıtların açıklamaları detaylarıyla verilmiştir. Önerilen KTDP modelinde kullanılan simgeler ve açıklamaları Çizelge 3.1'de verilmiştir.

Çizelge 3.1. Önerilen KTDP modelinde kullanılan simgeler

Simge	Açıklama
<i>Indisler:</i>	
i, j	: İş (sipariş, müşteri)
m	: Makine
<i>Kümeler:</i>	
N_C	: İşler (siparişler, müşteriler) kümesi
N	: Tüm düğümler kümesi ($N_C \cup 0$)
M	: Makineler kümesi
<i>Parametreler:</i>	
d_i	: i . müşteri siparişine ait talep miktarı
p_{im}	: i . müşteri siparişinin m . makinedeki işlem süresi
t_{ij}	: i . ve j . düğümler arası ulaşım süresi
dd_i	: i . müşteri siparişine ait teslim tarihi
Q	: Araç kapasitesi
H	: Yeterince büyük bir sayı
<i>Değişkenler:</i>	
X_{ij}	: 1, araç i . düğümden hemen sonra j . düğüme gidiyorsa; 0, aksi durumda
W_{ij}	: 1, araç önceki turu i . düğümden tamamlayıp sonraki tura j . düğüme başlıyorsa; 0 aksi durumda
u_i	: i . müşteriye ziyaret etmeden hemen önce araç üzerindeki toplam yük miktarı
S_{im}	: i . müşteri siparişinin m . makinedeki üretime başlama zamanı
C_{im}	: i . müşteri siparişinin m . makinedeki üretim bitiş zamanı
A_i	: aracın i . müşteriye varış zamanı
Y_i	: i . müşteri siparişinin ait olduğu partinin üretim bitiş zamanı
T_i	: i . müşteri siparişi için gecikme miktarı

$$\min \sum_{i \in N_C} T_i + \sum_{i \in N} \sum_{j \in N, i \neq j} \sum_{k \in V} t_{ij} X_{ijk} \quad (3.1)$$

$$\sum_{i \in N} X_{ij} = 1 \quad \forall j \in N_C \quad (3.2)$$

$$\sum_{j \in N_C} X_{ij} = \sum_{j \in N_C} X_{ji} \quad \forall i \in N \quad (3.3)$$

$$u_j - u_i + QX_{ij} + (Q - d_i - d_j)X_{ji} \leq Q - d_i \quad \forall i, j \in N_C; i \neq j \quad (3.4)$$

$$u_i \geq d_i + \sum_{j \in N_C; j \neq i} d_j X_{ij} \quad \forall i \in N_C \quad (3.5)$$

$$u_i \leq Q - (Q - d_i)X_{i0} \quad \forall i \in N_C \quad (3.6)$$

$$\sum_{j \in N_C; j \neq i} W_{ij} \leq X_{i0} \quad \forall i \in N_C \quad (3.7)$$

$$\sum_{i \in N_C; i \neq j} W_{ij} \leq X_{0j} \quad \forall j \in N_C \quad (3.8)$$

$$\sum_{j \in N_C} X_{0j} - \sum_{i \in N_C} \sum_{j \in N_C; j \neq i} W_{ij} = 1 \quad (3.9)$$

$$Y_i \geq C_{im} \quad \forall i \in N_C \quad (3.10)$$

$$Y_i \leq C_{im} + H(1 - X_{i0}) \quad \forall i \in N_C \quad (3.11)$$

$$Y_j - Y_i \leq H(1 - X_{ij} - X_{ji}) \quad \forall i, j \in N_C; i \neq j \quad (3.12)$$

$$A_i - A_j + HX_{ij} + (H - t_{ij} - t_{ji})X_{ji} \leq H - t_{ij} \quad \forall i, j \in N_C; i \neq j \quad (3.13)$$

$$A_i - A_j + HW_{ij} \leq H - t_{i0} - t_{0j} \quad \forall i, j \in N_C; i \neq j \quad (3.14)$$

$$A_j \geq HW_{ij} - H + Y_j + t_{0j} \quad \forall i, j \in N_C; i \neq j \quad (3.15)$$

$$A_j \geq Y_j + t_{0j} - H \left(1 - X_{0j} + \sum_{i \in N_C} W_{ij} \right) \quad \forall j \in N_C; i \neq j \quad (3.16)$$

$$A_j \leq Y_j + t_{0j} - H \left(1 - X_{0j} + \sum_{i \in N_C} W_{ij} \right) \quad \forall j \in N_C; i \neq j \quad (3.17)$$

$$C_{im} - S_{jm} \leq H(1 - X_{ij} - W_{ij}) \quad \forall i, j \in N_C; i \neq j; \forall m \in M \quad (3.18)$$

$$S_{i(m+1)} \geq C_{im} \quad \forall i \in N_C; m \in M - 1 \quad (3.19)$$

$$C_{im} = S_{im} + p_{im} \quad \forall i \in N_C; m \in M \quad (3.20)$$

$$T_i \geq A_i - dd_i \quad \forall i \in N_C \quad (3.21)$$

$$A_i, u_i, S_{im}, C_{im}, Y_i, T_i \geq 0 \quad \forall i \in N_C; m \in M \quad (3.22)$$

$$X_{ij} \in \{0,1\} \quad \forall i, j \in N \quad (3.23)$$

$$W_{ij} \in \{0,1\} \quad \forall i, j \in N_C \quad (3.24)$$

Matematiksel modelin amacı Eş. (3.1), teslim tarihlerinden sonra yapılan teslimatlardan kaynaklanan gecikmelerin toplamı ile toplam tur zamanı toplamını en küçükmektir. Eş. (3.2) ve Eş. (3.3)'da verilen kısıtlar atama kısıtları olarak adlandırılmaktadır. Eş. (3.2) bir müşteriye mutlaka bir kez hizmet verilmesini garanti etmekte, Eş. (3.3) ise düğümlerin girdi çıktı dengesini sağlamaktadır. Eş. (3.4) – Eş. (3.6) kapasite ve alt tur eleme kısıtlarıdır. Eş. (3.4) herhangi bir düğüm üzerindeki yük miktarının araç kapasitesini geçmemesini sağlamaktadır. Eş. (3.5) ve Eş. (3.6) alt tur oluşumunu önlemek için kullanılan u_i değişkeninin sınırlarını belirlemektedir. Eş. (3.7) ve Eş. (3.8) turlarda ziyaret edilen ilk ve son müşteriler kullanılarak tur kombinasyonunu sağlayan kısıtlardır. Eş. (3.9), ardışık tur kombinasyonlarının sayısının her zaman toplam tur sayısından bir eksik olmasını garanti etmektedir. Eş. (3.10) – Eş. (3.12) aracın dağıtımına çıkması için o tura ait tüm müşteri siparişlerinin üretiminin tamamlanması gerektiğini belirtmektedir. Eş. (3.10) ve Eş. (3.11) ilgili turdaki son ziyaret edilen müşteri siparişinin üretim tamamlanma zamanını belirlerken, Eş. (3.12) son ziyaret edilen müşteriden önce ziyaret edilen müşteriler için de aracın turdaki son müşteri siparişinin üretiminin tamamlanması gerektiğini belirtmektedir. Eş. (3.13) aynı turda bulunan ardışık müşteriler için sipariş teslim zamanını belirlemektedir. Eş. (3.14) ve Eş. (3.15) başka bir turu izleyen turdaki ilk müşteri için sipariş teslim zamanını belirlemektedir. Eş. (3.14) ve

Eş. (3.15) sırasıyla dağıtımda aracın ve üretimin beklenildiği durum için kullanılmaktadır. Eş. (3.16) ve Eş. (3.17) ilk turdaki ilk müşteri için sipariş teslim zamanını belirlemektedir. Benzer şekilde Eş. (3.18) aynı turda bulunan ardışık müşteriler için ve başka bir turu izleyen turdaki ilk müşteri için üretime başlama zamanlarını belirlemektedir. Eş. (3.19) ve Eş. (3.20) permütasyon akış tipi üretim ortamında ardışık makineler ve işler için üretime başlama ve üretim tamamlanma zamanlarını göstermektedir. Eş. (3.21) müşterilerin belirlemiş oldukları teslim tarihi dikkate alınarak aracın geç teslimat yapması durumundaki gecikmeyi belirlemektedir. Eş. (3.22) – Eş. (3.24) modeldeki değişkenlere ait negatif olmama ve 0-1 tamsayı olma kısıtlarıdır.

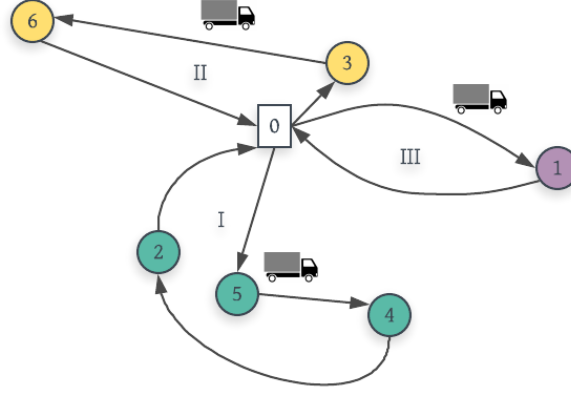
3.1.2. PAT_TA Problemi için açıklayıcı örnek

PAT_TA Probleminin daha iyi anlaşılabilmesi için örnek bir test verisi incelenmiştir. Problem, daha önce belirtilen varsayımlar altında 6 farklı müşteriye ait siparişlerin 2 makineli permütasyon akış tipi üretim ortamında üretildikten sonra tek araçla ilgili müşterilere teslimatların yapıldığı bir BÜDÇ Problemini içermektedir. İlgili test verisine ait parametre değerleri Çizelge 3.2’de verilmiştir. Çizelgenin ilk sütunu ‘0’ depo düğümü ve diğer düğümlerde müşterileri ifade edecek şekilde sistemdeki düğümleri göstermektedir. İkinci ve üçüncü sütunda depo düğüm ve müşterilerin koordinatlarına ait bilgiler yer almaktadır. PAT_TA Probleminde taşıma süresi matrisi için bu koordinatlardan yararlanarak öklit uzaklıkları hesaplanmıştır. Dördüncü sütunda müşteri talepleri yer alırken beşinci ve altıncı sütun sırasıyla birinci ve ikinci makinelerdeki işlem sürelerini içermektedir. Çizelgenin son sütunu ise müşteriler tarafından belirlenmiş teslim tarihi bilgilerini içermektedir. Rassal örneğe ilişkin araç kapasitesi Q , 211 birim olarak belirlenmiştir.

Çizelge 3.2. Örnek probleme ilişkin rassal veri seti

i	X	Y	d_i	p_{i1}	p_{i2}	dd_i
0	0	0	-	-	-	-
1	42	-12	89	90	88	904
2	-17	-31	87	91	96	544
3	9	15	90	91	92	472
4	17	-48	63	60	61	435
5	-6	-44	55	52	60	337
6	-38	29	64	61	62	810

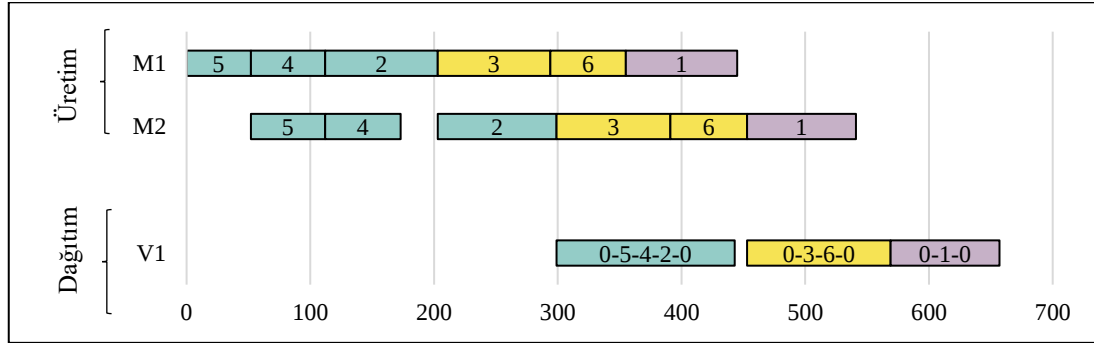
Test verisine ilişkin uygun bir çözüm Şekil 3.1’de verilmiştir. Şekil 3.1’de görüldüğü gibi sistemde tek aracın olduğu durumda araç üç farklı turda (0-5-4-2-0), (0-3-6-0), (0-1-0) müşteri siparişlerinin teslimatlarını gerçekleştirmektedir.



Şekil 3.1. Örnek probleme ilişkin uygun bir çözüm

PAT_TA Problemi için uygun çözüme ilişkin Gantt şeması ise Şekil 3.2’de verilmiştir. Şekil 3.2 incelendiğinde ilk turda yer alan müşteri siparişlerinin tamamlanma zamanı turdaki son müşteri olan 2 numaralı müşterinin son makinedeki üretim bitiş zamanına eşdeğerdir ($t = 299$). Araç bu anda depoda hazır bulunduğundan ilk turun dağıtımına $t = 299$ anında başlamıştır. Araç ilk turdaki müşterilere sırasıyla $A_5 = 344$, $A_4 = 368$ ve $A_2 = 407$ anında ulaşmıştır. Müşterilerin belirledikleri teslim tarihleri dikkate alındığında 4 ve 2 numaralı müşteriler için teslim tarihlerinden önce teslimat yapıldığı görülmüştür. 5 numaralı müşteride ise son teslim anı $dd_5 = 337$ olduğundan $T_5 = 344 - 337 = 7$ birim gecikme oluşmuştur. Araç ilk turda yer alan müşterilerin siparişlerini teslim ettikten sonra depoya $t = 443$ anında geri dönmüştür. Aracın ikinci tura başlayabilmesi için bu turdaki müşteri siparişlerinin üretimlerinin tamamlanması gerekmektedir. Bu nedenle araç en erken 6 numaralı siparişin son makinedeki üretim bitiş zamanı olan $t = 453$ anında dağıtımına çıkmış yani üretimin tamamlanması için $453 - 443 = 10$ birim atıl olarak depoda beklemiştir. Araç ikinci turdaki müşterilere sırasıyla $A_3 = 471$ ve $A_6 = 521$ anında ulaşmıştır. İlgili müşterilerin talep ettikleri teslim zamanı dikkate alındığında bu turda herhangi bir gecikme oluşmamıştır. Bu durumda araç ikinci turu $t = 569$ anında tamamlamıştır. Üçüncü turda teslim edilmesi gereken siparişlerin üretimi $t = 541$ anında bittiğinden araç üçüncü tura en erken depoda hazır olduğu $t = 569$ anında çıkabilmiştir. Araç son turdaki müşteriye ise $A_1 = 613$ anında ulaşmış ve

yine bir gecikme olmamıştır. Araç sistemde bulunan tüm müşteri siparişlerini teslim ettikten sonra depoya $t = 657$ anında dönmüştür. Son durumda PAT_TA Problemi için incelenen uygun çözümde toplam gecikme 7 birim bulunurken toplam tur zamanı ise 348 birim olarak hesaplanmıştır.



Şekil 3.2. Uygun çözüme ilişkin Gantt şeması

3.2. PAT_TA Problemi için Materyal ve Yöntem

Bu bölümde popülasyon tabanlı evrimsel bir algoritma olan MA ile ilgili genel bir bilgi verildikten sonra MA yaklaşımının PAT_TA Problemine uyarlanması üzerinde durulmuştur. Ardından önerilen MA'nın daha iyi anlaşılabilmesi amacıyla algoritma adımları örnek bir test verisine uygulanmış ve son olarak önerilen yöntemlerin performanslarını değerlendirmek amacıyla PAT_TA Problemi için test problemlerinin üretilme mekanizması anlatılmıştır.

3.2.1. Memetik Algoritmalar

Memetik Algoritma (MA) fikri ilk kez Golberg (1989)'in çalışmasında "Hibrit Genetik Algoritma" ismiyle Genetik Algoritma'nın bir varyantı olarak ortaya atılmış daha sonra bu yaklaşım Moscato (1989) tarafından "Memetik Algoritma" olarak adlandırılmıştır. MA'yı genetik algorithmadan ayıran en temel özellik çözüm kalitesini artırmak üzere kullanılan yerel arama sürecidir.

Sezgisel algoritmaların başarısı çeşitlendirme (diversification) ve yoğunlaşma (intensification) olarak adlandırılan iki temel stratejiye bağlıdır. MA ile popülasyon tabanlı bütünsel arama yaklaşımı ile arama uzayının büyük bir kısmı taranırken yerel arama yaklaşımı ile de taranan bölgelerde komşu çözümler incelenerek daha kaliteli bireylerin elde edilmesi sağlanmaktadır. Böylece iki temel stratejiyi de bünyesinde barındıran MA, polinomsal zamanda çözülemeyen pek çok problemin çözümünde

sıklıkla başvurulan bir yöntem olmuştur. Garg (2010), MA' da kullanılan yerel arama süreci ile genetik algoritmada ortaya çıkan erken yakınsama probleminin azaldığını belirterek her iki yaklaşımı çok sayıda veri seti üzerinde karşılaştırmış ve MA'nın genetik algoritmaya göre daha iyi sonuç verdiğini kanıtlamıştır. Krasnogor ve Smith (2005) MA'nın uygulandığı literatürdeki çalışmaları inceleyerek algoritmanın farklı varyantlarını ortaya koymuş ve algoritmaları tasarımsal olarak değerlendirerek sınıflandırmışlardır. Yapılan araştırma makaleleri incelendiğinde MA'nın literatürde yer alan pek çok probleme başarılı bir şekilde uygulandığı görülmüştür. MA geliştirildiği dönemlerde Moscato ve Norman (1992), tarafından gezgin satıcı problemine uygulanmıştır. MA'nın literatürde sıklıkla kullanılan farklı gezgin satıcı problemleri üzerindeki performansı Merz ve Freisleben (2001) tarafından yapılan çalışmada ayrıntılı olarak değerlendirilmiştir. Bunun dışında Kuadratik Atama Problemi (Tang ve ark., 2006), Yayılan Ağaç Problemi (Fischer ve Merz, 2007), Çizelgeleme problemleri (Basseur ve ark., 2002; Hasan ve ark., 2009; Marchiori ve Steenbeek, 2000) gibi kombinatoriyal problemlerde de algoritmanın etkinliği kanıtlanmıştır.

MA'nın uygulama adımlarının daha iyi anlaşılabilmesi için bu bölümde öncelikle evrimsel bir algoritmada kullanılan temel kavramlardan bahsedilmiştir.

Gen, belirli bir fiziksel özellikten sorumlu olan ve diğer genlerle birlikte yorumlanması gereken kromozom üzerindeki en küçük parçadır. Kromozom ise bir bireyin tüm genetik bilgilerini içeren bir dizidir. Bu tanımlardan yola çıkarak gen probleme ait anlamlı bilgi taşıyan en küçük genetik birim olarak tanımlanabilir. Kromozom ise incelenen problem için aday çözümleri ifade etmektedir. Evrimsel algoritmalarda, yeniden üretim sürecinde oluşan bireyler ebeveynlerinin sadece fiziksel özelliklerini (fenotip) değil aynı zamanda fiziksel özelliklerle ilgili bilgilerini (genotip) de miras olarak alır. Bu nedenle genetik operatörler genotip üzerinde çalışırken, bireylerin değerlendirilmesi ise fenotip üzerinden gerçekleştirilir. Kromozomun içerdiği bilgileri yani kod yapısını çözüm için anlamlı bilgiye dönüştürme süreci genetik kodun çözümü (decoding) olarak adlandırılırken, tam tersi işlem kodlama işlemi (encoding) olarak adlandırılmaktadır.

Evrimsel bir algoritmada kullanılan temel kavramlar açıklandıktan sonra MA'nın çalışma prensibi Çizelge 3.3'te verilmiştir. Algoritma başlatılmadan önce kullanılacak olan parametreler belirlenir (Adım 0). Bunlar popülasyon büyüklüğü, maksimum jenerasyon sayısı, çaprazlama oranı ve mutasyon oranıdır. Ardından başlangıç popülasyonu oluşturulur (Adım 1). Başlangıç popülasyonunda popülasyon büyüklüğü

kadar birey yer alır. Bundan sonraki süreçte popülasyonda yer alan bireylerin uyum değerleri hesaplanır. Herhangi bir bireyin uyum değeri belirlenirken öncelikle genetik kodun çözümü süreci gerçekleştirilir ardından problem için anlamlı bir çözüm haline gelen kromozomun amaç fonksiyonu değeri hesaplanır (Adım 3). Yeniden üretim mekanizması için popülasyonda yer alan bireyler arasından ebeveynler belirli bir seçim yöntemine göre seçilir (Adım 4). Ardından seçilen ebeveynlerin gen bilgileri kullanılarak çaprazlama operatörü ile yeni bireyler oluşturulur (Adım 5). Genetik çeşitliliği korumak adına oluşturulan bireyler belirli bir olasılıkla mutasyona uğratılır (Adım 6). Genetik operatörler (çaprazlama ve mutasyon) sonucu oluşan yeni bireyler üzerinde çeşitli komşu arama stratejileri aracılığıyla yerel arama süreci gerçekleştirilir (Adım 7). Bu şekilde yeni oluşturulan bireylerden daha kaliteli bireyler elde edilebilir. Son durumda elde edilen bireylerin uyum değerleri hesaplanır (Adım 8) ve popülasyon uyum değerlerine göre kaliteli bireyler popülasyonda kalacak şekilde güncellenir (Adım 9). Adım 3 ve Adım 9 arasında gerçekleşen süreç durma kriteri sağlanana kadar tekrarlanır (Adım 10). Ve durma kriteri sağlandıktan sonra algoritma durdurularak elde edilen en iyi birey (çözüm) raporlanır (Adım 11).

Çizelge 3.3. MA' nın temel adımları

Adımlar	Açıklama
Adım 0	Algoritma parametrelerini belirle.
Adım 1	Başlangıç popülasyonunu oluştur.
BAŞLA	Durma kriteri sağlanana kadar aşağıdaki adımları uygula.
Adım 3	Popülasyondaki bireylerin uyum değerlerini hesapla.
Adım 4	Ebeveynleri seç.
Adım 5	Seçilen ebeveynlerden yeniden üretme mekanizması (çaprazlama) ile yeni bireyler oluştur.
Adım 6	Üretilen bireyleri belirli bir olasılıkla mutasyona uğrat.
Adım 7	Yerel arama süreci ile oluşturulan bireyleri iyileştir.
Adım 8	Popülasyona yeni katılan bireylerin uyum değerlerini belirle.
Adım 9	Sonraki jenerasyon için uyum değerlerine göre popülasyonu güncelle.
Adım 10	Durma kriteri sağlanıyorsa Adım 11'e git; aksi takdirde Adım 3'e dön.
BİTİR	Algoritmayı durdur.
Adım 11	En iyi uyum değerine sahip bireyi (çözümü) elde et.

3.2.2. PAT_TA Problemi için önerilen MA

Permütasyon akış tipi çizelgeleme problemi ile araç rotalama problemi kendi içlerinde polinomsal zamanda optimal çözümleri bulan bir algoritmanın olmadığı NP-zor problem sınıfında yer almaktadır. Dolayısıyla iki problemi eş zamanlı olarak ele alan bütünleşik problemin de NP-zor yapıda olduğu açıktır. Bu nedenle, büyük boyutlu problemlerin çözümü için makul sürede optimal ya da optimale yakın çözümler veren

sezgisel bir algoritmanın geliştirilmesi önem arz etmektedir. Tez çalışması kapsamında PAT_TA Probleminin çözümü için popülasyon tabanlı evrimsel bir algoritma olan MA önerilmiştir. PAT_TA Probleminin çözümü için önerilen algoritmaya ait genel yapı Çizelge 3.4'te verilmiştir.

Çizelge 3.4. PAT_TA Problemi için önerilen MA'nın adımları

Adımlar	Açıklama
Adım 0	Algoritma parametrelerini (<i>popülasyon büyüklüğü, turnuva büyüklüğü, çaprazlama oranı, mutasyon oranı, maksimum jenerasyon sayısı</i>) belirle.
Adım 1	<i>Popülasyon büyüklüğü</i> kadar birey oluştur.
Adım 1.1	İlk bireyi Çizelge 3.5'te verilen algoritma ile oluştur. İkinci birey için dev turu Çizelge 3.6'da verilen algoritma ile oluştur. (<i>Popülasyon büyüklüğü/2</i>)-2 sayıda birey için dev turu Çizelge 3.7'de verilen algoritma ile oluştur. <i>Popülasyon büyüklüğü/2</i> birey için dev turu tamamen rassal olarak oluştur.
Adım 1.2	Her bir dev turu Çizelge 3.8'de verilen algoritma mantığı ile alt turlara böl.
BAŞLA	Jenerasyon sayısı <i>maksimum jenerasyon sayısından</i> küçük ve eşit olduğu sürece aşağıdaki adımları uygula.
Adım 2	Popülasyondaki bireylerin uyum değerlerini hesapla.
Adım 3	<i>Turnuva büyüklüğü</i> kadar rassal olarak seçilen bireye Turnuva Seçim Yöntemini uygula.
Adım 4	<i>Çaprazlama oranını</i> dikkate alarak Sıra Çaprazlama operatörü ile iki yeni birey oluştur.
Adım 5	<i>Mutasyon oranını</i> dikkate alarak yeni bireylere mutasyon operatörünü uygula.
Adım 6	Yeni oluşturulan bireyleri Çizelge 3.8'de verilen algoritma mantığı ile alt turlara böl.
Adım 7	Yeni bireyleri değerlendir. EĞER her iki yeni oluşturulan birey de popülasyonda varsa Rassal olarak seçilen birine mutasyon operatörü uygula ve Adım 7'ye dön. EĞER bireylerden biri popülasyonda varsa Popülasyonda yer almayan bireyi popülasyonla karşılaştırmak üzere seç. EĞER her iki birey de popülasyonda yer almıyorsa En iyi uyum değerine sahip olan bireyi popülasyonla karşılaştırmak üzere seç.
Adım 8	Yerel arama prosedürünü uygula.
Adım 9	Yeni jenerasyon için popülasyonu güncelle.
BİTİR	
Adım 10	En iyi uyum değerine sahip bireyi (çözümü) elde et.

Önerilen MA'ya ilişkin genel yapı verildikten sonra bu bölümde algoritmada kullanılan genetik gösterim, kromozomun genetik kodunun çözümü, seçim yöntemi, genetik operatörler ve yerel arama prosedürü detaylı olarak açıklanmıştır.

Önerilen algoritmada popülasyonda yer alan her bir kromozom tur bölünmesi olmaksızın müşterilere ait genel bir sıralamayı ifade eden permütasyon kodlama ile gösterilmektedir. Yani her bir kromozom sonsuz kapasiteli araç rotalama problemi ya da gezgin satıcı problemine ait dev turu temsil etmektedir. Ancak incelenen problemde araç kapasitesi sınırlı olduğundan aracın tek bir turda tüm müşterilere hizmet vermesi beklenemez. Sonuç olarak hangi müşterilerin hangi tura ait olduğu kararının verilmesi gerekmektedir. Önerilen algoritmada kromozomların genetik kodunun çözümünde Prins (2004) tarafından önerilen algoritmadan yararlanılmıştır. Burada dikkat edilmesi gereken

nokta Prins (2004) tarafından önerilen yaklaşımın sadece toplam kat edilen mesafenin en küçüklenmesini dikkate almasıdır. İncelenen problemde ise toplam gecikmelerin en küçüklenmesi de dikkate alındığından genetik kodun çözümünde Prins (2004) tarafından önerilen yaklaşım problem karakteristiğine göre yeniden uyarlanmıştır. Bu yaklaşıma göre Çizelge 3.8’de verilen algorithmada Adım 5’te ifade edilen hat maliyeti hesaplanırken sadece tur zamanı değil o tur için gerçekleşen gecikme zamanları da dikkate alınmıştır. Bunun yanı sıra tur oluşturulurken mevcut tur için aracın depoya dönüş zamanı da hesaplanarak bir sonraki turun üretiminin tamamlanıp aracın dağıtımına başlayabileceği zaman ile kıyaslanmıştır. Üretimi tamamlanan partinin aracı beklediği durumda bekleme zamanı da öteleme zamanı olarak hat maliyetine dahil edilmiştir. Bu süreç sonraki bölümde detaylı bir şekilde açıklanmıştır.

Kromozomların genetik kodunun çözümü gerçekleştirildikten sonra yani uygun çözüm durumu elde edildikten sonra her bir bireyin uyum değeri hesaplanmıştır. Uyum değeri ilgilenilen problemin Eşitlik (3.1)’de ifade edilen amaç fonksiyonu değeri ile eşdeğerdir.

Başlangıç popülasyonu oluşturulurken kullanılan yöntemlerden en temel ve basit olanı popülasyonun rassal oluşturulmasıdır. Ancak bu tür bir yaklaşım başlangıç popülasyonunun optimal çözümlerden uzak bir noktada olması ve yakınsamanın gecikmesi anlamına gelmektedir. Bu nedenle, popülasyonda iyi bir başlangıç çözümü temsil edecek bireylere de yer verilmiştir. Popülasyonda yer alan ilk birey araç rotalama problemlerinde toplam kat edilen mesafenin en küçüklenmesi için önerilen ve Çizelge 3.5’te adımları verilen Tasarruf Sezgiseli (Clarke ve Wright, 1964) ile oluşturulmuştur. Çizelge 3.5’te görüldüğü gibi algoritma başlatılmadan önce her bir düğüm çifti arasındaki maliyetler (mesafe, zaman) belirlenir (Adım 0). Ardından bağımsız turlarda yer alan herhangi bir düğüm çiftinin aynı turda yer alması durumunda elde edilecek tasarruf hesaplanarak tüm düğümler için tasarruf matrisi oluşturulur (Adım 1). Tasarruf matrisinde yer alan tasarruf miktarları büyükten küçüğe doğru sıralanır ve en büyük tasarrufu sağlayan düğümler sırasıyla belirlenir (Adım 2). Ele alınan düğüm çifti araç kapasitesi de dikkate alınarak düğümlerin daha önce bir rotada yer alıp almaması ve bulunduğu rotanın iç ve dış düğümü olması durumlarına göre farklı şekilde birleştirilir (Adım 3). Mevcut düğümlerle uygun çözüm durumunu koruyacak şekilde herhangi bir birleştirme yapılamıyorsa bir sonraki en büyük tasarrufu sağlayan düğüm çifti seçilerek Adım 3’e dönülür (Adım 4). Herhangi bir rotaya atanan düğümler, düğümler kümesinden

çıkartılır (Adım 5). Döğümler kümesindeki tüm elemanlar bir rotaya atanınca algoritma sonlandırılır ve oluşturulan alt turlar raporlanır (Adım 6).

Çizelge 3.5. Clarke ve Wright'ın Tasarruf Sezgiseli

Adımlar	Açıklama
Adım 0	i ve j döğümleri arası maliyetleri (c_{ij}) belirle.
Adım 1	Tasarruf matrisini $c_{i0} + c_{0j} - c_{ij}$ eşitliği ile oluştur.
Adım 2	Tasarruf miktarlarını (s_{ij}) büyükten küçüğe doğru sırala ve en büyük tasarrufu sağlayan döğümleri belirle.
BAŞLA	Döğümler kümesinde eleman kalmayınca kadar aşağıdaki adımları uygula.
Adım 3	Mevcut döğümleri (i, j) değerlendir. Kapasite sınırlamasını dikkate alarak aşağıdaki şekilde birleştirme işlemini gerçekleştir. EĞER i ve j döğümlerinin her ikisi de henüz bir rotada yer almıyorsa Bu döğümler ile $(0 - i - j - 0)$ şeklinde yeni bir rota oluştur. EĞER i ve j döğümlerinden sadece bir tanesi mevcut bir rotada yer alıyorsa ve bu döğüm mevcut rotada başlangıç noktasıyla bağlantılı (dışta) bir döğümse Diğer döğümü baştaki döğümün ait olduğu rota ile $(0 - j - i - \dots)$ veya $(\dots - j - i - 0)$ şeklinde birleştir. EĞER i ve j döğümlerinden her ikisi de farklı mevcut rotalarda yer alıyorsa ve bu döğümler kendi rotalarında dış döğümle Bu iki rotayı $(0 - i - \dots - j - 0)$ veya $(0 - j - \dots - i - 0)$ şeklinde birleştir.
Adım 4	Mevcut döğümlerle birleştirme işlemi yapılamıyorsa sonraki en büyük tasarrufu sağlayan döğümleri belirle ve Adım 3'e dön.
Adım 5	Birleştirilen döğümleri döğümler kümesinden çıkar.
BİTİR	
Adım 6	Dev turun araç kapasitesine göre bölündüğü alt turları elde et.

Popölasyonda yer alan diğér bireyler üretilirken önce rotala-sonra grupla (route-first cluster-second) yaklaşımı (Beasley, 1983) benimsenmiştir. Bu yaklaşıma göre, ilk aşamada problem araç kapasiteleri dikkate alınmadan gezgin satıcı problemi olarak ele alınarak dev tur (giant tour) olarak ifade edilen tur oluşturulur. İkinci aşamada ise birinci aşamadaki dev tur araç kapasitesine göre farklı yaklaşımlarla alt turlara bölünür. Popölasyonda yer alan bireylerin ilk aşamada dev turlarının oluşturulması için kullanılan algoritmalar En Yakın Komşu Sezgiseli ve Rassal Görev Sezgiseli'dir. En Yakın Komşu Sezgiseli (Rosenkrantz ve ark., 1977) araç rotalama literatüründe sıklıkla karşılaşılan ve uygulaması oldukça kolay olan bir sezgiseldir. Bu yöntemle göre öncelikle başlangıç noktası olarak rassal bir nokta belirlenir (Adım 1). Mevcut döğüme en yakın nokta tura dahil edilir ve bu nokta döğümler kümesinden çıkarılır (Adım 2). Döğümler kümesindeki tüm döğümler ekleninceye kadar Adım 2 tekrarlanır. Son adımda oluşan turun ilk döğümü ve son döğümü birleştirilerek dev tur elde edilir (Adım 3).

Çizelge 3.6. En Yakın Komşu Sezgiseli

Adımlar	Açıklama
Adım 1	Düğüm kümesinden rassal olarak bir düğüm seç.
BAŞLA	Düğüm kümesinde eleman kalmayıncaya kadar aşağıdaki adımları uygula.
Adım 2	Mevcut düğüme en yakın düğümü tura dahil et ve bu noktayı düğümler kümesinden çıkar.
BİTİR	
Adım 3	Oluşan turun ilk ve son düğümlerini birleştirerek dev turu elde et.

En Yakın Komşu sezgiseline rassallık katılarak geliştirilen Rassal Görev Sezgiseli (Prins ve ark., 2009)'nin adımları Çizelge 3.7'de verilmiştir. Bu yöntemle göre öncelikle başlangıç noktası olarak rassal bir nokta belirlenir (Adım 1). Mevcut düğüme en yakın nokta ve en uzak nokta belirlenir. Alt sınır en yakın noktanın uzaklığı üst sınır ise en yakın noktanın uzaklığına en uzak ve en yakın noktaların uzaklıkları arası farkın belirli bir oranı eklenerek elde edilir. Ardından mevcut düğüme olan uzaklıkları bu aralıkta kalan noktalardan rassal olarak bir düğüm seçilir (Adım 2). Seçilen düğüm tura dahil edilerek bu nokta düğümler kümesinden çıkarılır (Adım 3). Düğümler kümesindeki tüm düğümler ekleninceye kadar Adım 2 ve Adım 3 tekrarlanır. Son adımda oluşan turun ilk düğümü ve son düğümü birleştirilerek dev tur elde edilir (Adım 4).

Çizelge 3.7. Rassal Görev Sezgiseli

Adımlar	Açıklama
Adım 1	Düğüm kümesinden rassal olarak bir düğüm seç.
BAŞLA	Düğüm kümesinde eleman kalmayıncaya kadar aşağıdaki adımları uygula.
Adım 2	Mevcut düğüme en yakın düğümü ve en uzak düğümü belirle. Mevcut düğümlerle en yakın düğüm ve mevcut düğümlerle en uzak düğüm arasındaki farkın belirli bir oranı arasındaki bir uzaklıkta bulunan aday düğümleri belirle ve bunlardan birini rassal olarak seç.
Adım 3	Seçilen düğümü tura dahil et ve bu noktayı düğümler kümesinden çıkar.
BİTİR	
Adım 4	Oluşan turun ilk ve son düğümlerini birleştirerek dev turu elde et.

Önce rotala-sonra grupta yaklaşımının ikinci aşamasında kullanılan algoritma ise Prins (2004) tarafından önerilen algoritmaya göre şekillendirilmiştir. Prins (2004) tarafından önerilen algoritmanın adımları Çizelge 3.8'de verilmiştir. Çizelge 3.8'e göre öncelikle herhangi bir düğüme giden yolun maliyeti için başlangıç değerler atanır (Adım1). Ardından önce rotala-sonra grupta yaklaşımının ilk aşamasında elde edilen dev tura ait ilk düğüm, mevcut düğüm olarak atanır (Adım 2). Dev turdaki sıralamaya bağlı kalınarak araç kapasitesini aşmadan mevcut düğüme eklenebilecek olan düğümler kümesi ile olası rotalar belirlenerek (Adım 3) her bir rotaya ilişkin maliyetler hesaplanır (Adım 4). Önceki adımda hesaplanan maliyetlere göre yeni hattın maliyeti güncellenir (Adım 5). Başlangıç noktası mevcut düğüm olacak şekilde tüm olası rotalar için Adım 4 ve Adım 5

düğüm kümesinde eleman kalmayınca kadar tekrarlanır (Adım 6). Son olarak en düşük maliyeti veren rotalar belirlenerek verilen dev tur için optimal tur bölünmesi sağlanmış olur (Adım 7).

Çizelge 3.8. Prins' in Bölme Algoritması

Adımlar	Açıklama
Adım 1	V_i , i . düğüme giden yolun maliyetini ifade etmek üzere, $V_0 = 0$ ve $V_i = \infty$ şeklinde ata.
Adım 2	Dev turun ilk düğümünü mevcut düğüm olarak ata.
BAŞLA	Düğüm kümesinde eleman kalmayınca kadar aşağıdaki adımları uygula.
Adım 3	Dev turdaki sıraya bağlı kalınarak mevcut düğüme araç kapasitesini aşmadan eklenebilecek düğümler kümesi ile olası rotaları belirle.
Adım 4	Bu olası rotalar için $(i - 1, j)$ hattının maliyetini ifade etmek üzere $z_{i-1,j}$ maliyetlerini hesapla.
Adım 5	EĞER $V_{i-1} + z_{i-1,j} < V_j$ ise V_j maliyetini güncelle.
Adım 6	Başlangıç noktası mevcut düğüm olacak şekilde tüm olası rotalar için Adım 4 ve Adım 5 uygulandıktan sonra dev turdaki sıralamaya göre mevcut düğümden sonraki düğümü mevcut düğüm olarak ata ve bu noktayı düğümler kümesinden çıkararak Adım 3'e dön.
BİTİR	
Adım 7	En düşük V_i değerini veren rotaları sondan geriye doğru belirleyerek tur bölünmesini tamamla ve alt turları elde et.

Başlangıç popülasyonu oluşturulduktan sonra yeni bireyleri oluşturacak olan ebeveynler mevcut popülasyon arasından seçilir. Önerilen algorithmada turnuva seçim yöntemi kullanılmıştır. Turnuva seçim yönteminde popülasyon içerisinde belirli sayıda birey rassal olarak alınarak bu bireylerin içerisinde uyum değeri en iyi olan birey ebeveyn olarak seçilmektedir.

Çaprazlama belirli bir olasılıkla seçilen iki bireyin yapılarının rassal olarak birleştirilerek bu bireylerden yeni bireyler oluşturulması sürecidir. Geliştirilen algorithmada araç rotalama ve makine çizelgeleme problemlerinde sıklıkla kullanılan ve uygun çözüm yapısını koruyan sıra çaprazlama operatörü Davis (1985) kullanılmıştır.

Yeni oluşturulan bireylerin belirli bir oranına evrimsel çeşitliliği korumak için mutasyon operatörü uygulanmaktadır. Önerilen MA'da 2-opt-swap operatörü (Croes, 1958) mutasyon operatörü olarak kullanılmıştır.

Yerel arama sürecinde ise farklı komşu arama stratejileri kullanılmıştır. Bunlardan ilki swap operatörüdür. Swap operatörü turların yer değiştirmesi ve müşterilerin yer değiştirmesi şeklinde iki şekilde gerçekleşebilmektedir. Turların yer değiştirmesi, toplam tur zamanı açısından önem arz etmezken geç teslimatlardan kaynaklanan gecikmeler açısından oldukça önemlidir. Bu nedenle mevcut çözümde rassal olarak seçilen turların sırası birbiriyle yer değiştirilerek çözüm kalitesinde iyileşme sağlanabilir. Müşterilerin yer değiştirmesi ise rassal olarak seçilen iki müşterinin birbiri ile yer değiştirmesi yoluyla

gerçekleşmektedir. İkinci komşu arama operatörü olan insert operatörü ile rassal olarak seçilen bir düğüm rassal olarak seçilen bir tur içerisinde rassal bir konuma yerleştirilmektedir. Son komşu arama operatörü olan 2-opt operatörü, aynı turda ve farklı turda olmak üzere iki şekilde gerçekleştirilmektedir. Aynı turda 2-opt operatörü uygulanırken rassal olarak seçilen bir turda rassal olarak iki nokta seçilir ve bu iki nokta arasında yer alan düğümler ters çevrilir. Farklı turda 2-opt operatörü uygulanırken rassal olarak iki tur seçilir. Yine rassal olarak bu turlarda kesme noktaları belirlenir. İlk turda kesme noktasından önce yer alan düğümler ile son turda kesme noktasından sonra yer alan düğümler birleştirilir. Aynı şekilde son turda kesme noktasından önce yer alan düğümler ile ilk turda kesme noktasından sonra yer alan düğümler birleştirilir.

Yerel arama sürecinde bahsedilen operatörler kullanılırken araç kapasitesi göz önünde bulundurulmakta böylece uygun çözüm durumu korunmaktadır. Bu şekilde üretilen komşu çözümler içerisinde en iyi uyum değerine sahip olan çözüm yeni üretilen çözüm olarak popülasyonda yer alan diğer bireylerle karşılaştırılmaktadır.

3.2.3. Algoritma adımlarının örnek test verisine uygulanması

Bu bölümde önerilen algoritmanın daha iyi anlaşılabilmesi için algoritmanın adımları Çizelge 3.2’de parametreleri verilen örnek üzerinde bir jenerasyon uygulanmıştır.

Adım 0: Yapılan test koşulları sonucunda önerilen MA’ da kullanılan algoritma parametreleri Çizelge 3.9’daki gibi belirlenmiştir.

Çizelge 3.9. Önerilen algorithmada kullanılan parametreler

Algoritma parametresi	Değer
Popülasyon büyüklüğü	= 30
Turnuva büyüklüğü	= 3
Çaprazlama oranı	= 0,90
Mutasyon oranı	= 0,10
Maksimum jenerasyon sayısı	= $N \times 50$ (Küçük boyutlu) = $N \times 100$ (Orta boyutlu) = $N \times 200$ (Büyük boyutlu)

Adım 1: Örnek test verisi dikkate alınarak 8 bireyden oluşan bir başlangıç popülasyonu Şekil 3.3’te gösterildiği gibi oluşturulmuştur.

<i>Birey 1:</i>	4	5	2	1	3	6
<i>Birey 2:</i>	3	1	4	5	2	6
<i>Birey 3:</i>	3	1	5	4	2	6
<i>Birey 4:</i>	3	1	5	2	4	6
<i>Birey 5:</i>	5	4	1	3	6	2
<i>Birey 6:</i>	1	6	3	5	4	2
<i>Birey 7:</i>	4	5	1	3	6	2
<i>Birey 8:</i>	2	6	1	3	4	5

Şekil 3.3. Başlangıç popülasyonu

Birey 1, Clarke ve Wright (1964) tarafından önerilen tasarruf algoritmasına göre oluşturulmuş ve (0-4-5-2-0), (0-1-3-0) ve (0-6-0) şeklinde üç tur elde edilmiştir. *Birey 2* için en yakın komşu sezgiseli ile (0-3-1-4-5-2-6-0) şeklinde dev tur oluşturulmuştur. *Birey 3* ve *Birey 4* için rassal görev sezgiseli ile sırasıyla (0-3-1-5-4-2-6-0) ve (0-3-1-5-2-4-6-0) şeklinde dev tur oluşturulmuştur. *Birey 5,6,7 ve 8* için dev turlar tamamen rassal olarak sırasıyla (0-5-4-1-3-6-2-0), (0-1-6-3-5-4-2-0), (0-4-5-1-3-6-2-0) ve (0-2-6-1-3-4-5-0) şeklinde oluşturulmuştur. Görüldüğü gibi *Birey 1* dışındaki tüm bireyler önce rotala-sonra grupla yaklaşımıyla üretilmiştir. Gruplama aşamasında ise Prins (2004) tarafından önerilen algoritma mantığı kullanılmıştır. Şekil 3.3'te görüldüğü gibi her bir tur farklı bir renkle ifade edilmiştir.

Tur bölünmesi için önerilen yaklaşımın daha iyi anlaşılabilmesi için *Birey 2*'de yer alan dev tur (0-3-1-4-5-2-6-0) üzerinde tur bölünmesinin nasıl gerçekleştirildiği anlatılmıştır. Çizelge 3.10' un ilk sütunu dev turdaki sıralama göz önünde bulundurularak araç kapasitesi aşılmayacak şekilde gerçekleştirilecek turları içermektedir. İkinci sütunda ilgili turun tamamlanması için gereken zaman bilgisi yer almaktadır. Üçüncü sütun ilgili turda oluşan gecikmenin hesaplanmasına ilişkin bilgileri içermektedir. Üçüncü sütuna ait ilk sütunda aracın ilgili turdaki müşterilere ulaşma zamanı verilirken ikinci sütunda ilgili müşterilerin teslim tarihi bilgileri yer almaktadır. Üçüncü sütunda ise, ilk iki sütundaki bilgiler kullanılarak turdaki gecikme hesaplanmıştır. Dördüncü sütunda ilgili turun sonraki turu öteleme bilgisi verilmiştir. Öteleme sütununa ilişkin ilk sütun aracın o turu tamamlayıp depoya döndüğü zamanı, ikinci sütun ise o turdan sonra taşınacak olan partinin üretim tamamlanma zamanı bilgilerini içermektedir. Buna göre aracın depoya dönüş anı ve sonraki partinin üretim tamamlanma zamanı arasındaki fark pozitifse çizelge ötelenecektir, bu miktar da son sütunda hesaplanmıştır. Son olarak maliyet sütununda toplam tur zamanı, toplam gecikme ve öteleme miktarları toplanarak tura ilişkin maliyet bilgileri hesaplanmıştır. Toplam tur zamanı ve toplam gecikme direkt olarak araç

fonksiyonunu oluştururken, öteleme miktarının toplam gecikme üzerinde bir etkisinin olduğu açık bir şekilde söylenebilir. Toplam tur uzunluğu için Prins (2004) algoritması ilgili dev tura ilişkin en iyi bölünmeyi garanti ederken, ilgili problem için optimal bölünmeyi garanti edemeyecektir.

Çizelge 3.10. Olası alt turlara ilişkin maliyet hesaplanması

Uygun turlar	Tur zamanı	Gecikme			Öteleme			Maliyet
0-3-0	36	201	472	0	219	271	0	36
0-3-1-0	105	289,332	472,904	0,0	288	332	0	105
0-1-0	88	315	904	0	359	332	27	115
0-1-4-0	139	376,420	904,435	0,0	471	392	79	218
0-1-4-5-0	157	436,480,504	904,435,337	0,45,167	549	488	61	430
0-4-0	102	383	435	0	434	392	42	144
0-4-5-0	120	443,467	435,337	8,130	512	488	24	282
0-4-5-2-0	129	539,563,581	435,337,544	104,226,37	617	550	67	563
0-5-0	90	437	337	100	482	488	0	190
0-5-2-0	99	533,551	337,544	196,7	587	550	37	339
0-5-2-6-0	175	595,613,677	337,544,810	258,69,0	725	-	-	502
0-2-0	72	524	544	0	560	550	10	82
0-2-6-0	148	586,650	544,810	42,0	698	-	-	190
0-6-0	96	598	810	0	646	-	-	96

Ardından Prins (2004) tarafından önerilen algoritma maliyet fonksiyonunun probleme göre yeniden uyarlanmasıyla ilgili dev tura Çizelge 3.11’ deki gibi uygulanmıştır.

Çizelge 3.11. Prins algoritmasının uygulanması

$V_0 = 0;$	Eğer $V_{i-1} + z_{i-1,j} < V_j$ ise
$V_1 = V_2 = V_3 = V_4 = V_5 = V_6 = \infty$	$V_j \leftarrow V_{i-1} + z_{i-1,j}$
$V_0 + 36 < V_1$	$V_1 = 36$
$V_0 + 105 < V_2$	$V_2 = 105$ (0+105) 0-3-1-0
$V_1 + 115 < V_2$	$V_2 = 105$
$V_1 + 218 < V_3$	$V_3 = 254$
$V_1 + 430 < V_4$	$V_4 = 466$
$V_2 + 144 < V_3$	$V_3 = 249$
$V_2 + 282 < V_4$	$V_4 = 387$ (105+282) 0-4-5-0
$V_2 + 563 < V_5$	$V_5 = 668$
$V_3 + 190 < V_4$	$V_4 = 387$
$V_3 + 339 < V_5$	$V_5 = 588$
$V_3 + 502 < V_6$	$V_6 = 751$
$V_4 + 82 < V_5$	$V_5 = 469$ (387+82) 0-2-0
$V_4 + 190 < V_6$	$V_6 = 577$
$V_5 + 96 < V_6$	$V_6 = 565$ (469+96) 0-6-0

Sonuç olarak (0-3-1-4-5-2-6-0) dev turu için hem tur uzunluğu hem de gecikme ve öteleme dikkate alınarak (0-3-1-0), (0-4-5-0), (0-2-0) ve (0-6-0) şeklinde tur bölünmesi sağlanmıştır.

Adım 2: Uyum değeri amaç fonksiyonu değerini ifade edecek şekilde her bir bireyin uyum değeri Çizelge 3.12’deki gibi hesaplanmıştır.

Çizelge 3.12. Popülasyonda yer alan bireylerin uyum değerleri

Bireyler	Toplam Tur Zamanı	Toplam Gecikme	Uyum Değeri
Birey 1:	330	131	461
Birey 2:	393	142	535
Birey 3:	393	130	523
Birey 4:	417	212	629
Birey 5:	412	35	447
Birey 6:	412	319	731
Birey 7:	412	35	447
Birey 8:	484	406	890

Adım 3: Önerilen algorithmda turnuva seçim mekanizması kullanılmıştır. Turnuva büyüklüğü üç olarak belirlendiğinden popülasyondan rassal olarak seçilen üç bireyin (*Birey 2*, *Birey 8*, *Birey 6*) uyum değerleri dikkate alındığında *Birey 2* diğer iki bireye galip gelerek ebeveyn olarak seçilmiştir.

Adım 4: Önerilen MA’da araç rotalama ve makine çizelgeleme problemlerinde sıklıkla kullanılan sıra çaprazlama operatörü kullanılmıştır. Seçilen iki ebeveyne (*Birey 2* ve *Birey 5*) sıra çaprazlama operatörü Şekil 3.4’teki gibi uygulanmıştır. Sıra çaprazlama operatörüne göre öncelikle rassal olarak iki nokta belirlenir. Ebeveynlerde bu iki nokta arasındaki genler çocuklara aynen aktarılır. Ardından diğer ebeveynde yer alan ve çocuklarda bulunmayan genler sırasıyla çocuk kromozomlara kodlanır.

		↓	↓			
Ebeveyn 1:	3	1	4	5	2	6
Ebeveyn 2:	5	4	1	3	6	2
Çocuk 1:	5	1	4	3	6	2
Çocuk 2:	3	4	1	5	2	6

Şekil 3.4. Çaprazlama operatörünün uygulanması

Adım 5: Çaprazlama operatörü uygulandıktan sonra *Çocuk 1*, 2-opt-swap operatörü ile Şekil 3.5’teki gibi mutasyona uğratılmıştır. 2-opt-swap operatörü uygulanırken öncelikle rassal olarak iki nokta belirlenir. Kesme noktaları arasındaki genler ters çevrilerek bu noktalardan önceki ve sonraki genler aynen kalmak şartıyla yeni kromozom oluşturulur.

			↓		↓	
Çocuk 1:	5	1	4	3	6	2
Çocuk 1*:	5	1	2	6	3	4

Şekil 3.5. Mutasyon operatörünün uygulanması

Adım 6: Genetik operatörlerden sonra oluşturulan yeni bireylerin genetik kodunun çözümünde yine başlangıç popülasyonunda olduğu gibi probleme göre uyarlanan Prins (2004) algoritması kullanılarak Şekil 3.6’ da verilen bireyler elde edilmiştir.

Çocuk 1:	5	1	2	6	3	4
Çocuk 2:	3	4	1	5	2	6

Şekil 3.6. Yeni oluşturulan bireyler

Adım 7: Yeni oluşturulan bireyler uygun çözüm durumuna getirildikten sonra uyum değerleri Çizelge 3.13’teki gibi hesaplanmıştır.

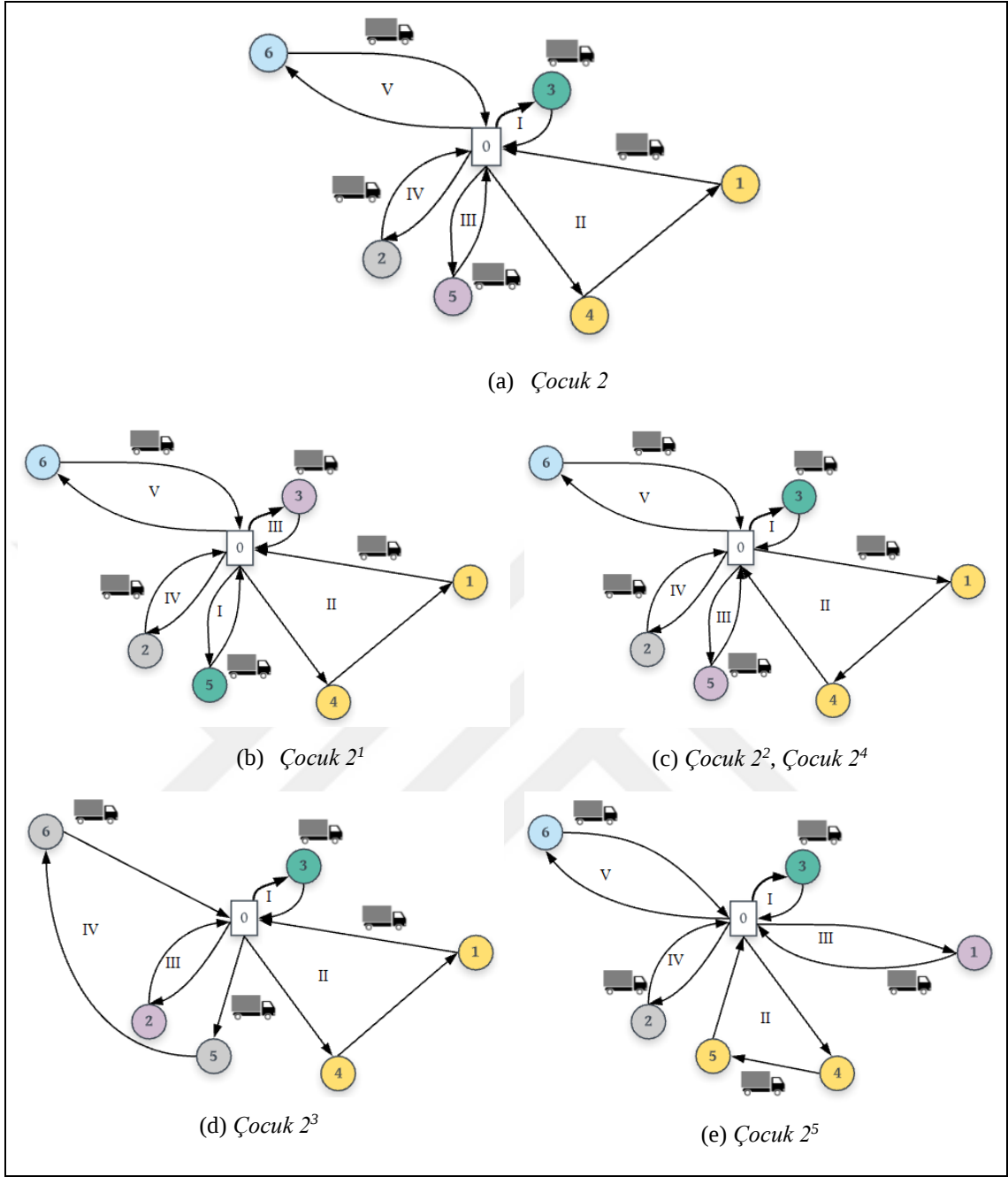
Çizelge 3.13. Yeni oluşturulan bireylerin uyum değerlerinin hesaplanması

Bireyler	Toplam Tur Zamanı	Toplam Gecikme	Uyum Değeri
Çocuk 1:	484	203	687
Çocuk 2:	433	232	665

Adım 8: Önerilen yerel arama sürecinde bahsedilen operatörler Çocuk 2 üzerinde Şekil 3.7’ deki gibi uygulanmıştır. Çizelge 3.14’te uyum değerleri verilen komşu çözümler değerlendirildiğinde en kaliteli çözüm olan Çocuk 2¹ popülasyonla karşılaştırmak üzere seçilmiştir.

Çizelge 3.14. Komşu çözümlerin uyum değerlerinin hesaplanması

Bireyler	Toplam Tur Zamanı	Toplam Gecikme	Uyum Değeri
Çocuk 2 ¹	433	0	433
Çocuk 2 ²	433	232	665
Çocuk 2 ³	420	258	678
Çocuk 2 ⁴	433	232	665
Çocuk 2 ⁵	412	46	458



Şekil 3.7. Yerel arama prosedürü

Adım 9: Yeni oluşturulan bireyler arasında en iyi uyum değerine sahip birey (Çocuk 2¹) popülasyonda en kötü uyum değerine sahip olan bireyle (Birey 8) yer değiştirir. Bu durumda bir jenerasyon sonunda güncellenen yeni popülasyon Şekil 3.8’de verilmiştir. Algoritma bir jenerasyon uygulandıktan sonra popülasyon uyum değeri ortalaması 582,88’den 525,75’e düşmüştür.

Birey 1:	4	5	2	1	3	6
Birey 2:	3	1	4	5	2	6
Birey 3:	3	1	5	4	2	6
Birey 4:	3	1	5	2	4	6
Birey 5:	5	4	1	3	6	2
Birey 6:	1	6	3	5	4	2
Birey 7:	4	5	1	3	6	2
Birey 8:	5	4	1	3	2	6

Şekil 3.8. Güncel popülasyon

3.2.4. PAT_TA Problemi için test problemlerinin üretilmesi

Tez kapsamında PAT_TA Probleminin çözümü için geliştirilen yöntemlerin değerlendirilmesi amacıyla yeni test problemleri üretilmiştir. Bu bölümde test problemlerinin yapısı ve üretilme mekanizması anlatılmıştır. Çizelge 3.15'te görüldüğü gibi üretilen test problemleri müşteri ve makine sayılarına göre küçük, orta ve büyük boyutlu problemler olmak üzere üç grupta incelenmiştir. Bunun dışında dağıtım aşamasında müşterilerin belirledikleri son teslim tarihleri sıkı, orta ve gevşek olmak üzere üç farklı seviyede incelenmiştir. Üretim aşamasında ise ürünlerin makinelerdeki işlem sürelerindeki değişkenliğe göre iki farklı seviye ele alınmıştır. Her bir kombinasyon için beş örnek üretilmiştir. Böylece 180 küçük boyutlu, 540 orta boyutlu ve 360 büyük boyutlu olmak üzere toplamda 1080 örnek incelenmiştir.

Çizelge 3.15. PAT_TA Problemi için geliştirilen test problemlerinin yapısı

Örnek	N	M	V	dd	π
Küçük boyutlu	5,6,7	2,3	1	Sıkı, orta, gevşek	0,1, 0,3
Orta boyutlu	10,11,12,13,14,15	3,4,5	1	Sıkı, orta, gevşek	0,1, 0,3
Büyük boyutlu	20,30,40,50	4,5,6	1	Sıkı, orta, gevşek	0,1, 0,3

Yapılan tez çalışmasında kullanılan test problemleri Ullrich (2013) tarafından yapılan çalışmaya benzer bir şekilde üretilmiştir. Çizelge 3.16'da problemde kullanılan parametrelerin üretilme süreci verilmiştir. Test problemlerinin üretilme süreci genel bir ρ değerinin belirlenmesiyle başlamıştır. İlk olarak müşterilere ait talepler Eşitlik (3.25) ile belirlenmiştir. Ardından her bir ürünün her bir makinedeki birim üretim zamanları parametreleri $(1 - \mu)$ ve $(1 + \mu)$ olan düzgün dağılıma uyacak şekilde üretilmiştir. μ değeri arttıkça seçilen bir siparişin makinelerdeki işlem süreleri arasındaki değişkenlik de artacaktır. Her bir müşteriye ait siparişlerin üretimi için gereken zaman ise birim üretim zamanı ile müşteri talebinin çarpımı ile Eşitlik (3.26)'da görüldüğü gibi elde edilmektedir. Eşitlik (3.27)'de araç kapasitesi belirlenirken yine düzgün dağılımdan

yararlanılmış ve uygun çözüm durumunu korumak için alt sınır maksimum müşteri talebi olarak ayarlanmıştır. Üst sınır ise toplam talebin yarısı olacak şekilde seçilmiştir. Sistemde yer alan düğümler arası zamanlar üretilirken bütünleşik problemi oluşturan makine çizelgeleme ve araç rotalama problemlerinin birbirine baskın gelmemesi için üretim ve dağıtım süreleri arasında bir ödünleşme sağlanmaya çalışılmıştır. Bu yaklaşımda ilk olarak iki düğüm arasındaki en büyük uzaklık için bir eşik değeri (α) Eşitlik (3.28) ile belirlenmiştir. Bu değer belirlenirken üretimde ve dağıtımda geçirilen süreler kestirilmeye çalışılmıştır. Eşitlik (3.30)'da depo düğümün koordinatları belirlenmiştir. Ardından sistemde yer alan müşteriler arasındaki süre bu eşik değerini geçmeyecek şekilde, müşteriler rassal olarak X ve Y koordinatlarına Eşitlik (3.31) ile konumlandırılmıştır. Müşterilerin son teslim tarihleri belirlenirken en iyi durumda sistemde yer alan ilk müşteri oldukları düşünülmüş ve ilgili müşteri siparişinin tüm makinelerdeki işlem süreleri toplamı ile depodan müşteriye gidilen sürenin toplamı alt sınır olarak belirlenmiştir. Sistemde geçen süre (π) üretim ve dağıtımda geçirilen sürelerin toplamı $\pi = \rho[(M + N - 1) + (N + 1)]$ olarak kestirilmiş ve teslim tarihleri sıkı, orta ve gevşek olacak şekilde farklı β seviyelerine göre Eşitlik (3.32)'deki gibi üretilmiştir.

Çizelge 3.16. İncelenen problem parametrelerinin üretilmesi

Parametre	Değer	
d_i	$\sim UNIF\left(\frac{\rho}{2}, \rho\right)$	$\rho = 100$ (3.25)
p_{im}	$\sim UNIF(1 - \mu, 1 + \mu)d_i$	(3.26)
Q	$\sim UNIF\left(\max d_i, \frac{\sum d_i}{2}\right)$	(3.27)
α	$= \left\lfloor \frac{3\rho(M + N - 1)}{2(N + 1)} \right\rfloor$	(3.28)
ω	$= \sqrt{\frac{\alpha^2}{2}}$	(3.29)
(X_0, Y_0)	$= \left(\frac{\omega}{2}, \frac{\omega}{2}\right)$	(3.30)
(X_i, Y_i)	$\sim UNIF(0, \omega)$	(3.31)
dd_i	$dd_i = \sum_m p_{im} + t_{oi} + UNIF(0, \beta\pi)$	$\beta = 1/4, 1/3, 1/2$ (3.32)

3.3. PAT_TA Problemi için Karşılaştırmalı Sonuçlar

Bu bölüm önerilen KTDP modelinin performansı ile önerilen MA ve CPLEX ile elde edilen sonuçların karşılaştırılması olmak üzere iki başlıkta incelenmiştir. Önerilen yöntemleri analiz etmek amacıyla Çizelge 3.17'deki performans kriterleri kullanılmıştır.

Çizelge 3.17. Geliştirilen yöntemler için kullanılan performans kriterleri

İfade	Açıklama
GSD	: Görelî sapma değeri
OÇUPS	: CPLEX tarafından verilen süre sınırları içinde optimal çözüme ulaşılan problem sayısı
OÇS	: Matematiksel model ve önerilen algoritmaya ait ortalama saniye cinsinden koşum süresi
YSD	: Yüzde sapma değeri

Geliştirilen matematiksel model GAMS 28.2 ara yüzünde kodlanmış ve matematiksel model çözücüsü olarak CPLEX kullanılmıştır. CPLEX için verilen süre sınırı 10800 saniye (sn) olarak ayarlanmıştır.

Çizelge 3.17’de ifade edilen GSD, CPLEX ile MA’nın görelî performanslarını değerlendirmektedir. Bu şekilde eğer CPLEX ile elde edilen çözüm (Z^{CPLEX}), MA ile elde edilen çözümden (Z^{MA}) daha iyi ise; MA’nın CPLEX’e göre görelî performansı Eşitlik (3.33) ile değerlendirilmektedir. MA ile üç koşumda elde edilen en iyi çözüm, en kötü çözüm ve çözüm ortalaması için sırasıyla GSD^{enk} , GSD^{enb} ve GSD^{ort} değerleri hesaplanmaktadır. Tam tersi durumda ise CPLEX’in MA’ya göre görelî performansı Eşitlik (3.34) ile değerlendirilmektedir.

$$GSD = 100 \times \left(\frac{Z^{MA} - Z^{CPLEX}}{Z^{CPLEX}} \right) \quad (3.33)$$

$$GSD = 100 \times \left(\frac{Z^{CPLEX} - Z^{MA}}{Z^{MA}} \right) \quad (3.34)$$

Çizelge 3.17’de ifade edilen YSD ise verilen süre sınırı içerisinde CPLEX ile elde edilen üst sınır ($Z^{\bar{U}S}$) ile alt sınır (Z^{AS}) arasındaki sapma değerini ifade etmektedir ve Eşitlik (3.35) ile hesaplanmaktadır.

$$YSD = 100 \times \left(\frac{Z^{\bar{U}S} - Z^{AS}}{Z^{\bar{U}S}} \right) \quad (3.35)$$

3.3.1. Önerilen KTDP modelinin sonuçları

Bu bölümde önerilen matematiksel modelin küçük, orta ve büyük boyutlu problemlere göre genel performansı değerlendirilmiştir. Çizelge 3.18’de CPLEX ile elde edilen sonuçlar verilmiştir. Çizelgenin ilk iki sütunu sırasıyla örnekteki müşteri ve makine sayısını ifade etmektedir. Üçüncü ve dördüncü sütunda sırasıyla CPLEX tarafından verilen süre sınırı içerisinde elde edilen üst sınır ve alt sınır ortalaması verilmiştir. Beşinci

sütunda yüzde sapma değeri ortalaması verilirken bu değerin sıfır olması optimal çözümün elde edildiği anlamına gelmektedir. Altıncı sütunda her müşteri makine kombinasyonunda yer alan tüm örnekler içinde optimal çözüme ulaşılan problem sayısı verilirken son sütunda ise saniye cinsinden ortalama çözüm süresi verilmiştir.

Çizelge 3.18. CPLEX ile elde edilen sonuçlar

	<i>N</i>	<i>M</i>	<i>Z</i> ^{ÜS}	<i>Z</i> ^{AS}	YSD	OÇUPS	OÇS
<i>Küçük</i>	5	2	519,47	519,47	0,00	30/30	0,39
		3	623,93	623,93	0,00	30/30	0,32
	6	2	460,27	460,27	0,00	30/30	3,08
		3	786,40	786,40	0,00	30/30	3,54
	7	2	617,47	617,47	0,00	30/30	49,28
		3	775,17	775,17	0,00	30/30	33,60
		Ort.	630,45	630,45	0,00	180/180	15,04
<i>Orta</i>	10	3	1188,73	869,85	23,61	15/30	7091,41
		4	1121,60	621,28	37,36	6/30	8976,39
		5	1478,40	722,22	40,10	8/30	8522,76
	11	3	1264,83	558,34	45,35	3/30	10337,17
		4	1434,10	695,01	42,31	6/30	10062,91
		5	1440,87	892,91	28,90	11/30	7345,90
	12	3	1631,03	513,84	57,07	1/30	10757,46
		4	1744,97	831,12	40,08	6/30	8792,25
		5	1757,47	817,18	45,18	3/30	9876,02
	13	3	1841,93	690,43	50,41	2/30	10081,96
		4	1680,73	564,66	59,56	0/30	10800,00
		5	1959,77	696,33	56,54	2/30	10083,93
	14	3	2138,50	741,91	53,73	0/30	10800,00
		4	2093,80	871,33	47,79	6/30	8683,03
		5	1950,73	739,65	53,98	0/30	10800,00
	15	3	2075,80	632,94	57,25	4/30	9816,08
		4	2344,73	508,15	71,02	0/30	10800,00
		5	2416,93	948,41	49,05	3/30	10009,33
		Ort.	1753,61	717,53	47,74	76/540	9846,48
<i>Büyük</i>	20	4	3536,30	717,66	73,03	0/30	10800,00
		5	3824,83	720,03	73,94	0/30	10800,00
		6	3714,90	624,38	74,19	0/30	10800,00
	30	4	7468,43	879,03	79,83	0/30	10800,00
		5	7812,00	556,19	87,05	0/30	10800,00
		6	10108,07	1168,02	82,96	0/30	10800,00
	40	4	18155,67	585,70	94,01	0/30	10800,00
		5	18321,00	629,36	93,94	0/30	10800,00
		6	20071,17	639,63	96,27	0/30	10800,00
	50	4	30462,50	1236,79	94,98	0/30	10800,00
		5	32028,43	648,41	97,65	0/30	10800,00
		6	31376,63	1126,77	95,46	0/30	10800,00
		Ort.	15573,33	794,33	86,94	0/360	10800,00

Çizelge 3.18’de görüldüğü gibi CPLEX küçük boyutlu problemlerin hepsinde oldukça kısa bir sürede optimal çözümlere ulaşabilmiştir. Orta boyutlu problemlerde müşteri sayısının artmasıyla verilen süre sınırları içerisinde bulunan optimal çözüm sayısı büyük oranda azalmış, çözüm süresi ise artış göstermiştir. Büyük boyutlu problemlerde

ise CPLEX hiçbir örnekte verilen süre sınırı içerisinde optimal çözüme ulaşamamıştır. Bununla birlikte müşteri sayısının artmasıyla YSD değerinde bariz bir artış görülmektedir. Çizelge 3.18'in bir diğer çıkarımı ise makine sayısının YSD ve OÇS'ye doğrudan bir etkisinin olmadığıdır.

3.3.2. Önerilen MA'nın performansı

Bu bölüm, üretim sürecindeki değişkenliğin ve teslim tarihlerinin problem üzerindeki etkisini incelemek amacıyla iki alt başlıkta ele alınmıştır.

3.3.2.1. Üretim sürecindeki değişkenliğin etkisi

Bu bölümde üretim sürecindeki değişkenliğin CPLEX ve MA üzerindeki etkisi araştırılmıştır. Çizelge 3.19'un ilk sütunu üretim sürelerindeki değişkenlik katsayısını (μ) ifade etmektedir. Önceki bölümde bahsedildiği gibi μ değerinin artmasıyla aynı işin farklı makinelerdeki işlem zamanına ilişkin değişkenlik de artacaktır. İkinci sütunda müşteri sayıları yer almaktadır. Önerilen MA her bir örneğe ilişkin üç koşul çalıştırılmıştır. Çizelgenin üçüncü, dördüncü ve beşinci sütunu sırasıyla bu koşullarda elde edilen en iyi çözüme, en kötü çözüme ve çözümlerin ortalamasına ilişkin MA'nın CPLEX'e göre GSD değerlerini ifade etmektedir. Çizelgenin altıncı ve dokuzuncu sütununda sırasıyla önerilen MA ve CPLEX ile elde edilen optimal çözüm sayıları verilirken yedinci ve onuncu sütunda sırasıyla önerilen MA ve CPLEX'in saniye cinsinden ortalama çözüm süreleri verilmiştir. Sekizinci sütun ise CPLEX'in MA'ya göre GSD değerlerini ifade etmektedir.

Çizelge 3.19'da görüldüğü gibi CPLEX incelenen toplam 1080 örneğin 256 tanesinde verilen süre sınırı içerisinde optimal çözümü elde edebilmiştir. Optimal çözümü bilinen 256 örneğin 243 tanesinde önerilen MA ile saniyelerle ifade edilen sürelerde optimal çözümler elde edilebilmiştir. Orta boyutlu problemlerde CPLEX ve MA'nın göreceli performansları birbirine oldukça yakın bulunurken, müşteri sayısı arttıkça CPLEX'in göreceli performansının azaldığı görülmektedir. Büyük boyutlu problemlerde ise önerilen MA'nın performansı CPLEX'e göre daha iyi bulunmuştur. Özellikle müşteri sayısının 40 ve 50 olduğu durumlarda GSD değerlerine göre MA'yla elde edilen çözümler CPLEX ile elde edilen çözümlere göre oldukça başarılı bulunmuştur.

Çizelge 3.19. Farklı μ seviyelerine göre MA ve CPLEX' in performansı

μ	N	MA					CPLEX		
		GSD ^{enk}	GSD ^{enb}	GSD ^{ort.}	OÇUPS	OÇS	GSD	OÇUPS	OÇS
0,1	5	0,00	0,08	0,06	30/30	1,50	0,00	30/30	0,52
	6	0,06	0,11	0,09	29/30	1,32	0,00	30/30	3,85
	7	0,06	0,28	0,17	28/30	1,23	0,00	30/30	44,44
	Ort.	0,04	0,16	0,11	87/90	1,35	0,00	90/90	16,27
	10	0,23	0,48	0,37	14/15	2,26	0,20	15/45	8187,65
	11	0,14	0,26	0,18	11/11	2,74	0,32	11/45	9154,68
	12	0,09	0,52	0,23	6/6	3,25	0,26	6/45	9720,72
	13	0,02	0,46	0,21	2/2	3,70	0,86	2/45	10320,79
	14	0,29	1,03	0,54	1/3	4,35	0,86	3/45	10100,31
	15	0,11	1,13	0,53	2/3	5,18	1,14	3/45	10297,29
	Ort.	0,15	0,65	0,34	36/40	3,58	0,61	40/270	9630,24
	20	0,08	0,85	0,27	-	9,11	4,14	0/45	10800,00
	30	0,00	0,00	0,00	-	20,94	39,05	0/45	10800,00
	40	0,00	0,00	0,00	-	34,26	149,78	0/45	10800,00
	50	0,00	0,00	0,00	-	57,77	292,44	0/45	10800,00
	Ort.	0,02	0,21	0,07	-	30,52	121,35	0/180	10800,00
0,3	5	0,00	0,00	0,00	30/30	1,88	0,00	30/30	0,20
	6	0,00	0,00	0,00	30/30	1,65	0,00	30/30	2,77
	7	0,21	0,50	0,35	26/30	2,02	0,00	30/30	38,44
	Ort.	0,07	0,17	0,12	86/90	1,85	0,00	90/90	13,80
	10	0,06	0,30	0,17	14/14	3,57	0,19	14/45	8206,06
	11	0,06	0,14	0,11	9/9	4,04	0,32	9/45	9342,64
	12	0,06	0,40	0,18	4/4	3,47	0,48	4/45	9896,43
	13	0,12	0,46	0,21	2/2	3,69	1,07	2/45	10323,17
	14	0,09	0,46	0,21	2/3	4,29	1,53	3/45	10088,41
	15	0,05	0,30	0,15	3/4	5,10	2,08	4/45	10119,68
	Ort.	0,07	0,34	0,17	34/36	4,03	0,95	36/270	9662,73
	20	0,01	0,30	0,08	-	7,97	6,81	0/45	10800,00
	30	0,00	0,00	0,00	-	18,53	38,05	0/45	10800,00
	40	0,00	0,00	0,00	-	33,28	123,36	0/45	10800,00
	50	0,00	0,00	0,00	-	52,56	264,37	0/45	10800,00
	Ort.	0,00	0,08	0,02	-	28,09	108,15	0/180	10800,00

Üretim sürecindeki değişkenliğin etkisi incelendiğinde μ değerinin 0,1 ve 0,3 olduğu durumlarda orta boyutlu problemlerde CPLEX'in elde ettiği optimal çözüm sayısı sırasıyla 40 ve 36 olarak, ortalama çözüm süreleri ise 9630,24 ve 9662,73 sn olarak bulunmuştur. Böylece üretim sürecindeki değişkenliğin problemi bir miktar zorlaştırdığı ancak bu etkinin oldukça sınırlı olduğu görülmüştür.

3.3.2.2. Teslim tarihlerinin etkisi

İncelenen problemde araştırılması gereken bir diğer parametre müşterilerin belirlemiş oldukları son teslim tarihleridir. Amaç fonksiyonunda yer alan toplam gecikme zamanı müşterilerin teslim tarihlerine göre belirlendiğinden bu bölümde farklı teslim tarihi seviyelerinin problem üzerindeki etkisi tartışılmıştır. Bu amaçla her bir örnek için gevşek, orta ve sıkı olmak üzere üç farklı seviye teslim tarihi belirlenmiştir. İlk sütun

hariç Çizelge 3.20’de yer alan tüm sütunlar Çizelge 3.19’daki sütunlarla aynıdır. İlk sütunda bu kez teslim tarihi seviyeleri yer almıştır.

Çizelge 3.20. Farklı teslim tarihi seviyelerine göre MA ve CPLEX’ in performansı

dd_t	N	MA					CPLEX		
		GSD ^{enk}	GSD ^{enb}	GSD ^{ort.}	OÇUPS	OÇS	GSD	OÇUPS	OÇS
Geçek	5	0,00	0,00	0,00	20/20	1,81	0,00	20/20	0,23
	6	0,08	0,16	0,14	19/20	1,56	0,00	20/20	1,71
	7	0,17	0,18	0,18	19/20	1,49	0,00	20/20	27,71
	Ort.	0,08	0,11	0,11	58/60	1,62	0,00	60/60	9,88
	10	0,10	0,22	0,18	18/19	2,69	0,22	19/30	5362,11
	11	0,00	0,01	0,00	16/16	3,12	0,20	16/30	7046,52
	12	0,15	0,77	0,39	9/9	3,14	0,28	9/30	8029,91
	13	0,00	0,29	0,12	4/4	3,59	0,45	4/30	9365,87
	14	0,35	0,62	0,49	3/6	3,88	0,44	6/30	8683,02
	15	0,10	0,63	0,35	5/7	4,56	1,10	7/30	9025,39
	Ort.	0,12	0,42	0,26	55/61	3,50	0,45	61/180	7918,80
	20	0,01	0,67	0,20	-	7,56	4,94	0/30	10800,00
	30	0,00	0,00	0,00	-	16,73	47,75	0/30	10800,00
	40	0,00	0,00	0,00	-	27,10	255,04	0/30	10800,00
	50	0,00	0,00	0,00	-	42,76	612,70	0/30	10800,00
	Ort.	0,00	0,17	0,05	-	23,54	230,11	0/120	10800,00
Orta	5	0,00	0,00	0,00	20/20	1,76	0,00	20/20	0,38
	6	0,00	0,00	0,00	20/20	1,57	0,00	20/20	3,14
	7	0,01	0,34	0,22	19/20	1,82	0,00	20/20	19,60
	Ort.	0,00	0,11	0,07	59/60	1,72	0,00	60/60	7,71
	10	0,20	0,53	0,35	6/6	3,02	0,18	6/30	9103,15
	11	0,17	0,25	0,21	4/4	3,55	0,47	4/30	9899,42
	12	0,05	0,49	0,17	1/1	3,42	0,47	1/30	10595,80
	13	0,01	0,31	0,14	-	3,68	1,22	0/30	10800,00
	14	0,12	1,06	0,40	-	4,56	1,62	0/30	10800,00
	15	0,00	0,98	0,43	-	5,41	2,15	0/30	10800,00
	Ort.	0,09	0,60	0,28	11/11	3,94	1,02	11/180	10333,06
	20	0,00	0,36	0,03	-	8,92	6,42	0/30	10800,00
	30	0,00	0,00	0,00	-	20,83	44,21	0/30	10800,00
	40	0,00	0,00	0,00	-	37,86	113,20	0/30	10800,00
	50	0,00	0,00	0,00	-	61,03	154,43	0/30	10800,00
	Ort.	0,00	0,09	0,01	-	32,16	79,57	0/120	10800,00
Sıkı	5	0,00	0,13	0,09	20/20	1,49	0,00	20/20	0,46
	6	0,00	0,00	0,00	20/20	1,32	0,00	20/20	5,08
	7	0,22	0,65	0,39	16/20	1,56	0,00	20/20	77,01
	Ort.	0,07	0,26	0,16	56/60	1,46	0,00	60/60	27,52
	10	0,13	0,41	0,27	4/4	3,05	0,19	4/30	10125,30
	11	0,14	0,34	0,22	-	3,51	0,30	0/30	10800,00
	12	0,03	0,13	0,05	-	3,52	0,36	0/30	10800,00
	13	0,21	0,77	0,37	-	3,82	1,22	0/30	10800,00
	14	0,11	0,57	0,22	-	4,51	1,51	0/30	10800,00
	15	0,13	0,53	0,24	-	5,44	1,59	0/30	10800,00
	Ort.	0,13	0,46	0,23	4/4	3,98	0,86	4/180	10687,55
	20	0,12	0,69	0,31	-	9,14	5,06	0/30	10800,00
	30	0,00	0,00	0,00	-	21,64	23,69	0/30	10800,00
	40	0,00	0,00	0,00	-	39,70	41,46	0/30	10800,00
	50	0,00	0,00	0,00	-	61,70	68,08	0/30	10800,00
	Ort.	0,03	0,17	0,08	-	33,05	34,57	0/120	10800,00

Çizelge 3.20 incelendiğinde teslim tarihlerinin CPLEX'in başarısı üzerindeki etkisini araştırmak için ilk olarak orta boyutlu problemler incelenmiştir. Bunun sebebi CPLEX'in küçük boyutlu örneklerin hepsinde optimal çözüme ulaşması, büyük boyutlu problemlerde ise hiçbirinde optimal çözüm elde edememesidir. Orta boyutlu problemlerde CPLEX tarafından elde edilen optimal çözüm sayıları teslim tarihlerinin gevşek, orta ve sıkı olduğu durumlar için sırasıyla 61, 11 ve 4 olarak bulunmuştur. Çizelge 3.20'den müşterilerin belirledikleri son teslim tarihleri daraldıkça bulunan optimal çözüm sayısının ciddi oranda azaldığı ortalama çözüm süresinin ise hızla arttığı görülmektedir. Amaç fonksiyonu içerisinde toplam gecikme zamanı yer aldığından herhangi bir müşteri için gecikmenin sıfırdan farklı olması problemi zorlaştırmaktadır. Büyük boyutlu problemlerde ise CPLEX'in MA'ya göre performansı GSD değerleri incelendiğinde teslim tarihlerinin gevşek olduğu durumda en kötü seviyededir. Bunun sebebi olarak sıkı teslim tarihi seviyesinde önerilen MA'nın da CPLEX gibi zorlanması böylece CPLEX'in MA'ya göre GSD değerinin daha az bulunması verilebilir.

4. PERMÜTASYON AKIŞ TİPİ ÜRETİM ORTAMI İLE ÇOK ARAÇLI DAĞITIM ORTAMINI İÇEREN BÜDÇ (PAT_ÇA) PROBLEMİ

Tez kapsamında incelenen ilk problemin dağıtım ortamında tek bir araç hizmet vermektedir. Bu durum gerçek hayat problemleri açısından değerlendirildiğinde çok da olası değildir. Bu nedenle tez kapsamında incelenen ikinci problemin dağıtım ortamında sistemde birden fazla aracın olduğu durum incelenmiştir. Kaynak araştırması bölümünde belirtildiği gibi literatürde BÜDÇ Problemi için en fazla çalışılan dağıtım ortamı sistemde sınırlı sayıda homojen aracın olduğu durumdur. Bu nedenle tez kapsamında ele alınan ikinci problemde dağıtım ortamında heterojen kapasiteli araçlar incelenmiştir. Literatürde dikkat çeken bir diğer eksiklik ise araç sayısının birden fazla olduğu durumu konu edinen kaynakların büyük çoğunluğunda sistemde yer alan araçlar planlama periyodu boyunca sadece bir kez kullanılabilmektedir. Bu durum yine gerçeği yansıtmamaktadır. Bu nedenle PAT_ÇA Problemi olarak adlandırılan problemde sistemde yer alan sınırlı sayıdaki heterojen kapasiteli aracın planlama periyodu boyunca birden fazla kez kullanımına izin verilmiştir.

Bu bölüm PAT_ÇA Probleminin tanımı, PAT_ÇA Problemi için önerilen yöntemler ve karşılaştırmalı sonuçlar olmak üzere üç alt başlıkta incelenmiştir.

4.1. PAT_ÇA Probleminin Tanımı

PAT_ÇA Problemi temelde PAT_TA Problemi ile aynı üretim ortamına sahiptir. Yani N adet siparişin üretimi yine seri olarak yerleştirilmiş M adet makineden oluşan bir üretim ortamında gerçekleşecek ve siparişlerin her makinedeki işlenme sırası da aynı olacaktır. Üretimi tamamlanmış siparişlerin müşterilere teslimatı ise PAT_TA Probleminden farklı olarak sınırlı sayıda araçla yapılmaktadır. Dağıtım sürecinde hizmet veren her bir k aracı sınırlı bir kapasiteye Q_k sahiptir ve planlama periyodu boyunca birden fazla kez kullanılabilmektedir.

Böylece incelenen problem sonucunda (i) müşteri siparişlerinin makinelerdeki operasyon sıralarının belirlenmesi, (ii) her bir müşteri siparişinin üretime başlama ve üretim tamamlanma zamanının belirlenmesi, (iii) belirli bir turda hangi aracın hizmet vereceğinin belirlenmesi (iv) her bir aracın dağıtım yapacağı müşterilerin ziyaret sıralarının belirlenmesi, (v) her bir müşteri siparişinin teslimat zamanının belirlenmesine ilişkin kararlar alınmaktadır.

PAT_ÇA Probleminin daha iyi anlaşılabilmesi amacıyla öncelikle ele alınan problem için çok amaçlı KTP modeli geliştirilmiş ardından PAT_ÇA Problemine ilişkin örnek bir çözüm incelenmiştir.

4.1.1. PAT_ÇA Problemi için önerilen KTP modeli

Bu bölümde PAT_ÇA Problemi için önerilen KTP modelinde kullanılan simgeler, model ve modelde yer alan kısıtların açıklamaları detaylarıyla verilmiştir. Önerilen KTP modelinde kullanılan simgeler Çizelge 4.1’de verilmiştir.

Çizelge 4.1. Önerilen KTP modelinde kullanılan simgeler

Simge	Açıklama
<i>İndisler:</i>	
i, j	: İş (sipariş, müşteri)
m	: Makine
k	: Araç
N_C	: İşler (siparişler, müşteriler) kümesi
N	: Tüm düğümler kümesi ($N_C \cup 0$)
M	: Makineler kümesi
V	: Araçlar kümesi
d_i	: i . müşteri siparişine ait talep miktarı
p_{im}	: i . müşteri siparişinin m . makinedeki işlem süresi
t_{ij}	: i . ve j . düğümler arası ulaşım süresi
dd_i	: i . müşteri siparişine ait teslim tarihi
Q_k	: k . araç kapasitesi
H	: Yeterince büyük bir sayı
X_{ijk}	: 1, k . araç i . düğümden hemen sonra j . düğüme gidiyorsa; 0, aksi durumda
W_{ijk}	: 1, k . araç önceki turu i . düğümden tamamlayıp sonraki tura j . düğüme
O_{ij}	: 1, önceki tur i . düğümden tamamlanıp sonraki tur j . düğüme başlıyorsa; 0
Z_{ijk}	: 1, k . araçla taşınan i . ürünün üretimi j . ürünün üretiminden önce
u_i	: i . müşteriye ziyaret etmeden hemen önce araç üzerindeki toplam yük
S_{im}	: i . müşteri siparişinin m . makinedeki üretime başlama zamanı
C_{im}	: i . müşteri siparişinin m . makinedeki üretim bitiş zamanı
A_i	: aracın i . müşteriye varış zamanı
Y_i	: i . müşteri siparişinin ait olduğu partinin üretim bitiş zamanı
T_i	: i . müşteri siparişi için gecikme miktarı
<i>Yardımcı Değişkenler:</i>	
$AX1_{ijk}$	$= u_j X_{ijk}$
$AX2_{ijk}$	$= u_j X_{jik}$
$AX3_{ijk}$	$= u_i X_{ijk}$
$AX4_{ijk}$	$= u_i X_{jik}$
$AX5_{ijk}$	$= X_{i0k} X_{0ik}$
$AX6_{ijk}$	$= C_{im} AX5_{ijk}$
$AX7_{ijk}$	$= C_{jm} AX5_{ijk}$

$$\min \sum_{i \in N_C} T_i + \sum_{i \in N} \sum_{j \in N_C; j \neq i} \sum_{k \in V} t_{ij} X_{ijk} \quad (4.1)$$

$$\sum_{k \in V} \sum_{i \in N} X_{ijk} = 1 \quad \forall j \in N_C \quad (4.2)$$

$$\sum_{j \in N} X_{ijk} = \sum_{j \in N} X_{jik} \quad \forall i \in N; k \in V \quad (4.3)$$

$$(u_j - u_i)(X_{ijk} + X_{jik}) + Q_k X_{ijk} + (Q_k - d_i - d_j) X_{jik} \leq (Q_k - d_i)(X_{ijk} + X_{jik}) \quad \forall i, j \in N_C; k \in V \quad (4.4)$$

$$u_i \geq d_i + \sum_{k \in V} \sum_{j \in N_C; j \neq i} d_j X_{ijk} \quad \forall i \in N_C \quad (4.5)$$

$$u_i \leq \sum_{k \in V} \sum_{j \in N_C; j \neq i} Q_k X_{ijk} + \sum_{k \in V} d_i X_{i0k} \quad \forall i \in N_C \quad (4.6)$$

$$\sum_{j \in N_C; j \neq i} W_{ijk} \leq X_{i0k} \quad \forall i \in N_C; k \in V \quad (4.7)$$

$$\sum_{i \in N_C; j \neq i} W_{ijk} \leq X_{0jk} \quad \forall j \in N_C; k \in V \quad (4.8)$$

$$\sum_{j \in N_C} X_{0jk} - \sum_{i \in N_C} \sum_{j \in N_C; i \neq j} W_{ijk} \leq 1 \quad \forall k \in V \quad (4.9)$$

$$\sum_{j \in N_C; j \neq i} O_{ij} \leq \sum_{k \in V} X_{i0k} \quad \forall i \in N_C \quad (4.10)$$

$$\sum_{i \in N_C; i \neq j} O_{ij} \leq \sum_{k \in V} X_{0jk} \quad \forall j \in N_C \quad (4.11)$$

$$\sum_{k \in V} \sum_{j \in N_C} X_{0jk} - \sum_{i \in N_C} \sum_{j \in N_C} O_{ij} = 1 \quad (4.12)$$

$$Y_i \geq C_{iM} \quad \forall i \in N_C \quad (4.13)$$

$$Y_i \leq C_{iM} + H \left(1 - \sum_{k \in V} X_{i0k} \right) \quad \forall i \in N_C \quad (4.14)$$

$$Y_j - Y_i \leq H \left(1 - \sum_{k \in V} X_{ijk} - \sum_{k \in V} X_{jik} \right) \quad \forall i, j \in N_C; i \neq j \quad (4.15)$$

$$A_i - A_j + H \sum_{k \in V} X_{ijk} + (H - t_{ij} - t_{ji}) \sum_{k \in V} X_{jik} \leq H - t_{ij} \quad \forall i, j \in N_C; i \neq j \quad (4.16)$$

$$A_i - A_j + H \sum_{k \in V} W_{ijk} \leq H - t_{i0} - t_{0j} \quad \forall i, j \in N_C; i \neq j \quad (4.17)$$

$$A_j \geq H \sum_{k \in V} W_{ijk} - H + Y_j + t_{0j} \quad \forall i, j \in N_C; i \neq j \quad (4.18)$$

$$A_j \geq Y_j + t_{0j} - H \left(1 - X_{0jk} + \sum_{i \in N_C} W_{ijk} \right) \quad \forall j \in N_C; i \neq j; k \in V \quad (4.19)$$

$$A_j \leq Y_j + t_{0j} + H \left(1 - X_{0jk} + \sum_{i \in N_C} W_{ijk} \right) \quad \forall j \in N_C; i \neq j; k \in V \quad (4.20)$$

$$C_{im} - S_{jm} \leq H \left(1 - \sum_{k \in V} X_{ijk} - O_{ij} \right) \quad \forall i, j \in N_C; i \neq j; m \in M \quad (4.21)$$

$$S_{i(m+1)} \geq C_{im} \quad \forall i \in N_C; m \in M - 1 \quad (4.22)$$

$$C_{im} = S_{im} + p_{im} \quad \forall i \in N_C; m \in M \quad (4.23)$$

$$(C_{iM} - C_{jM})(X_{i0k}X_{0jk}) \leq H(1 - Z_{ijk}) \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.24)$$

$$(C_{jM} - C_{iM})(X_{i0k}X_{0jk}) \leq HZ_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.25)$$

$$Z_{ijk} \leq X_{i0k}X_{0jk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.26)$$

$$W_{ijk} \leq Z_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.27)$$

$$\sum_{i \in N_C} W_{ijk} \leq 1 \quad \forall j \in N_C; k \in V \quad (4.28)$$

$$T_i \geq A_i - dd_i \quad \forall i \in N_C \quad (4.29)$$

$$A_i, u_i, S_{im}, C_{im}, Y_i, T_i \geq 0 \quad (4.30)$$

$$X_{ijk} \in \{0,1\} \quad \forall i, j \in N; k \in V \quad (4.31)$$

$$W_{ijk}, Z_{ijk} \in \{0,1\} \quad \forall i, j \in N_C; k \in V \quad (4.32)$$

$$O_{ij} \in \{0,1\} \quad \forall i, j \in N_C \quad (4.33)$$

Modelde görüldüğü gibi Eşitlik (4.4) ve Eşitlik (4.24) – (4.26) kısıtları modelin doğrusal yapısını bozmaktadır. Bu nedenle geliştirilen modeli doğrusal bir yapıya dönüştürmek için Çizelge 4.1’de yardımcı değişkenler tanımlanmıştır. Buna göre modele eklenen doğrusallaştırma kısıtları aşağıda verilmiştir.

$$AX1_{ijk} \leq HX_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.34)$$

$$AX1_{ijk} \leq u_j \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.35)$$

$$AX1_{ijk} \geq u_j - H(1 - X_{ijk}) \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.36)$$

$$AX2_{ijk} \leq HX_{jik} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.37)$$

$$AX2_{ijk} \leq u_j \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.38)$$

$$AX2_{ijk} \geq u_j - H(1 - X_{jik}) \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.39)$$

$$AX3_{ijk} \leq HX_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.40)$$

$$AX3_{ijk} \leq u_i \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.41)$$

$$AX3_{ijk} \geq u_i - H(1 - X_{ijk}) \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.42)$$

$$AX4_{ijk} \leq HX_{jik} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.43)$$

$$AX4_{ijk} \leq u_i \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.44)$$

$$AX4_{ijk} \geq u_i - H(1 - X_{jik}) \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.45)$$

$$AX5_{ijk} \leq X_{i0k} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.46)$$

$$AX5_{ijk} \leq X_{0jk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.47)$$

$$AX5_{ijk} \geq X_{i0k} + X_{0jk} - 1 \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.48)$$

$$AX6_{ijk} \leq HAX1_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.49)$$

$$AX6_{ijk} \leq C_{iM} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.50)$$

$$AX6_{ijk} \geq C_{iM} - H(1 - AX1_{ijk}) \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.51)$$

$$AX7_{ijk} \leq HAX1_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.52)$$

$$AX7_{ijk} \leq C_{jM} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.53)$$

$$AX7_{ijk} \geq C_{jM} - H(1 - AX1_{ijk}) \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.54)$$

$$AX5_{ijk} \in \{0,1\} \quad \forall i, j \in N; i \neq j; k \in V \quad (4.55)$$

$$AX1_{ijk}, AX2_{ijk}, AX3_{ijk}, AX4_{ijk}, AX6_{ijk}, AX7_{ijk} \geq 0 \quad \forall i, j \in N; i \neq j; k \in V \quad (4.56)$$

Yeni değişkenlerin modele eklenmesiyle Eşitlik (4.4) ve (4.24) – (4.26) kısıtları aşağıdaki gibi güncellenmiştir.

$$(AX1_{ijk} + AX2_{ijk}) - (AX3_{ijk} + AX4_{ijk}) + Q_k X_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.57)$$

$$+ (Q_k - d_i - d_j) X_{jik} \leq (Q_k - d_i)(X_{ijk} + X_{jik})$$

$$AX2_{ijk} - AX3_{ijk} \leq H(1 - Z_{ijk}) \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.58)$$

$$AX3_{ijk} - AX2_{ijk} \leq HZ_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.59)$$

$$Z_{ijk} \leq AX5_{ijk} \quad \forall i, j \in N_C; i \neq j; k \in V \quad (4.60)$$

Matematiksel modelin amacı Eşitlik (4.1), teslim tarihlerinden sonra yapılan teslimatlardan kaynaklanan gecikmelerin toplamı ile toplam tur zamanını en küçükleme. Eşitlik (4.2) ve Eşitlik (4.3)'de verilen kısıtlar atama kısıtları olarak adlandırılmaktadır. Eşitlik (4.2) bir müşteriye mutlaka tek bir araçla bir kez hizmet verilmesini garanti etmekte, Eşitlik (4.3) ise araç bazında düğümlerin girdi çıktı dengesini sağlamaktadır. Eşitlik (4.4) – Eşitlik (4.6) kapasite ve alt tur eleme kısıtlarıdır. Eşitlik (4.4) herhangi bir düğüm üzerindeki yük miktarının ilgili araç kapasitesini geçmemesini sağlamaktadır. Eşitlik (4.5) ve Eşitlik (4.6) alt tur oluşumunu önlemek için kullanılan u_i değişkeninin sınırlarını belirlemektedir. Eşitlik (4.7) ve Eşitlik (4.8) turlarda ziyaret edilen ilk ve son müşteriler kullanılarak araç bazında tur kombinasyonunu sağlayan kısıtlardır. Eşitlik (4.9), araç bazında ardışık tur kombinasyonlarının sayısının her zaman her bir aracın toplam tur sayısından en fazla bir eksik olmasını garanti etmektedir. Eşitlik (4.10) ve Eşitlik (4.11) her araç için turlarda ziyaret edilen ilk ve son müşteriler kullanılarak bütünsel tur kombinasyonunu sağlayan kısıtlardır. Eşitlik (4.12) sistemdeki tüm araçlar için toplam ardışık tur kombinasyonları sayısının her zaman bütünsel olarak toplam tur sayısından bir eksik olmasını garanti etmektedir. Eşitlik (4.13) – Eşitlik (4.15) aracın dağıtımına çıkması için o tura ait tüm müşteri siparişlerinin üretiminin tamamlanması gerektiğini belirtmektedir. Eşitlik (4.13) ve Eşitlik (4.14) ilgili turdaki son

ziyaret edilen müşteri siparişinin üretim tamamlanma zamanını belirlerken, Eşitlik (4.15) son ziyaret edilen müşteriden önce ziyaret edilen müşteriler için de aracın turdaki son müşteri siparişinin üretiminin tamamlanması gerektiğini belirtmektedir. Eşitlik (4.16) aynı turda bulunan ardışık müşteriler için sipariş teslim zamanını belirlemektedir. Eşitlik (4.17) ve Eşitlik (4.18) başka bir turu izleyen turdaki ilk müşteri için sipariş teslim zamanını belirlemektedir. Eşitlik (4.17) ve Eşitlik (4.18) sırasıyla dağıtımda aracın ve üretimin beklenildiği durum için kullanılmaktadır. Eşitlik (4.19) ve Eşitlik (4.20) her bir aracın ilk turundaki ilk müşteri için sipariş teslim zamanını belirlemektedir. Benzer şekilde Eşitlik (4.21) aynı turda bulunan ardışık müşteriler için ve başka bir turu izleyen turdaki ilk müşteri için üretime başlama zamanlarını belirlemektedir. Eşitlik (4.22) ve Eşitlik (4.23) akış tipi üretim ortamında ardışık makineler ve işler için üretime başlama ve üretim tamamlanma zamanlarını göstermektedir. Eşitlik (4.24) ve Eşitlik (4.25) bütünsel tur kombinasyonu için aynı araçla taşınan ürünlerin hangisinin daha önce dağıtılacağını üretim tamamlanma zamanlarına göre belirlendiği kısıt kümesidir. Eşitlik (4.26) Z_{ijk} karar değişkeninin sadece aynı araçla taşınan ürünler için anlamlı olmasını sağlar. Eşitlik (4.27) ve Eşitlik (4.28) W_{ijk} karar değişkeninin Z_{ijk} değişkeni yardımıyla değer almasını sağlar. Eşitlik (4.29) müşterilerin belirlemiş oldukları teslim tarihi dikkate alınarak aracın geç teslimat yapması durumundaki gecikmeyi belirlemektedir. Eşitlik (4.30) – Eşitlik (4.33) modeldeki değişkenlere ait negatif olmama ve 0-1 tamsayı olma kısıtlarıdır.

Eşitlik (4.4) ve (4.24)–(4.26) ‘da yer alan kısıtların doğrusal yapıyı bozması nedeniyle model $AX1_{ijk}$, $AX2_{ijk}$, $AX3_{ijk}$, $AX4_{ijk}$, $AX5_{ijk}$, $AX6_{ijk}$ ve $AX7_{ijk}$ yardımcı değişkenleri ile doğrusal hale getirilmiştir. Buna göre güncellenen kısıtlar Eşitlik (4.57) ve (4.58)–(4.60) olarak düzenlenmiştir. İki 0-1 tam sayılı değişkenin çarpımına ilişkin doğrusallaştırma kısıtları Eşitlik (4.46) – Eşitlik (4.48)’de verilirken; 0-1 tam sayılı değişken ile pozitif bir değişkenin çarpımına ilişkin doğrusallaştırma kısıtları Eşitlik (4.34) – Eşitlik (4.45) ile Eşitlik (4.49) – Eşitlik (4.54)’da verilmiştir. Eşitlik (4.55) ve Eşitlik (4.56) ise kullanılan yardımcı değişkenlere ilişkin sırasıyla tamsayı olma ve negatif olmama kısıtlarıdır.

4.1.2. PAT_ÇA Problemi için açıklayıcı örnek

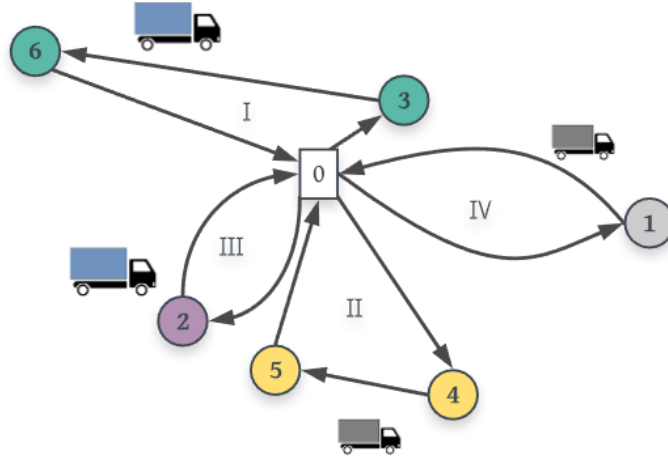
PAT_ÇA Probleminin daha iyi anlaşılabilmesi için örnek bir test verisi incelenmiştir. Problem daha önce belirtilen varsayımlar altında 6 farklı müşteriye ait

siparişlerin 2 makineli permütasyon akış tipi üretim ortamında üretildikten sonra kapasiteleri farklı iki araçla ilgili müşterilere teslimatların yapıldığı bir bütünleşik üretim ve dağıtım ortamını içermektedir. İlgili test verisine ait parametre değerleri Çizelge 4.2’de verilmiştir. Çizelgenin ilk sütunu ‘0’ depo düğümü ve diğer düğümlerde müşterileri ifade edecek şekilde sistemdeki düğümleri göstermektedir. İkinci ve üçüncü sütunda müşterilerin koordinatlarına ait bilgiler yer almaktadır. PAT_TA Probleminde taşıma süresi matrisi için bu koordinatlardan yararlanarak öklit uzaklıkları hesaplanırken PAT_ÇA Probleminde sistemde birden fazla araç olduğundan üretimde geçen sürenin dağıtımına baskın gelmemesi için PAT_TA Probleminde kullanılan uzaklık değerleri sistemdeki araç sayısının karekökü ($\sqrt{V}$) ile doğru orantılı olarak artırılmıştır. Dördüncü sütunda müşteri talepleri yer alırken beşinci ve altıncı sütun sırasıyla birinci ve ikinci makinelerdeki işlem sürelerini içermektedir. Çizelgenin son sütunu ise müşteriler tarafından belirlenmiş teslim tarihi bilgilerini içermektedir. Sistemde yer alan araçların kapasiteleri Q_1 ve Q_2 sırasıyla 211 ve 219 olarak belirlenmiştir.

Çizelge 4.2. Örnek probleme ilişkin rassal veri seti

i	X	Y	d_i	p_{i1}	p_{i2}	dd_i
0	0	0	-	-	-	-
1	42	-12	89	90	88	535
2	-17	-31	87	91	96	453
3	9	15	90	91	92	309
4	17	-48	63	60	61	436
5	-6	-44	55	52	60	457
6	-38	29	64	61	62	347

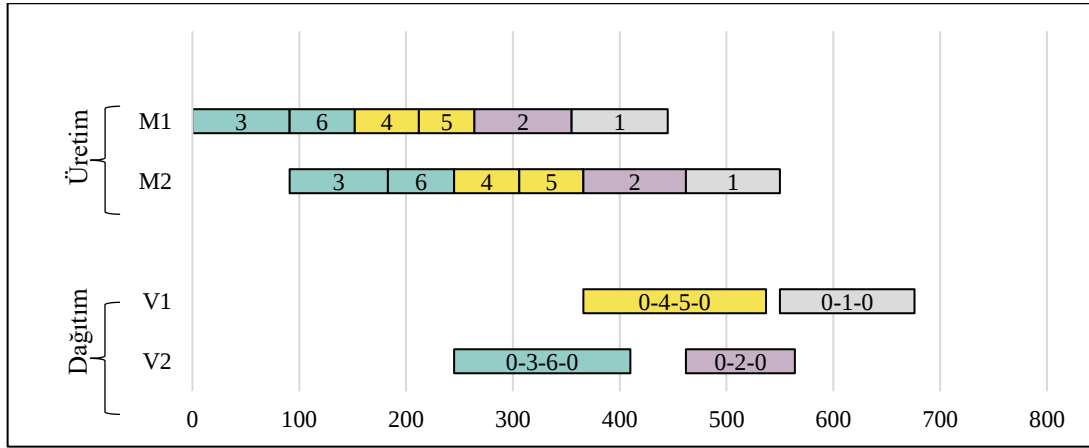
Test verisine ilişkin uygun bir çözüm Şekil 4.1’de verilmiştir. Şekil 4.1’de görüldüğü gibi sistemde iki aracın olduğu durumda birinci araç (0-4-5-0) ve (0-1-0) şeklinde, ikinci araç ise (0-3-6-0) ve (0-2-0) şeklinde ikişer turla müşterilere hizmet vermektedir.



Şekil 4.1. Örnek probleme ilişkin uygun bir çözüm

PAT_ÇA Problemi için uygun çözüme ait Gantt şeması Şekil 4.2’de verilmiştir. Şekil 4.2 incelendiğinde ilk turda yer alan müşteri siparişlerinin tamamlanma zamanı turdaki son müşteri olan 6 numaralı müşterinin son makinedeki üretim bitiş zamanına eşdeğerdir ($t = 245$). Her iki araç da bu anda depoda hazır bulunduğundan 2 numaralı araç ilk turun dağıtımına $t = 245$ anında başlamıştır. 2 numaralı araç ilk turdaki müşterilere sırasıyla $A_3 = 271$ ve $A_6 = 342$ anında ulaşmış ve depoya ($t = 410$) anında dönmüştür. İlgili müşterilerin talep ettikleri servis zamanı dikkate alındığında bu turda herhangi bir gecikme oluşmamıştır. 4 ve 5 numaralı siparişin bulunduğu ikinci partinin üretimi ($t = 366$) anında tamamlanmıştır. 2 numaralı araç bu esnada taşıma durumunda olduğundan sistemde yer alan 1 numaralı araç 4 ve 5 numaralı müşterilere sırasıyla ($A_4 = 439, A_5 = 473$) anında ulaşarak depoya ($t = 537$) anında dönmüştür. İlgili müşterinin talep ettiği servis zamanı dikkate alındığında bu turda 19 birimlik ($T_4 = 3, T_5 = 16$) bir gecikme oluşmuştur. Üçüncü partide yer alan 2 numaralı müşteri siparişinin üretimi ($t = 462$) anında tamamlanmıştır. 1 numaralı araç henüz depoya dönmediğinden üçüncü partinin dağıtımını 2 numaralı araç yapmıştır. 2 numaralı araç 2 numaralı müşteriye $A_2 = 513$ anında ulaşmış ve depoya ($t = 564$) anında dönmüştür. 2 numaralı müşteri için son teslim anı $dd_2 = 453$ olduğundan $T_2 = 513 - 453 = 60$ birimlik bir gecikme oluşmuştur. Son partide yer alan 1 numaralı müşterinin siparişi ($t = 550$) anında tamamlanmış bu anda 2 numaralı araç dağıtımda olduğundan depoda hazır olarak bekleyen 1 numaralı araç 1 numaralı müşteriye $A_1 = 613$ anında ulaşmış ve depoya ($t = 676$) anında dönmüştür. Bu turda da 78 birimlik bir gecikme oluşmuştur.

Son durumda PAT_ÇA Problemi için incelenen uygun çözümde toplam gecikme 157 birim bulunurken toplam tur zamanı ise 564 birim olarak hesaplanmıştır.



Şekil 4.2. Uygun çözüme ilişkin Gantt şeması

4.2. PAT_ÇA Problemi için Materyal ve Yöntem

Bu bölümde PAT_ÇA Problemi için önerilen MA'nın çalışma mantığı açıklandıktan sonra algoritmanın daha iyi anlaşılabilmesi amacıyla önerilen MA örnek bir test problemine uygulanmıştır. Son olarak önerilen yöntemlerin performanslarını değerlendirmek amacıyla PAT_ÇA Problemi için test problemlerinin üretilme mekanizması verilmiştir.

4.2.1. PAT_ÇA Problemi için önerilen MA

PAT_ÇA Problemi için geliştirilen algoritma mantığı temel olarak PAT_TA Problemi için geliştirilen algoritma ile aynıdır. PAT_TA Probleminde sistemde tek araç varken PAT_ÇA Probleminde sistemde heterojen kapasiteli sınırlı sayıda araç bulunmaktadır. Bu nedenle PAT_ÇA Problemi için geliştirilen algorithmada tur bölünmesi gerçekleştirildikten sonra her bir turun uygun çözüm durumunu bozmayacak şekilde sistemde yer alan araçlardan birine atanması gerekmektedir.

Çizelge 4.3. PAT_ÇA Problemi için önerilen MA'nın adımları

Adımlar	Açıklama
Adım 0:	Algoritma parametrelerini (<i>popülasyon büyüklüğü, turnuva büyüklüğü, çaprazlama oranı, mutasyon oranı, maksimum jenerasyon sayısı</i>) belirle.
Adım 1:	<i>Popülasyon büyüklüğü</i> kadar birey oluştur.
Adım 1.1:	İlk bireyi Çizelge 3.5'te verilen algoritma ile oluştur. İkinci birey için dev turu Çizelge 3.6'da verilen algoritma ile oluştur. (<i>Popülasyon büyüklüğü/2</i>)-2 birey için dev turu Çizelge 3.7'de verilen algoritma ile oluştur. <i>Popülasyon büyüklüğü/2</i> birey için dev turu tamamen rassal olarak oluştur.
Adım 1.2:	Her bir dev turu Çizelge 3.8'de verilen algoritma mantığı ile alt turlara böl. Tur bölmesini maksimum kapasiteli aracın kapasitesine göre gerçekleştir.
Adım 1.3:	Tur bölünmesi gerçekleştikten sonra her bir turu uygun çözüm durumunu bozmayacak şekilde sistemde yer alan araçlardan birine ata.
BAŞLA	Jenerasyon sayısı <i>maksimum jenerasyon sayısı</i> ' ndan küçük ve eşit olduğu sürece aşağıdaki adımları uygula.
Adım 2:	Her bir bireyin uyum değerini hesapla.
Adım 3:	<i>Turnuva büyüklüğü</i> kadar rassal olarak seçilen bireye Turnuva Seçim Yöntemini uygula.
Adım 4:	<i>Çaprazlama oranı</i> 'nı dikkate alarak Sıra Çaprazlama operatörü ile iki yeni birey oluştur.
Adım 5:	<i>Mutasyon oranı</i> 'nı dikkate alarak yeni bireylere mutasyon operatörünü uygula.
Adım 6:	Yeni oluşturulan bireyleri Çizelge 3.8'de verilen algoritma mantığı ile alt turlara böl.
Adım 7:	Tur bölünmesi gerçekleştikten sonra her bir turu uygun çözüm durumunu bozmayacak şekilde sistemde yer alan araçlardan birine ata.
Adım 8:	Yeni bireyleri değerlendir. EĞER her iki yeni oluşturulan birey de popülasyonda varsa Rassal olarak seçilen birine mutasyon operatörü uygula ve Adım 8'e dön. EĞER bireylerden biri popülasyonda varsa Popülasyonda yer almayan bireyi popülasyonla karşılaştırmak üzere seç. EĞER her iki birey de popülasyonda yer almıyorsa En iyi uyum değerine sahip olan bireyi popülasyonla karşılaştırmak üzere seç.
Adım 9:	Yerel arama prosedürünü uygula.
Adım 10:	Yeni jenerasyon için popülasyonu güncelle.
BİTİR	
Adım 11:	En iyi uyum değerine sahip bireyi (çözümü) elde et.

Böylece Çizelge 4.3'te görüldüğü gibi PAT_TA Problemi için geliştirilen algoritmaya her bir turda hangi aracın hizmet vereceğinin kararını içeren adımlar eklenerek algoritma PAT_ÇA Problemine uyarlanmıştır.

PAT_ÇA Problemi için önerilen algorithma kullanılan genetik gösterim, algorithma kullanılan seçim yöntemi, genetik operatörler ve yerel arama prosedürü PAT_TA Problemi için önerilen algorithmadaki gibi olduğundan bu bölümde sadece kromozomun genetik kodunun çözümü ayrıntılı olarak açıklanmıştır. PAT_ÇA Problemi için uygun bir çözümü temsil eden kromozom yapısının gerçek bir çözüme dönüştürülmesi iki aşamada gerçekleştirilmektedir. İlk aşamada tur bölünmesi PAT_TA Problemi için önerilen algoritma mantığına göre gerçekleştirilmektedir. Tur bölünmesi gerçekleştirildikten sonra ilgili turlarda yer alan müşterilere hangi araçların hizmet vereceği belirlenmelidir. Bu aşamada ilgilenilen turda yer alan ürünlerin üretim bitiş zamanına ve araçların depoya geri döndükleri zamana göre farklı durumlar göz önüne

alınacaktır. İlgili parti için kapasite kısıtından dolayı tek bir aday araç varsa bu parti o araca atanmaktadır. İlgili parti için birden fazla alternatif araç varsa aşağıdaki durumlar gerçekleşebilir: (i) İlgilenilen partide yer alan siparişlerin üretim bitiş zamanı, alternatif araçların depoya geri dönüş zamanlarından küçükse bu parti depoya geri dönüş zamanı en erken olan araca atanır. (ii) İlgilenilen partide yer alan siparişlerin üretim bitiş zamanı, alternatif araçların depoya geri dönüş zamanlarından büyükse bu durumda zaten üretimin bitiş bekleneneğinden ilgili parti alternatif araçlardan birine rassal olarak atanır.

4.2.2. Algoritma adımlarının örnek test verisine uygulanması

Bu bölümde önerilen algoritmanın daha iyi anlaşılabilmesi için Çizelge 4.2’de parametreleri verilen örnek üzerinde algoritmanın adımları bir jenerasyon uygulanmıştır.

Adım 0: Yapılan test koşulları sonucunda önerilen MA’ da kullanılan algoritma parametreleri Çizelge 4.4’teki gibi belirlenmiştir.

Çizelge 4.4. Önerilen algoritmada kullanılan parametreler

Algoritma parametresi	Değer
Popülasyon büyüklüğü	= 30
Turnuva büyüklüğü	= 3
Çaprazlama oranı	= 0,90
Mutasyon oranı	= 0,10
Maksimum jenerasyon sayısı	= $N \times 50$

Adım 1: Örnek test verisi dikkate alınarak 6 bireyden oluşan bir başlangıç popülasyonu için turlar Şekil 4.3’te gösterildiği gibi oluşturulmuştur.

Birey 1:	4	5	2	1	3	6
Birey 2:	3	1	4	5	2	6
Birey 3:	3	1	5	4	2	6
Birey 4:	4	6	5	2	1	3
Birey 5:	1	2	4	5	6	3
Birey 6:	6	1	3	4	2	5

Şekil 4.3. Başlangıç popülasyonu (Turların oluşturulması)

Birey 1, Clarke ve Wright (1964) tarafından önerilen tasarruf algoritmasına göre oluşturulmuş ve (0-4-5-2-0), (0-1-3-0) ve (0-6-0) şeklinde üç tur elde edilmiştir. Birey 2 için en yakın komşu sezgiseli ile (0-3-1-4-5-2-6-0) şeklinde dev tur oluşturulmuştur. Birey 3 için rassal görev sezgiseli ile (0-3-1-5-4-2-6-0) şeklinde dev tur oluşturulmuştur. Birey 4,5 ve 6 için dev turlar tamamen rassal olarak sırasıyla (0-4-6-5-2-1-3-0), (0-1-2-4-

5-6-3-0) ve (0-6-1-3-4-2-5-0) şeklinde oluşturulmuştur. Görüldüğü gibi *Birey 1* dışındaki tüm bireyler önce rotala-sonra grupta yaklaşımıyla üretilmiştir.

Adım 1.2: Gruplama aşamasında ise Prins (2004) tarafından önerilen algoritma mantığına göre önerilen ve PAT_TA Problemi için geliştirilen MA' da bahsedilen yaklaşım bu dev turlara uygulanmıştır. Şekil 4.3'te görüldüğü gibi her bir tur farklı bir renkle ifade edilecek şekilde oluşturulmuştur.

Adım 1.3: Turlar oluşturulduktan sonra hangi turda hangi aracın hizmet vereceği *Birey 2* üzerinden anlatılmıştır. Diğer bireyler içinde benzer yaklaşım kullanılmıştır. İlk aşamada uygun çözüm durumunu korumak için her bir turda hizmet verebilecek araçlar kümesi oluşturulmuştur. Görüldüğü gibi ilk turda yer alan müşterilerin talepleri toplamı $d_3 + d_1 = 179$ birim olduğundan bu turda her iki araç da hizmet verebilmektedir. Her bir tur için uygun araçlar kümesi oluşturulduktan sonra ilgili turun üretim bitiş zamanı ve alternatif araçların depoya geri dönüş zamanları dikkate alınacaktır. Bunun için her bir tura ilişkin üretim tamamlanma zamanları ve tur zamanları belirlenmiştir.

Çizelge 4.5. *Birey 2* için araçların belirlenmesi

Turlar	Üretim Tamamlanma Zamanı	Tur Zamanı	Alternatif Araçlar
3-1	271	150	1,2
4-5	392	171	1,2
2	488	102	1,2
6	550	136	1,2

Buna göre ilk partide yer alan müşterilere hem 1 hem de 2 numaralı araç hizmet verebilir. Bu durumda rassal olarak seçilen 1 numaralı araç $t = 271$ anında dağıtım çıkarmakta ve görevini $t = 421$ anında tamamlayarak depoya geri dönmektedir. İkinci partide yer alan müşterilere hem 1 hem de 2 numaralı araç hizmet verebilmektedir. Bu partideki ürünlerin üretimi $t = 392$ anında tamamlanacağından ve bu anda 1 numaralı araç hala meşgul olduğundan atıl bekleme durumu oluşmaması için bu partide depoda hazır bulunan 2 numaralı araca atanmıştır. Bu durumda 2 numaralı araç $392 + 171 = 563$ anında depoya geri dönecektir. Üçüncü partide yer alan 2 numaralı siparişin üretimi $t = 488$ anında tamamlanmıştır. Bu anda 2 numaralı araç meşgul olduğundan 1 numaralı araç depoda hazır olduğundan dağıtım bu anda çıkarak $t = 590$ anında görevini tamamlayarak depoya geri dönmektedir. Son partide yer alan 6 numaralı müşterinin siparişi $t = 550$ anında tamamlanmıştır. Bu anda 2 numaralı araç depoda hazır olarak beklerken 1 numaralı araç 3. partinin dağıtımını henüz tamamlamamıştır. Bu nedenle 6 numaralı müşteriye yine 2 numaralı araç hizmet verecektir. Son durumda 3,1 ve 2

numaralı müşterilere 1 numaralı araç hizmet verirken 4,5 ve 6 numaralı müşterilere 2 numaralı araç hizmet vermiştir. Bu yaklaşımla üretim ve dağıtım aşamaları arasındaki atıl beklemler azaltılmaya ve kaynaklar etkin bir şekilde kullanılmaya çalışılmıştır. Diğer kromozomlar için de aynı yaklaşım kullanılarak turların araçlara atanması aşaması siyah renk 1 numaralı aracı kırmızı renk ise 2 numaralı aracı ifade edecek şekilde Şekil 4.4'te görüldüğü gibi tamamlanmıştır.

Birey 1:	4	5	2	1	3	6
Birey 2:	3	1	4	5	2	6
Birey 3:	3	1	5	4	2	6
Birey 4:	4	6	5	2	1	3
Birey 5:	1	2	4	5	6	3
Birey 6:	6	1	3	4	2	5

Şekil 4.4. Başlangıç popülasyonu (Araçların atanması)

Adım 2: Uyum değeri amaç fonksiyonu değerini ifade edecek şekilde her bir bireyin uyum değeri Çizelge 4.6'daki gibi hesaplanmıştır.

Çizelge 4.6. Popülasyonda yer alan bireylerin uyum değerleri

Bireyler	Toplam Tur Zamanı	Toplam Gecikme	Uyum Değeri
Birey 1:	470	563	1033
Birey 2:	559	441	1000
Birey 3:	559	424	983
Birey 4:	601	257	858
Birey 5:	519	645	1164
Birey 6:	690	330	1020

Adım 3: Önerilen algoritmada turnuva seçim mekanizması kullanılmıştır. Turnuva büyüklüğü üç olarak belirlendiğinden popülasyondan rassal olarak seçilen üç bireyin (*Birey 2*, *Birey 4*, *Birey 3*) uyum değerleri dikkate alındığında *Birey 4* diğer iki bireye galip gelerek seçilmiştir.

Adım 4: Geliştirilen algoritmada araç rotalama ve makine çizelgeleme problemlerinde sıklıkla kullanılan sıra çaprazlama operatörü kullanılmıştır. Seçilen iki ebeveyne (*Birey 4* ve *Birey 2*) sıra çaprazlama operatörü Şekil 4.5'teki gibi uygulanmıştır.

		↓		↓		
Ebeveyn 1:	4	6	5	2	1	3
Ebeveyn 2:	3	1	4	5	2	6
Çocuk 1:	3	6	5	2	1	4
Çocuk 2:	6	1	4	5	2	3

Şekil 4.5. Çaprazlama operatörünün uygulanması

Adım 5: Bu jenerasyonda mutasyon oranına bağlı olarak mutasyon operatörü uygulanmamıştır.

Adım 6 ve Adım 7: Genetik operatörlerden sonra oluşturulan yeni bireylerin genetik kodunun çözümünde yukarıda bahsedilen iki aşamalı yaklaşım kullanılarak Şekil 4.6' da verilen bireyler elde edilmiştir.

Çocuk 1:	3	6	5	2	1	4
Çocuk 2:	6	1	4	5	2	3

Şekil 4.6. Yeni oluşturulan bireyler

Adım 8: Yeni oluşturulan bireyler uygun çözüm durumuna getirildikten sonra uyum değerleri Çizelge 4.7'deki gibi hesaplanmıştır.

Çizelge 4.7. Yeni oluşturulan bireylerin uyum değerlerinin hesaplanması

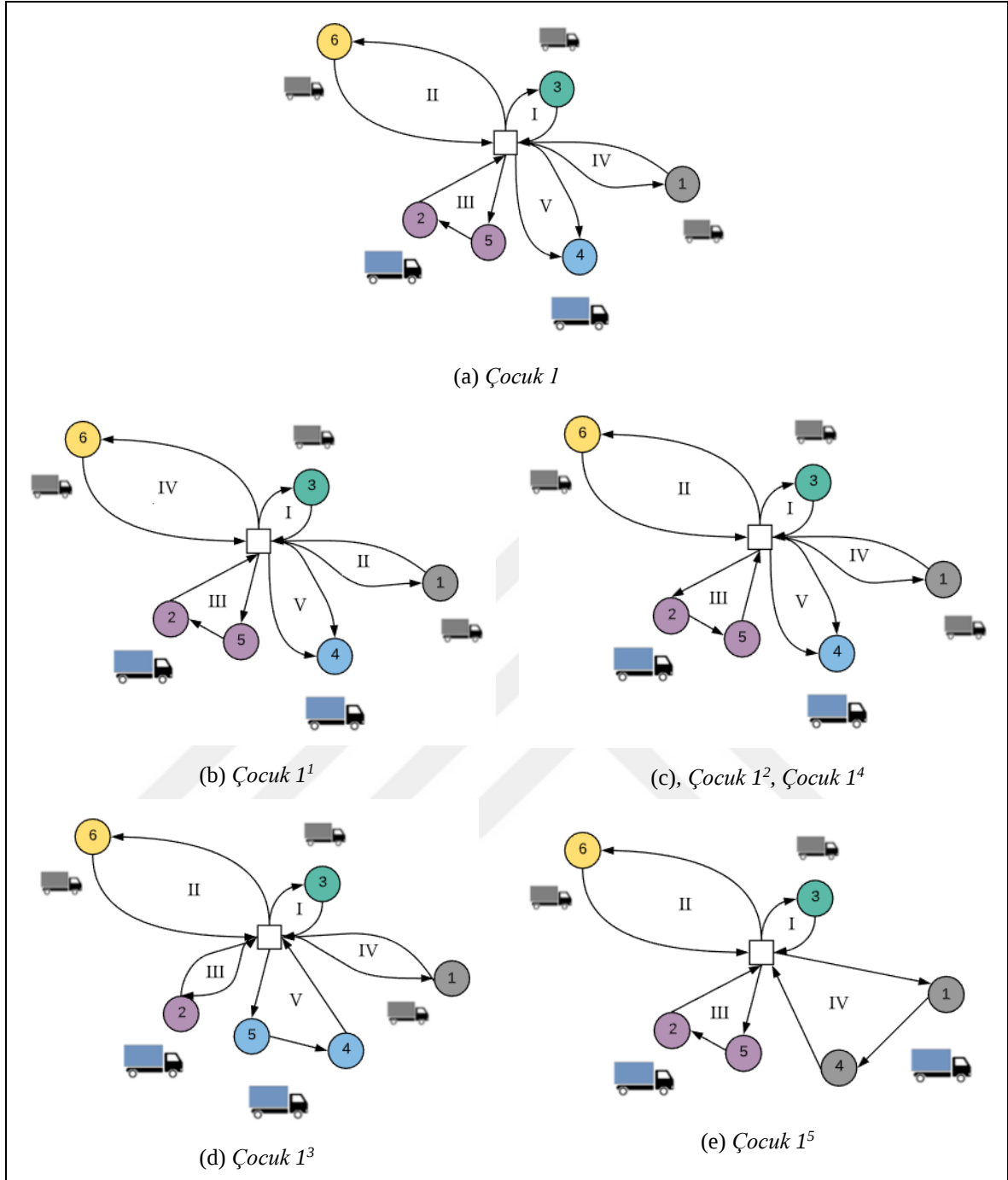
Bireyler	Toplam Tur Zamanı	Toplam Gecikme	Uyum Değeri
Çocuk 1:	601	250	851
Çocuk 2:	587	329	916

Adım 9: Önerilen yerel arama sürecinde bahsedilen operatörler Çocuk 1 üzerinde Şekil 4.7' deki gibi uygulanmıştır.

Çizelge 4.8'de uyum değerleri verilen komşu çözümler değerlendirildiğinde en kaliteli çözüm olan Çocuk 1² popülasyonla karşılaştırmak üzere seçilmiştir.

Çizelge 4.8. Komşu çözümlerin uyum değerlerinin hesaplanması

Bireyler	Toplam Tur Zamanı	Toplam Gecikme	Uyum Değeri
Çocuk 1 ¹	601	513	1114
Çocuk 1 ²	601	225	826 ←
Çocuk 1 ³	587	369	956
Çocuk 1 ⁴	601	225	826
Çocuk 1 ⁵	528	364	892



Şekil 4.7. Yerel arama prosedürü

Adım 10: Yeni oluşturulan bireyler arasından en iyi uyum değerine sahip birey (*Çocuk 1²*) popülasyonda en kötü uyum değerine sahip olan bireyle (*Birey 5*) yer değiştirir. Bu durumda bir jenerasyon sonunda güncellenen yeni popülasyon Şekil 4.8’de verilmiştir. Algoritma bir jenerasyon uygulandıktan sonra popülasyon uyum değeri ortalaması 1009,67’den 953,33’e düşmüştür.

Birey 1:	3	6	2	5	1	4
Birey 2:	4	6	5	2	1	3
Birey 3:	3	1	5	4	2	6
Birey 4:	3	1	4	5	2	6
Birey 5:	6	1	3	4	2	5
Birey 6:	4	5	2	1	3	6

Şekil 4.8. Güncel popülasyon

4.2.3. PAT_ÇA Problemi için test problemlerinin üretilmesi

Tez kapsamında PAT_ÇA Probleminin çözümü için geliştirilen yöntemlerin değerlendirilmesi amacıyla yeni test problemleri üretilmiştir. Çizelge 4.9’da görüldüğü gibi üretilen test problemleri müşteri, makine ve araç sayılarına göre küçük, orta ve büyük boyutlu problemler olmak üzere üç grupta incelenmiştir. Bunun dışında dağıtım aşamasında müşterilerin belirledikleri son teslim tarihleri sıkı, orta ve gevşek olmak üzere üç farklı seviyede incelenmiştir. Üretim aşamasında ise ürünlerin makinelerdeki işlem sürelerindeki değişkenliğe göre iki farklı seviye ele alınmıştır. Her bir kombinasyon için beş koşum yapılmıştır. Böylece 180 küçük boyutlu, 1080 orta boyutlu ve 720 büyük boyutlu olmak üzere toplamda 1980 örnek incelenmiştir.

Çizelge 4.9. PAT_ÇA Problemi için geliştirilen test problemlerinin yapısı

Örnek	N	M	V	dd	π
Küçük boyutlu	5,6,7	2,3	2	Sıkı, orta, gevşek	0,1, 0,3
Orta boyutlu	10,11,12,13,14,15	3,4,5	2,3	Sıkı, orta, gevşek	0,1, 0,3
Büyük boyutlu	20,30,40,50	4,5,6	4,5	Sıkı, orta, gevşek	0,1, 0,3

PAT_ÇA Problemi için kullanılan test problemlerinin üretim mekanizması PAT_TA Problemi için geliştirilen süreçle benzerdir. Burada dikkat edilmesi gereken nokta PAT_ÇA Probleminde araç sayısı birden fazla olduğu için üretim ve dağıtım sürelerinin ödünleşmesi adına PAT_TA Probleminde kullanılan uzaklık matrisindeki değerler araç sayısına bağlı olarak belirli bir oranda $(\sqrt{v})$ artırılmıştır. Yine PAT_TA Problemi için kullanılan müşterilerin belirttikleri teslim tarihleri PAT_ÇA Problemi için gevşek bulunduğundan teslim tarihlerinin üretilme sürecinde kullanılan ve sistemde geçen zamanı ifade eden π değeri $\pi = \rho \left[(M + N - 1) + \left(\frac{N+1}{v} \right) \right]$ şeklinde güncellenmiştir.

4.3. PAT_ÇA Problemi için Karşılaştırmalı Sonuçlar

Bu bölüm önerilen KTP modelinin performansı ve önerilen MA ve CPLEX ile elde edilen sonuçların karşılaştırılması olmak üzere iki alt başlıkta incelenmiştir.

4.3.1. Önerilen KTP modelinin sonuçları

Bu bölümde önerilen matematiksel modelin farklı müşteri, makine ve araç sayılarına göre genel performansı değerlendirilmiştir. Çizelge 4.10'da CPLEX ile elde edilen sonuçlar verilmiştir. Çizelgenin ilk üç sütunu sırasıyla örnekteki müşteri, makine ve araç sayısını ifade etmektedir. Dördüncü ve beşinci sütunda sırasıyla CPLEX tarafından verilen süre sınırı içerisinde elde edilen üst sınır ve alt sınır ortalaması verilmiştir. Altıncı sütunda yüzde sapma değeri ortalaması verilirken bu değerlerin sıfır olması optimal çözümün elde edildiği anlamına gelmektedir. Yedinci sütunda her müşteri makine kombinasyonunda yer alan tüm örnekler içinde bulunan optimal çözüme ulaşılan problem sayısı verilirken son sütunda ise saniye cinsinden ortalama çözüm süresi verilmiştir.

Çizelge 4.10'da görüldüğü gibi CPLEX müşteri sayısının 5 ve 6 olduğu örneklerin hepsinde optimal çözümlere ulaşabilmiştir. Müşteri sayısının 5 olduğu örneklerde ortalama çözüm süresi yaklaşık 15 sn iken müşteri sayısının 5'ten 6'ya çıktığı durumda çözüm süresi büyük oranda artmıştır. Müşteri sayısının 7 olduğu durumda ise 60 örneğin sadece 25 tanesinde CPLEX verilen süre sınırı içerisinde optimal çözüm elde edebilmiştir. Orta boyutlu problemlerde CPLEX tüm örneklerde verilen süre sınırı içerisinde bir üst sınır elde edebilmesine rağmen şaşırtıcı bir şekilde hiçbir örnekte optimal çözümü bulamamıştır. Bununla birlikte müşteri sayısındaki artışın problemin karmaşıklığı üzerindeki etkisi YSD değerlerinden bariz bir şekilde görülmektedir. Küçük boyutlu örneklerde ortalama YSD değeri 7,15 iken bu değer orta boyutlu örneklerde 73,46'ya ulaşmıştır. Büyük boyutlu problemlerde ise incelenen 720 örneğin sadece 388 tanesinde CPLEX ile bir üst sınır elde edilebilirken geriye kalan 332 örnekte herhangi bir üst sınır elde edilememiştir.

Çizelge 4.10. CPLEX ile elde edilen sonuçlar

	<i>N</i>	<i>M</i>	<i>V</i>	Z^{US}	Z^{AS}	YSD	OÇUPS	OÇS
<i>Küçük</i>	5	2	2	700,50	700,50	0,00	30/30	16,53
		3	2	781,03	781,03	0,00	30/30	13,69
	6	2	2	759,47	759,47	0,00	30/30	703,74
		3	2	1077,10	1077,10	0,00	30/30	618,38
	7	2	2	1009,13	677,28	26,44	11/30	8037,71
		3	2	1135,47	903,75	16,43	14/30	6709,29
Ort.				910,45	816,52	7,15	145/180	2683,22
<i>Orta</i>	10	3	2	1997,70	644,43	64,30	0/30	10800,00
		3	3	2227,20	799,89	60,48	0/30	10800,00
		4	2	1874,83	625,96	62,55	0/30	10800,00
		4	3	2218,93	711,30	64,96	0/30	10800,00
		5	2	2118,83	763,15	57,98	0/30	10800,00
		5	3	2627,17	782,32	68,15	0/30	10800,00
	11	3	2	2049,47	605,78	65,60	0/30	10800,00
		3	3	2477,20	727,61	67,12	0/30	10800,00
		4	2	2362,00	698,46	65,62	0/30	10800,00
		4	3	2962,27	770,95	70,59	0/30	10800,00
		5	2	2416,17	742,95	65,33	0/30	10800,00
		5	3	2699,27	826,58	67,06	0/30	10800,00
	12	3	2	2612,67	587,10	73,86	0/30	10800,00
		3	3	3050,97	691,46	74,92	0/30	10800,00
		4	2	2787,90	874,46	63,07	0/30	10800,00
		4	3	3285,53	805,15	73,24	0/30	10800,00
		5	2	2835,17	681,20	72,86	0/30	10800,00
		5	3	3459,70	923,42	70,56	0/30	10800,00
	13	3	2	3199,47	590,21	79,05	0/30	10800,00
		3	3	3889,53	711,55	80,35	0/30	10800,00
		4	2	3438,67	610,83	78,65	0/30	10800,00
		4	3	3691,17	719,12	78,18	0/30	10800,00
		5	2	3077,23	670,33	73,44	0/30	10800,00
		5	3	3660,70	790,96	76,46	0/30	10800,00
	14	3	2	3723,80	613,55	81,01	0/30	10800,00
		3	3	3872,20	694,76	79,77	0/30	10800,00
		4	2	3771,73	643,72	80,77	0/30	10800,00
		4	3	4224,53	717,24	80,86	0/30	10800,00
		5	2	3638,03	668,85	77,70	0/30	10800,00
		5	3	4125,27	762,56	78,88	0/30	10800,00
	15	3	2	4014,93	557,79	82,54	0/30	10800,00
		3	3	4384,73	631,13	83,85	0/30	10800,00
		4	2	3965,70	638,27	80,88	0/30	10800,00
		4	3	4998,13	757,57	83,18	0/30	10800,00
		5	2	4119,70	731,83	78,85	0/30	10800,00
		5	3	5116,83	841,73	81,90	0/30	10800,00
Ort.				3249,31	711,50	73,46	0/1080	10800,00

Çizelge 4.10. CPLEX ile elde edilen sonuçlar (devam)

	<i>N</i>	<i>M</i>	<i>V</i>	Z^{US}	Z^{AS}	YSD	OÇUPS	OÇS
<i>Büyük</i>	20	4	4	9856,90	887,68	90,19	0/30	10800,00
		4	5	10457,43	991,53	90,08	0/30	10800,00
		5	4	9545,47	915,66	89,39	0/30	10800,00
		5	5	10564,17	1031,72	89,78	0/30	10800,00
		6	4	10555,87	989,84	89,87	0/30	10800,00
		6	5	11969,77	1112,62	89,77	0/30	10800,00
	30	4	4	25405,25	1055,80	95,65	0/16	10800,00
		4	5	26801,17	1192,36	95,21	0/12	10800,00
		5	4	26410,70	1030,51	96,05	0/10	10800,00
		5	5	26913,94	1165,55	95,55	0/17	10800,00
		6	4	25102,78	1135,73	94,73	0/9	10800,00
		6	5	30766,15	1288,46	95,72	0/13	10800,00
	40	4	4	53280,93	1110,53	97,85	0/14	10800,00
		4	5	47002,00	1262,04	97,22	0/12	10800,00
		5	4	55122,29	1190,25	97,83	0/7	10800,00
		5	5	54076,38	1347,15	97,40	0/16	10800,00
		6	4	50051,40	1215,87	97,37	0/5	10800,00
		6	5	54291,25	1379,25	97,40	0/12	10800,00
	50	4	4	73516,85	1236,80	98,18	0/13	10800,00
		4	5	75608,50	1406,56	98,01	0/10	10800,00
		5	4	85823,71	1212,75	98,54	0/7	10800,00
		5	5	79423,71	1382,63	98,15	0/17	10800,00
		6	4	78460,44	1278,90	98,32	0/9	10800,00
		6	5	82571,78	1452,43	98,24	0/9	10800,00
	Ort.			42232,45	1178,02	95,27	0/388	10800,00

Büyük boyutlu problemlerde YSD değeri 95,27 olarak bulunmuştur. Bu durumda CPLEX'in gerçek hayat problemleri için kısa sürede tutarlı bir çözüm üretemeyeceği açıkça görülmektedir. Çizelge 4.10'dan elde edilen bir diğer çıkarım müşteri sayısının problem karmaşıklığı üzerindeki etkisinin doğrudan gözlemlenmesine rağmen aynı durumun makine ve araç sayısı için geçerli olmadığıdır.

Çizelge 4.10'dan CPLEX'in müşteri sayısının 30 veya daha fazla olduğu örneklerde üst sınır bulmakta zorlandığı sonucuna varılabilir ki bu gerçek hayat problemleri için orta büyüklükte bir problem olarak yorumlanabilir.

4.3.2. Önerilen MA'nın performansı

Bu bölümde üretim sürecindeki değişkenliğin ve teslim tarihlerinin problem üzerindeki etkisi incelenmiştir.

4.3.2.1. Üretim sürecindeki değişkenliğin etkisi

Bu bölümde üretim sürecindeki değişkenliğin CPLEX ve MA üzerindeki etkisi araştırılmıştır. Çizelge 4.11'un ilk sütunu üretim sürelerindeki değişkenlik katsayısını (μ) ifade etmektedir. İkinci sütunda müşteri sayıları yer almaktadır. Önerilen MA her bir

örneğe ilişkin üç koşul çalıştırılmıştır. Çizelgenin üçüncü, dördüncü ve beşinci sütunu sırasıyla bu koşullarda elde edilen en iyi çözüme, en kötü çözüme ve çözümlerin ortalamasına ilişkin MA'nın CPLEX'e göre GSD değerlerini ifade etmektedir. Bu sapmaların hesaplanması Bölüm 3.3'te verilmiştir. Çizelgenin altıncı ve dokuzuncu sütununda sırasıyla önerilen MA ve CPLEX ile elde edilen optimal çözüm sayıları verilirken yedinci ve onuncu sütunda sırasıyla önerilen MA ve CPLEX'in saniye cinsinden ortalama çözüm süreleri verilmiştir. Sekizinci sütun ise CPLEX'in MA'ya göre GSD değerlerini ifade etmektedir.

Çizelge 4.11. Farklı μ seviyelerine göre MA ve CPLEX' in performansı

μ	N	MA					CPLEX		
		GSD ^{enk}	GSD ^{enb}	GSD ^{ort.}	OCUPS	OCS	GSD	OCUPS	OCS
0,1	5	0,00	0,00	0,00	30/30	1,27	0,00	30/30	20,72
	6	0,00	0,00	0,00	30/30	1,17	0,00	30/30	650,82
	7	0,02	0,02	0,02	11/11	1,18	0,01	11/30	7860,29
	Ort.	0,00	0,00	0,00	71/71	1,21	0,00	71/90	2843,94
	10	0,00	0,00	0,00	-	1,69	4,33	0/90	10800,00
	11	0,00	0,00	0,00	-	2,03	5,85	0/90	10800,00
	12	0,00	0,00	0,00	-	2,35	8,38	0/90	10800,00
	13	0,00	0,02	0,00	-	2,56	11,52	0/90	10800,00
	14	0,00	0,00	0,00	-	2,91	15,40	0/90	10800,00
	15	0,00	0,00	0,00	-	3,35	20,52	0/90	10800,00
	Ort.	0,00	0,00	0,00	-	2,48	11,00	0/540	10800,00
	20	0,00	0,00	0,00	-	5,36	39,18	0/90	10800,00
	30	0,00	0,00	0,00	-	12,15	83,54	0/38	10800,00
	40	0,00	0,00	0,00	-	23,83	113,20	0/39	10800,00
	50	0,00	0,00	0,00	-	40,17	151,88	0/31	10800,00
	Ort.	0,00	0,00	0,00	-	20,38	96,95	0/198	10800,00
0,3	5	0,00	0,00	0,00	30/30	1,39	0,00	30/30	9,50
	6	0,00	0,00	0,00	30/30	1,14	0,00	30/30	671,30
	7	0,00	0,00	0,00	14/14	1,24	0,08	14/30	6886,71
	Ort.	0,00	0,00	0,00	74/74	1,26	0,02	74/90	2522,50
	10	0,00	0,03	0,01	-	1,67	4,39	0/90	10800,00
	11	0,00	0,05	0,01	-	2,02	6,52	0/90	10800,00
	12	0,00	0,00	0,00	-	2,34	10,21	0/90	10800,00
	13	0,00	0,00	0,00	-	2,56	13,26	0/90	10800,00
	14	0,00	0,00	0,00	-	2,90	17,41	0/90	10800,00
	15	0,00	0,00	0,00	-	3,31	23,03	0/90	10800,00
	Ort.	0,00	0,00	0,00	-	2,47	12,47	0/540	10800,00
	20	0,00	0,00	0,00	-	5,33	40,53	0/90	10800,00
	30	0,00	0,00	0,00	-	12,16	85,05	0/39	10800,00
	40	0,00	0,00	0,00	-	24,01	114,00	0/27	10800,00
	50	0,00	0,00	0,00	-	40,85	136,85	0/34	10800,00
	Ort.	0,00	0,00	0,00	-	20,59	94,11	0/190	10800,00

Önerilen MA ile CPLEX tarafından optimalite elde edilen 145 küçük boyutlu örneğin hepsi için bu değerlere ortalama 1,2 sn içerisinde ulaşılmıştır. Orta ve büyük boyutlu örneklerin neredeyse tamamında MA ile elde edilen çözümlerin CPLEX ile elde edilen

çözümlerden daha kaliteli olduğu GSD değerlerinden görülmektedir. Özellikle büyük boyutlu problemlerde CPLEX'in performansı üstel bir şekilde azalmıştır.

Üretim sürecindeki değişkenliğin etkisi incelendiğinde ise μ değerinin 0,1 ve 0,3 olduğu durumlarda küçük boyutlu problemlerde CPLEX'in elde ettiği optimal çözüm sayısı sırasıyla 71 ve 74 olarak, ortalama çözüm süreleri ise 2843,94 ve 2522,50 sn olarak bulunmuştur. Böylece üretim sürecindeki değişkenliğin problem karmaşıklığı üzerinde bariz bir etkisinin olmadığı söylenebilir.

4.3.2.2. Teslim tarihlerinin etkisi

Amaç fonksiyonunda yer alan toplam gecikme zamanı müşterilerin teslim tarihlerine göre belirlendiğinden bu bölümde farklı teslim tarihi seviyelerinin problem üzerindeki etkisi tartışılmıştır. Bu amaçla her bir örnek için gevşek, orta ve sıkı olmak üzere üç farklı teslim tarihi belirlenmiştir.

Çizelge 4.12. Farklı teslim tarihi seviyelerine göre MA ve CPLEX' in performansı

dd_i	N	MA					CPLEX		
		GSD ^{enk}	GSD ^{enb}	GSD ^{ort.}	OCUPS	OCS	GSD	OCUPS	OCS
Gevşek	5	0,00	0,00	0,00	20/20	1,21	0,00	20/20	5,79
	6	0,00	0,00	0,00	20/20	1,13	0,00	20/20	211,89
	7	0,00	0,01	0,00	17/20	1,21	0,00	17/20	2938,51
	Ort.	0,00	0,00	0,00	57/60	1,18	0,00	57/60	1052,06
	10	0,00	0,00	0,00	-	1,62	4,29	0/60	10800,00
	11	0,00	0,01	0,00	-	1,89	7,01	0/60	10800,00
	12	0,00	0,00	0,00	-	2,24	10,67	0/60	10800,00
	13	0,00	0,04	0,00	-	2,46	16,97	0/60	10800,00
	14	0,00	0,00	0,00	-	2,70	21,35	0/60	10800,00
	15	0,00	0,00	0,00	-	3,17	30,79	0/60	10800,00
	Ort.	0,00	0,00	0,00	-	2,35	15,18	0/360	10800,00
	20	0,00	0,00	0,00	-	5,21	56,83	0/60	10800,00
	30	0,00	0,00	0,00	-	11,88	130,84	0/27	10800,00
	40	0,00	0,00	0,00	-	23,46	168,22	0/23	10800,00
	50	0,00	0,00	0,00	-	39,77	222,01	0/28	10800,00
	Ort.	0,00	0,00	0,00	-	20,08	144,48	0/138	10800,00
Orta	5	0,00	0,00	0,00	20/20	1,33	0,00	20/20	11,57
	6	0,00	0,00	0,00	20/20	1,14	0,00	20/20	637,44
	7	0,02	0,02	0,02	5/20	1,19	0,00	5/20	8947,90
	Ort.	0,00	0,00	0,00	45/60	1,22	0,00	45/60	3198,97
	10	0,00	0,00	0,00	-	1,69	4,98	0/60	10800,00
	11	0,00	0,00	0,00	-	2,09	6,51	0/60	10800,00
	12	0,00	0,00	0,00	-	2,38	9,43	0/60	10800,00
	13	0,00	0,00	0,00	-	2,58	11,65	0/60	10800,00
	14	0,00	0,00	0,00	-	3,00	14,77	0/60	10800,00
	15	0,00	0,00	0,00	-	3,41	19,93	0/60	10800,00
	Ort.	0,00	0,00	0,00	-	2,53	11,21	0/360	10800,00
	20	0,00	0,00	0,00	-	5,39	34,99	0/60	10800,00
	30	0,00	0,00	0,00	-	12,29	67,50	0/21	10800,00
	40	0,00	0,00	0,00	-	24,08	89,85	0/23	10800,00
	50	0,00	0,00	0,00	-	40,84	97,00	0/16	10800,00
	Ort.	0,00	0,00	0,00	-	20,65	72,34	0/120	10800,00

Çizelge 4.12. Farklı teslim tarihi seviyelerine göre MA ve CPLEX' in performansı (devam)

dd_i	N	MA					CPLEX		
		GSD ^{enk}	GSD ^{enb}	GSD ^{ort}	OÇUPS	OÇS	GSD	OÇUPS	OÇS
Sıkı	5	0,00	0,00	0,00	20/20	1,44	0,00	20/20	27,97
	6	0,00	0,00	0,00	20/20	1,20	0,00	20/20	1133,85
	7	0,00	0,00	0,00	3/20	1,24	0,14	3/20	10234,10
	Ort.	0,00	0,00	0,00	43/60	1,29	0,05	43/60	3798,64
	10	0,00	0,00	0,00	-	1,74	3,81	0/60	10800,00
	11	0,00	0,00	0,00	-	2,09	5,03	0/60	10800,00
	12	0,00	0,00	0,00	-	2,42	7,79	0/60	10800,00
	13	0,00	0,00	0,00	-	2,65	8,55	0/60	10800,00
	14	0,00	0,00	0,00	-	3,01	13,09	0/60	10800,00
	15	0,00	0,00	0,00	-	3,42	14,60	0/60	10800,00
	Ort.	0,00	0,00	0,00	-	2,56	8,81	0/360	10800,00
	20	0,00	0,00	0,00	-	5,44	27,74	0/60	10800,00
	30	0,00	0,00	0,00	-	12,30	53,15	0/29	10800,00
	40	0,00	0,00	0,00	-	24,22	77,86	0/20	10800,00
	50	0,00	0,00	0,00	-	40,93	75,87	0/21	10800,00
	Ort.	0,00	0,00	0,00	-	20,72	58,66	0/130	10800,00

İlk sütun hariç Çizelge 4.12’de yer alan tüm sütunlar Çizelge 4.11’deki sütunlarla aynıdır. İlk sütunda bu kez teslim tarihi seviyeleri yer almıştır.

Çizelge 4.12’ de üretim sürecindeki değişkenliğin aksine teslim tarihlerinin problem karmaşıklığı üzerindeki etkisi bariz bir şekilde gözlenmiştir. CPLEX ile küçük boyutlu problemler için gevşek orta ve sıkı teslim tarihi seviyelerinde verilen süre sınırı içerisinde optimal çözüme ulaşılan problem sayıları sırasıyla 57, 45 ve 43 olarak elde edilmiştir. Bunun yanı sıra küçük boyutlu problemlerde teslim tarihlerinin gevşek olduğu durumdan sıkı olduğu duruma geçerken ortalama çözüm süreleri sırasıyla 1052, 3198, ve 3798 olarak bulunmuştur. Buradan çıkarılan sonuç problemin teslim tarihlerinin sıkı olduğu durumun gevşek olduğu duruma göre çözümünün daha zor olduğudur. Çizelge 4.12’den orta boyutlu problemlerde CPLEX’in MA’ya göre GSD değerinin teslim tarihlerinin gevşek olduğu durumdan sıkı olduğu duruma geçerken 15,18’den 11,21’e 11,21’den de 8,81’e düştüğü görülmektedir. Benzer şekilde büyük boyutlu problemlerde de CPLEX için GSD değerleri gevşek, orta ve sıkı teslim tarihi seviyeleri için sırasıyla 144,48, 72,34 ve 58,66 olarak bulunmuştur. Aynı durum PAT_TA Probleminde de gözlenmiştir. Bunun sebebi sıkı teslim tarihinin gevşek teslim tarihine göre MA’yı da daha fazla zorlaması böylece CPLEX ile MA arasındaki göreceli farkın kapanması durumu olabilir.

5. ATÖLYE TİPİ ÜRETİM ORTAMI İLE ÇOK ARAÇLI DAĞITIM ORTAMINI İÇEREN BÜDC (AT_ÇA) PROBLEMİ

Tez kapsamında incelenen üçüncü problemde siparişler atölye tipi üretim ortamına göre üretilmektedir. Bu tür üretim ortamına sahip işletmeler müşteri siparişlerini karşılamak için genellikle TZÜ sistemine başvurmaktadır. Müşteriler siparişlerini kendi ihtiyaçlarını karşılayacak şekilde özelleştirebildiğinden nihai ürünler hacimsel olarak küçük ve değişken miktarlarda üretilmektedir. Ürünler özelleştirilebilir olduğundan müşteri siparişinin alındığı ana kadar üretim bekleyebilmekte bu da üretim süresini büyük ölçüde artırmaktadır. Böylece tez kapsamında incelenen üçüncü problem, bu prensibe göre faaliyet gösteren işletmeler için bütünleşik yaklaşımın gerekliliğinin savunulduğu en temel uygulama alanlarından birisi olarak görülmektedir.

Günümüzde tükenmekte olan doğal kaynakların varlığı sürdürülebilirlik için yeni çözüm arayışları bulma zorunluluğunu da beraberinde getirmiştir. Sürdürülebilirliği sağlamak için sosyal gelişim ve ekonomik büyümenin yanı sıra çevresel faktörlerin de dikkate alınması gerekmektedir. Bu nedenle işletmeler ekonomik kalkınmayı hedeflerken doğaya mümkün olduğunca az müdahale ederek çevresel faktörleri de dikkate almak zorundadır. Tez kapsamında incelenen AT_ÇA Probleminin birinci amaç fonksiyonu ile çevrenin korunması adına dağıtım sürecinde hizmet veren araçlar tarafından havaya salınan toplam CO₂ emisyonunun en küçüklenmesi hedeflenmiştir. İkinci amaç fonksiyonu ise müşteri hizmet düzeyini artırmak amacıyla en büyük gecikmenin en küçüklenmesi olarak ele alınmıştır.

Böylece AT_ÇA Problemi hem üretim ve dağıtım ortamı hem de amaç fonksiyonu açısından PAT_TA ve PAT_ÇA Problemlerinden ayrılmaktadır.

Bu bölüm AT_ÇA Probleminin tanımı, AT_ÇA Problemi için önerilen yöntemler ve karşılaştırmalı sonuçlar olmak üzere üç bölümde incelenmiştir.

5.1. AT_ÇA Probleminin Tanımı

Bu bölümde AT_ÇA Probleminin tanımı verilmiştir. Sistemde farklı coğrafi konumlarda bulunan her bir i müşterisi ($i \in N_C$) kendi siparişi için d_i birimlik bir talebe sahiptir. Sistemde yer alan tüm düğümler kümesi olarak ifade edilen küme N müşteriler kümesi ile depo düğümünün birleşiminden oluşmaktadır. Herhangi bir i siparişinin nihai ürün haline gelmesi için i ürünü sistemde yer alan makineleri tam olarak bir kez ziyaret ederek birden fazla operasyondan geçmektedir. Siparişlerin makine ziyaret sıraları

birbirinden farklı olup ürünler atölye tipi üretim ortamına göre üretilmektedir. Talebi d_i birim olan i siparişinin m makinesindeki ($m \in M$) işlem zamanı önceden bilinmekte ve p_{im} kadardır. Siparişler partiler halinde dağıtılmakta ve herhangi bir partide üretim sırasına bağlı kalınmaksızın birden fazla müşteri siparişi taşınabilmektedir. Dağıtım sürecinde hizmet veren her bir k aracı sınırlı bir kapasiteye Q_k sahiptir ve planlama periyodu boyunca birden fazla kez kullanılabilir. Bunun dışında her bir i müşterisi, siparişi için bir son teslim tarihi dd_i belirlemiş olup bu tarihten sonraki yapılan teslimatlarda gecikme oluşmaktadır. k aracı tarafından havaya salınan CO₂ emisyonu ilgili aracın kapasitesi ve araç üzerindeki yük miktarına göre değişmektedir. Havaya salınan toplam CO₂ emisyonu hesaplanırken kullanılan yakıt tüketim oranı formülasyonu için Xiao ve ark. (2012) ile S. Zhang ve ark. (2014) 'nın yapmış oldukları çalışmalardan esinlenilmiştir. Buna göre herhangi bir k aracının i düğümünden j düğümüne giderken ($i, j \in N$) tüketmiş olduğu yakıt miktarı $\rho_k^0 + (\rho_k^* - \rho_k^0)u_{ijk}/Q_k$ eşitliği ile hesaplanmaktadır. ρ_k^0 ve ρ_k^* sırasıyla k . aracın boşken ve tam kapasite doluyken yakıt tüketim oranlarını, u_{ijk} ise k aracın i düğümünden j düğümüne ($i, j \in N$) giderken üzerinde taşıdığı toplam yük miktarını ifade etmektedir. S. Zhang ve ark. (2014)'nın çalışmasında havaya salınan toplam CO₂ emisyonu, litre cinsinden yakıt tüketim miktarı ve 1 litre yakıt tüketilirken havaya salınan kg cinsinden CO₂ miktarı çarpımıyla elde edilmiştir. Hesaplama kolaylığı açısından bu değer $1 \text{ kg CO}_2/\text{lt}$ olarak alınmıştır.

Böylece incelenen problem sonucunda (i) müşteri siparişlerinin makinelerdeki operasyon sıralarının belirlenmesi, (ii) her bir müşteri siparişinin üretime başlama ve üretim tamamlanma zamanının belirlenmesi, (iii) belirli bir turda hangi aracın hizmet vereceğinin belirlenmesi (iv) her bir aracın dağıtım yapacağı müşterilerin ziyaret sıralarının belirlenmesi, (v) her bir müşteri siparişinin teslimat zamanının belirlenmesine ilişkin kararlar alınmaktadır.

AT_ÇA Probleminin daha iyi anlaşılabilmesi amacıyla öncelikle ele alınan problem için çok amaçlı KTDP modeli geliştirilmiş ardından AT_ÇA Problemine ilişkin örnek bir çözüm incelenmiştir.

5.1.1. AT_ÇA Problemi için önerilen çok amaçlı KTDP modeli

Bu bölümde çalışma kapsamında AT_ÇA Problemi için önerilen çok amaçlı KTDP modelinde kullanılan simgeler, model ve modelde yer alan kısıtların açıklamaları

detaylarıyla verilmiştir. Önerilen KTDP modelinde kullanılan simgeler Çizelge 5.1’de verilmiştir.

Çizelge 5.1. Önerilen Çok Amaçlı KTDP modeli modelinde kullanılan simgeler

Simge	Açıklama
<i>İndisler:</i>	
i, j	: İş (sipariş, müşteri)
m, l	: Makine
k	: Araç
<i>Kümeler:</i>	
N_C	: İşler (siparişler, müşteriler) kümesi
N	: Tüm düğümler kümesi ($N_C \cup 0$)
M	: Makineler kümesi
V	: Araçlar kümesi
DA	: Ayrık arklar kümesi
CA	: Birleşik arklar kümesi
LM	: İşlerin en son ziyaret ettiği makineler kümesi
<i>Parametreler:</i>	
d_i	: i . müşteri siparişine ait talep miktarı
p_{im}	: i . müşteri siparişinin m . makinedeki işlem süresi
t_{ij}	: i . ve j . düğümler arası ulaşım süresi
dd_i	: i . müşteri siparişine ait teslim tarihi
Q_k	: k . araç kapasitesi
H	: Yeterince büyük bir sayı
ρ_k^0	: k . aracın yakıt tüketim oranı (araç boşken)
ρ_k^*	: k . aracın yakıt tüketim oranı (araç tam kapasite dolu iken)
<i>Değişkenler:</i>	
O_{ijm}	: 1, m . makinede i . iş j . işten önce işleniyorsa; 0, aksi durumda
X_{ijk}	: 1, k . araç i . düğümden hemen sonra j . düğüme gidiyorsa; 0, aksi durumda
W_{ijk}	: 1, k . araç önceki turu i . düğümden tamamlayıp sonraki tura j . düğüme başlıyorsa; 0 aksi durumda
u_{ijk}	: i . müşteriyi ziyaret etmeden hemen önce araç üzerindeki toplam yük miktarı
S_{im}	: i . müşteri siparişinin m . makinedeki üretime başlama zamanı
C_{im}	: i . müşteri siparişinin m . makinedeki üretim bitiş zamanı
B_i	: i . müşteri siparişinin sırası üzerindeki son makinedeki üretim bitiş zamanı
A_i	: aracın i . müşteriye varış zamanı
Y_i	: i . müşteri siparişinin ait olduğu partinin üretim bitiş zamanı
T_i	: i . müşteri siparişi için gecikme miktarı
T_{max}	: en büyük gecikme miktarı

$$\min f_1 = \sum_{i \in N} \sum_{j \in N} \sum_{k \in V} \left(\rho_k^0 X_{ijk} + \frac{\rho_k^* - \rho_k^0}{Q_k} u_{ijk} \right) t_{ij} \quad (5.1)$$

$$\min f_2 = T_{max} \quad (5.2)$$

$$S_{im} - S_{jm} + H O_{ijm} \geq p_{jm} \quad \forall (i, m) \rightarrow (j, m) \in DA \quad (5.3)$$

$$S_{jm} - S_{im} + H(1 - O_{ijm}) \geq p_{im} \quad \forall (i, m) \rightarrow (j, m) \in DA \quad (5.4)$$

$$O_{ijm} + O_{jim} = 1 \quad \forall (i, m) \rightarrow (j, m) \in DA \quad (5.5)$$

$$S_{jm} - S_{jl} \geq p_{jl} \quad \forall (j, l) \rightarrow (j, m) \in CA \quad (5.6)$$

$$C_{jm} - S_{jm} = p_{jm} \quad \forall j \in N_C; m \in M \quad (5.7)$$

$$B_j = C_{jm} \quad \forall j \in N_C; m \in LM \quad (5.8)$$

$$\sum_{k \in V} \sum_{i \in N_C} X_{ijk} = 1 \quad \forall j \in N_C \quad (5.9)$$

$$\sum_{j \in N} X_{ijk} = \sum_{j \in N} X_{jik} \quad \forall i \in N; k \in V \quad (5.10)$$

$$u_{iok} = 0 \quad \forall i \in N_C; k \in V \quad (5.11)$$

$$\sum_{j \in N} u_{jik} - \sum_{j \in N} u_{ijk} = \sum_{j \in N} X_{ijk} d_i \quad \forall i \in N_C; k \in V \quad (5.12)$$

$$u_{ijk} \leq (Q_k - d_i) X_{ijk} \quad \forall i, j \in N; k \in V \quad (5.13)$$

$$\sum_{j \in N_C} W_{ijk} \leq X_{iok} \quad \forall i \in N_C; i \neq j; k \in V \quad (5.14)$$

$$\sum_{i \in N_C} W_{ijk} \leq X_{ojk} \quad \forall j \in N_C; i \neq j; k \in V \quad (5.15)$$

$$\sum_{j \in N_C} X_{ojk} - \sum_{i \in N_C} \sum_{j \in N_C; i \neq j} W_{ijk} \leq 1 \quad \forall k \in V \quad (5.16)$$

$$Y_i \geq B_i \quad \forall i \in N_C \quad (5.17)$$

$$Y_j - Y_i \leq H \left(1 - \sum_{k \in V} X_{ijk} - \sum_{k \in V} X_{jik} \right) \quad \forall i, j \in N_C; i \neq j \quad (5.18)$$

$$A_i - A_j + H \sum_{k \in V} X_{ijk} + (H - t_{ij} - t_{ji}) \sum_{k \in V} X_{jik} \leq H - t_{ij} \quad \forall i, j \in N_C; i \neq j \quad (5.19)$$

$$A_i - A_j + H \sum_{k \in V} W_{ijk} \leq H - t_{i0} - t_{0j} \quad \forall i, j \in N_C; i \neq j \quad (5.20)$$

$$A_j \geq H \sum_{k \in V} W_{ijk} - H + Y_j + t_{0j} \quad \forall i, j \in N_C; i \neq j \quad (5.21)$$

$$A_j \geq Y_j + t_{0j} - H \left(1 - X_{ojk} + \sum_{i \in N_C} W_{ijk} \right) \quad \forall j \in N_C; i \neq j; k \in V \quad (5.22)$$

$$A_j \leq Y_j + t_{0j} + H \left(1 - X_{ojk} + \sum_{i \in N_C} W_{ijk} \right) \quad \forall j \in N_C; i \neq j; k \in V \quad (5.23)$$

$$T_i \geq A_i - dd_i \quad \forall i \in N_C \quad (5.24)$$

$$T_{max} \geq T_i \quad \forall i \in N_C \quad (5.25)$$

$$u_{ijk} \geq 0 \quad \forall i, j \in N; k \in V \quad (5.26)$$

$$A_i, S_{im}, C_{im}, Y_i, T_i \geq 0 \quad \forall i, j \in N_C; m \in M \quad (5.27)$$

$$X_{ijk} \in \{0, 1\} \quad \forall i, j \in N; k \in V \quad (5.28)$$

$$W_{ijk} \in \{0, 1\} \quad \forall i, j \in N_C; k \in V \quad (5.29)$$

$$O_{ijm} \in \{0, 1\} \quad \forall i, j \in N_C; m \in M \quad (5.30)$$

Eşitlik (5.1) de ifade edilen birinci amaç fonksiyonu yakıt tüketimine bağlı havaya salınan CO₂ emisyonunun en küçüklenmesidir. Bu eşitlikte CO₂ emisyonu yakıt tüketimine bağlı bir fonksiyondur. Matematiksel modelin ikinci amacı ise en büyük gecikme zamanının en küçüklenmesi olarak Eşitlik (5.2)'de verilmiştir.

Atölye tipi üretim ortamına ait kısıtlar Eşitlik (5.3) – (5.8) arasında verilmiştir. Eşitlik (5.3) – (5.5) i ve j işlerinin m makinesi üzerindeki sıralamasını kullanarak üretime başlama zamanlarını belirler. Eşitlik (5.6) bir işin sırası üzerindeki ardışık makinelerde üretim başlama zamanlarının belirlenmesini sağlar. Eşitlik (5.7) herhangi bir işin herhangi bir makinedeki üretim tamamlanma zamanının o işin ilgili makinedeki üretime başlama zamanı ile makinedeki işlem süresinin toplamı kadar olmasını garanti eder. Eşitlik (5.8) bir işin üretim tamamlanma zamanının o işin son makinedeki işleminin tamamlanma zamanına eşit olmasını sağlar.

Dağıtım ortamına ait kısıtlar Eşitlik (5.9) – (5.17) arasında verilmiştir. Eşitlik (5.9) ve Eşitlik (5.10)'da verilen kısıtlar atama kısıtları olarak adlandırılmaktadır. Eşitlik (5.9) bir müşteriye mutlaka tek bir araçla bir kez hizmet verilmesini garanti etmekte, Eşitlik (5.10) araç bazında düğümlerin girdi çıktı dengesini sağlamaktadır. Eşitlik (5.11) – Eşitlik (5.13) kapasite ve alt tur eleme kısıtlarıdır. Eşitlik (5.11) depoya dönüş yapılan herhangi bir düğüm üzerindeki yük miktarının sıfır olmasını sağlamaktadır. Eşitlik (5.12) aynı turda yer alan ardışık düğümler ile araç üzerindeki yükün belirlenmesinde kullanılmaktadır. Eşitlik (5.13) ve Eşitlik (5.14) alt tur oluşumunu önlemek için kullanılan u_{ijk} değişkeninin sınırlarını belirlemektedir. Eşitlik (5.15) ve Eşitlik (5.16) turlarda ziyaret edilen ilk ve son müşteriler kullanılarak araç bazında tur kombinasyonunu sağlayan kısıtlardır. Eşitlik (5.17), araç bazında ardışık tur kombinasyonlarının sayısının her zaman her bir aracın toplam tur sayısından en fazla bir eksik olmasını garanti etmektedir. Eşitlik (5.18) – Eşitlik (5.20) aracın dağıtıma çıkması için o tura ait tüm müşteri siparişlerinin üretiminin tamamlanması gerektiğini belirtmektedir. Eşitlik (5.18) ve Eşitlik (5.19) ilgili turdaki son ziyaret edilen müşteri siparişinin üretim tamamlanma zamanını belirlerken, Eşitlik (5.20) son ziyaret edilen müşteriden önce ziyaret edilen müşteriler için de aracın turdaki son müşteri siparişinin üretiminin tamamlanması gerektiğini belirtmektedir. Eşitlik (5.21) aynı turda bulunan ardışık müşteriler için sipariş teslim zamanını belirlemektedir. Eşitlik (5.22) ve Eşitlik (5.23) başka bir turu izleyen turdaki ilk müşteri için sipariş teslim zamanını belirlemektedir. Eşitlik (5.22) ve Eşitlik (5.23) sırasıyla dağıtımda aracın ve üretimin beklenildiği durum için kullanılmaktadır. Eşitlik (5.24) ve Eşitlik (5.25) her bir aracın ilk turundaki ilk müşteri için sipariş teslim

zamanını belirlemektedir. Eşitlik (5.26) müşterilerin belirlemiş oldukları teslim tarihi dikkate alınarak aracın geç teslimat yapması durumundaki gecikmeyi belirlemektedir. Eşitlik (5.27) – Eşitlik (5.30) modeldeki değişkenlere ait negatif olmama ve 0-1 tamsayı olma kısıtlarıdır.

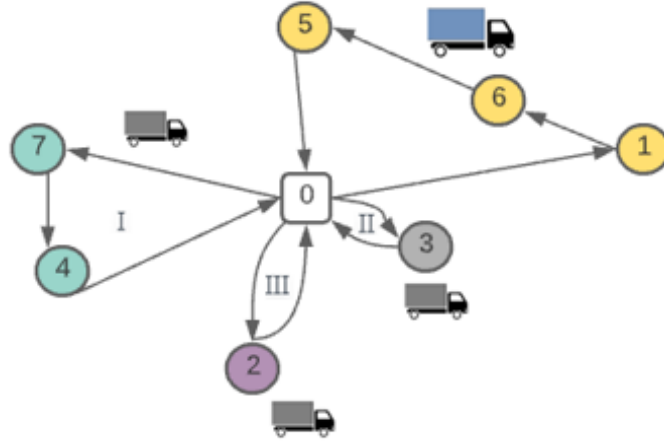
5.1.2. AT_ÇA Problemi için açıklayıcı örnek

AT_ÇA Probleminin daha iyi anlaşılabilmesi için örnek bir test verisi incelenmiştir. Problem daha önce belirtilen varsayımlar altında 7 farklı müşteriye ait siparişlerin 3 makineli atölye tipi üretim ortamında üretildikten sonra sistemde yer alan üç farklı tip araçla ilgili müşterilere teslimatların yapıldığı bir bütünleşik üretim ve dağıtım ortamını içermektedir. Çizelge 5.2’de ilk sütun depo düğüm ve müşterileri ifade ederken ikinci ve üçüncü sütun depo düğüm ve müşterilere ait X ve Y koordinatlarını vermektedir. Dördüncü sütunda ilgili müşterilere ait talep miktarları yer alırken, beşinci, altıncı ve yedinci sütun sırasıyla ürünlerin birinci, ikinci ve üçüncü makinedeki işlem zamanlarını vermektedir. Sekizinci sütunda işlerin makine rotaları verilirken son sütunda müşteriler tarafından belirlenen teslim tarihleri yer almaktadır. Uzaklıklar hesaplanırken öklit uzaklık baz alınmıştır.

Çizelge 5.2. Örnek probleme ilişkin rassal veri seti

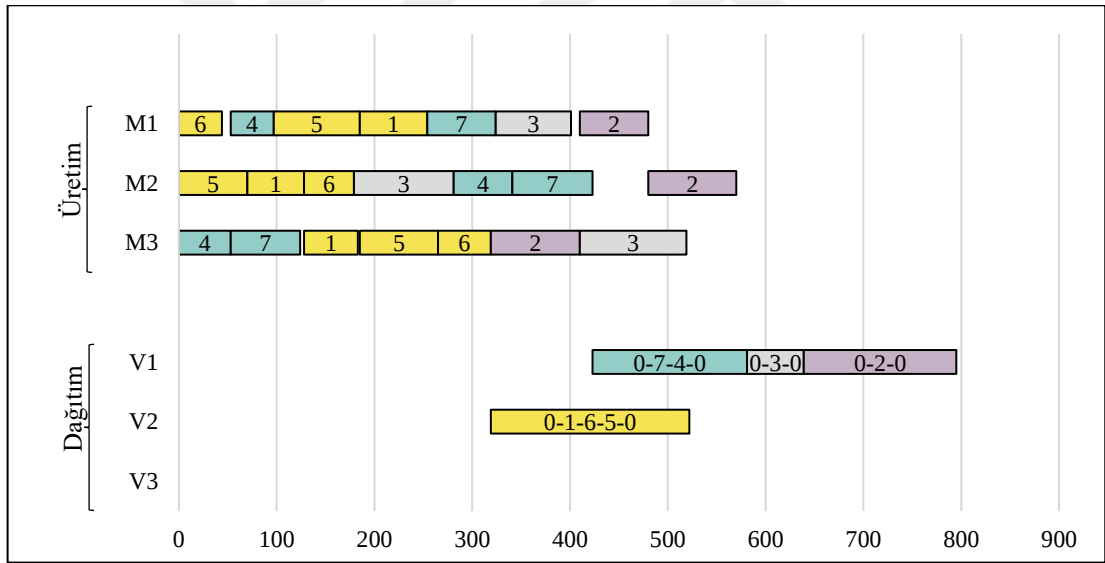
i	X	Y	d_i	p_{i1}	p_{i2}	p_{i3}	Makine sırası	dd_i
0	0	0	-	-	-	-	-	-
1	52	11	60	69	58	55	2,3,1	319
2	-14	-76	80	70	90	91	3,1,2	471
3	15	-24	95	77	102	109	2,1,3	395
4	-45	-37	52	44	60	53	3,1,2	271
5	-3	68	85	88	70	80	2,1,3	319
6	27	35	50	44	51	54	1,2,3	275
7	-49	10	84	70	82	71	3,1,2	395

Sistemde yer alan üç tip aracın (küçük, orta ve büyük) kapasiteleri sırasıyla Q_1, Q_2 ve Q_3 sırasıyla 140, 210 ve 280 birim olarak belirlenmiştir. Araçların yakıt tüketimi hesaplanırken kullanılan yakıt tüketim oranları araçlar boşken ρ_1^0, ρ_2^0 ve ρ_3^0 sırasıyla 0,2, 0,3 ve 0,4 birim iken araçlar tam kapasite doluyken ρ_1^*, ρ_2^* ve ρ_3^* sırasıyla 0,26, 0,39 ve 0,52 birimdir.



Şekil 5.1. Örnek probleme ilişkin uygun bir çözüm

Test verisine ilişkin uygun bir çözüm Şekil 5.1’de verilmiştir. Şekil 5.1’de görüldüğü gibi sistemde üç aracın olduğu durumda küçük kapasiteli araç (V_1) birden fazla turda hizmet verirken büyük kapasiteli araç (V_3) bu örnekte hiç kullanılmamıştır.



Şekil 5.2 Probleme ilişkin örnek bir çözüm

AT_ÇA Problemi için uygun çözüme ait Gantt şeması ise Şekil 5.2’de verilmiştir. Şekil 5.2 incelendiğinde ilk partide yer alan müşteri siparişlerinin tamamlanma zamanı 6 numaralı müşterinin rotası üzerindeki son makinedeki üretim bitiş zamanına eşdeğerdir ($t = 319$). Bu partinin dağıtımını yapacak olan araç (V_2) turdaki müşterilere sırasıyla $A_1 = 373$, $A_6 = 408$ ve $A_5 = 453$ anlarında ulaşmış ve depoya $t = 522$ anında dönmüştür. İlgili müşterilerin son teslim tarihleri dikkate alındığında bu turda oluşan gecikmeler sırasıyla 54,133 ve 134 birimdir.

7 ve 4 numaralı müşterilerin yer aldığı partinin üretim tamamlanma zamanı 7 numaralı müşteriye ait siparişin üretim tamamlanma zamanına ($t = 423$) eşdeğerdir. Bu partinin dağıtımını yapacak olan araç (V_1) bu anda depoda hazır olduğundan dağıtıma bu anda başlamış ve turdaki müşterilere sırasıyla $A_7 = 474$ ve $A_4 = 522$ anlarında ulaşarak $t = 581$ anında depoya geri dönmüştür. Bu turda oluşan gecikmeler sırasıyla 79 ve 251 birimdir.

3 numaralı müşterinin siparişi $t = 519$ anında tamamlanmış ancak araç bu anda hala önceki partinin dağıtımını yaptığından bu turun dağıtımına aracın depoya döndüğü anda $t = 581$ başlanmıştır. Sistemde yer alan en küçük araç ikinci turunda 3 numaralı müşteriye $A_3 = 610$ anında ulaşmış ve $t = 639$ anında depoya geri dönmüştür. 2 numaralı müşterinin üretimi $t = 570$ anında tamamlanmış olmasına rağmen bu müşteriye hizmet veren araç önceki partinin dağıtımını tamamlamadığından $t = 639$ anına kadar sipariş beklemiştir. Araç son turunda 2 numaralı müşteriye $A_2 = 717$ anında ulaşarak görevini tamamlamıştır. 2 numaralı müşteri için geçilme miktarı 246 birim iken 3 numaralı müşterinin sipariş teslimatında 215 birimlik bir gecikme yaşanmıştır. Sonuç olarak en büyük gecikme miktarı bu çözüm için 4 numaralı müşteride yaşanmış olup 251 birimdir.

Örnek çözüme ilişkin toplam CO₂ emisyonunun hesaplanması ise Çizelge 5.3'te ayrıntılı olarak açıklanmıştır.

Çizelge 5.3. Örnek probleme ilişkin toplam CO₂ emisyonu

i, j	t_{ij}	u_{ijk}	ρ^0, ρ^*, Q_k	CO ₂
0,7	51	84+52=136	0,2, 0,26, 140	13,17
7,4	48	52	0,2, 0,26, 140	10,67
4,0	59	0	0,2, 0,26, 140	11,80
0,3	29	95	0,2, 0,26, 140	6,98
3,0	29	0	0,2, 0,26, 140	5,80
0,2	78	80	0,2, 0,26, 140	18,27
2,0	78	0	0,2, 0,26, 140	15,60
0,1	54	60+50+85=195	0,3, 0,39, 210	20,71
1,6	35	50+85=135	0,3, 0,39, 210	12,53
6,5	45	85	0,3, 0,39, 210	15,14
5,0	69	0	0,3, 0,39, 210	20,70
				151,37

Çizelgenin ilk sütununda aracın rotasında yer alan i, j düğümleri, ikinci sütununda ise bu düğümler arasındaki mesafe verilmiştir. Üçüncü sütun aracın i düğümünden j düğümüne giderken üzerindeki yük miktarını ifade etmektedir. Dördüncü sütunda ilgili turda görev alan aracın sırasıyla boşken ve tam

kapasite dolu iken yakıt tüketim oranları ile kapasitesi verilmiştir. Son sütunda ise ilgili tur için tahsis edilen aracın i düğümünden j düğümüne giderken tükettiği yakıt miktarına bağlı olarak toplam CO_2 emisyonu hesaplanmıştır.

Buna göre ilgili örnek için birinci amaç fonksiyonu 251 birim, ikinci amaç fonksiyonu ise 151,37 birim olarak elde edilmiştir.

5.2. AT_ÇA Problemi için Materyal ve Yöntem

Bu bölümde AT_ÇA Problemi için önerilen AEKY, PYA ve BSGA-II yaklaşımları sırasıyla açıklanmış ardından önerilen yöntemlerin daha iyi anlaşılabilmesi için her bir yöntemin adımları örnek test verisine uygulanmıştır. Son olarak yöntemlerin performanslarının karşılaştırılması amacıyla AT_ÇA Problemi için test problemlerinin üretilme mekanizması açıklanmıştır.

5.2.1. Artırılmış epsilon kısıt yöntemi

Çok amaçlı optimizasyon problemleri için geliştirilen ve yaygın olarak kullanılan çözüm yöntemlerinden birisi Epsilon Kısıt (ϵ -constraint) Yöntemidir. Bu yönteme göre amaç fonksiyonlarından birisi optimize edilmek üzere seçilir ve diğer amaç fonksiyonları kısıtlara dönüştürülür. Bu durumda bir en küçükleme problemi, e_i i . amaç fonksiyonu için belirlenen üst sınır değeri, p amaç fonksiyonu sayısı ve S uygun çözüm kümesi olmak üzere aşağıdaki gibi ifade edilebilir:

$$\begin{aligned} \min f_1(x) \\ f_i(x) \leq e_i \quad i = 2, \dots, p \\ x \in S \end{aligned}$$

Epsilon Kısıt Yöntemini doğru bir şekilde uygulayabilmek için amaç fonksiyonlarının etkin küme içerisindeki değer aralıklarının hesaplanması gerekmektedir. Amaç fonksiyonlarına ait en iyi değer bireysel optimizasyonla (tek bir amacın optimizasyonu) kolayca elde edilebilirken en kötü değer (nadir point) belirlenmesi kolay değildir. Bir diğer dikkat edilmesi gereken durum, çok amaçlı bir problemde etkin bir çözümün garanti edilebilmesi için diğer amaç fonksiyonlarının kısıtlarının bağlayıcı olması gerekmektedir. Aksi takdirde elde edilen çözüm etkin değil zayıf etkin çözüm olarak adlandırılır. Bu belirsizliğin üstesinden gelmek için amaç fonksiyonu kısıtları yapay değişkenlerle eşitlik haline dönüştürülmüştür. Sadece etkin çözümlerin üretilmesi

adına bu değişkenlerin toplamı amaç fonksiyonuna ikincil terim olarak oldukça düşük bir katsayı δ ile yansıtılmıştır. δ değeri genellikle $10^{-3} \leq \delta \leq 10^{-6}$ aralığında seçilmektedir. Böylece düzenlenmiş formülasyon aşağıda verilmiştir.

$$\begin{aligned} \min & \left(f_1(x) - \delta(s_2 + s_3 + \dots + s_p) \right) \\ & f_i(x) + s_i = e_i \quad i = 2, \dots, p \\ & x \in S, s_i \in R^+ \end{aligned}$$

Klasik Epsilon Kısıt Yönteminde bu değişkenlerin toplamı doğrudan amaç fonksiyonuna yansıtılırken bazı ölçekleme problemleri oluşabilmektedir. Bu problemin üstesinden gelebilmek amacıyla amaç fonksiyonundaki ikinci terim olan yapay s_i değişkeni, r_i i . amaç fonksiyonunun ödünleşme matrisindeki değer aralığını ifade etmek suretiyle, s_i/r_i ile değiştirilmiştir. Böylece klasik Epsilon Kısıt Yöntemindeki bu değişimle Artırılmış Epsilon Kısıt Yöntemi (AEKY) aşağıdaki gibi ifade edilmiştir (Mavrotas, 2009).

$$\begin{aligned} \min & \left(f_1(x) - \delta(s_2/r_2 + s_3/r_3 + \dots + s_p/r_p) \right) \\ & f_i(x) + s_i = e_i \quad i = 2, \dots, p \\ & x \in S, s_i \in R^+ \end{aligned}$$

AEKY yöntemi uygulanırken öncelikle ödünleşim matrisi oluşturulur. Bireysel optimizasyonla elde edilen çözümler zayıf etkin çözüm durumunda olabileceğinden Mavrotas (2009) ödünleşim matrisi oluşturulurken sadece Pareto optimal çözümlerin üretilmesi için Sözlük Sıralaması Yöntemini (Lexicographic Optimization) önermiştir. Bu yöntemle göre öncelikle ele alınan tek bir amaç en iyilenir ardından bu değer kısıt olarak modele yansıtılarak diğer amacın en iyilenmesi gerçekleştirilir. Ödünleşim matrisi oluşturulduktan sonra kısıt olarak kullanılacak olan, kalan $(p - 1)$ amaç fonksiyonunun her biri için bir aralık elde edilir. Ardından i . amaç fonksiyonuna ilişkin elde edilen aralık q_i eşit aralığa bölünerek ızgara noktaları (grid points) oluşturulur. Böylece i . amaç fonksiyonunun sağ taraf sabitini parametrik olarak değiştirmek için toplamda $(q_i + 1)$ aralık elde edilir. Izgara noktaların sayısı arttıkça daha yoğun bir etkin küme oluşturulabilir ancak bu durum çözüm süresini de artıracaktır. Çözüm süresini azaltmak amacıyla önerilen algorithmada sağ taraf sabiti en gevşekten en sıkıya doğru modele

girilerek çözüm gerçekleştirilmektedir. Böylece verilen sağ taraf sabitiyle uygun olmayan bir çözüm elde edilmesi durumunda döngünün kalan koşumlarını çözdürmenin hiçbir anlamı olmayacağından algoritma durdurulur.

5.2.1. AEKY'nin örnek test verisine uygulanması

Çok amaçlı optimizasyon çözüm yöntemi olarak AEKY önceki bölümde verilen örnek probleme aşağıdaki gibi uygulanmıştır: Öncelikle problemde yer alan amaçlara ilişkin ödünleşim matrisi oluşturulmuştur. Ödünleşim matrisi oluşturulurken sözlük sıralaması yöntemi kullanılmıştır. Buna göre amaçlardan sadece biri en iyilenmiş ardından en iyilenen amacın almış olduğu değer kısıt olarak modele girilerek diğer amaç en iyilenmeye çalışılmıştır. Böylece incelenen problem için oluşturulan ödünleşim matrisi Çizelge 5.4'te verilmiştir.

Çizelge 5.4. Örnek probleme ilişkin ödünleşim matrisi

Amaç	$\sum CO_2$	T_{max}
$\min \sum CO_2$	136,50	398
$\min T_{max}$	187,34	168

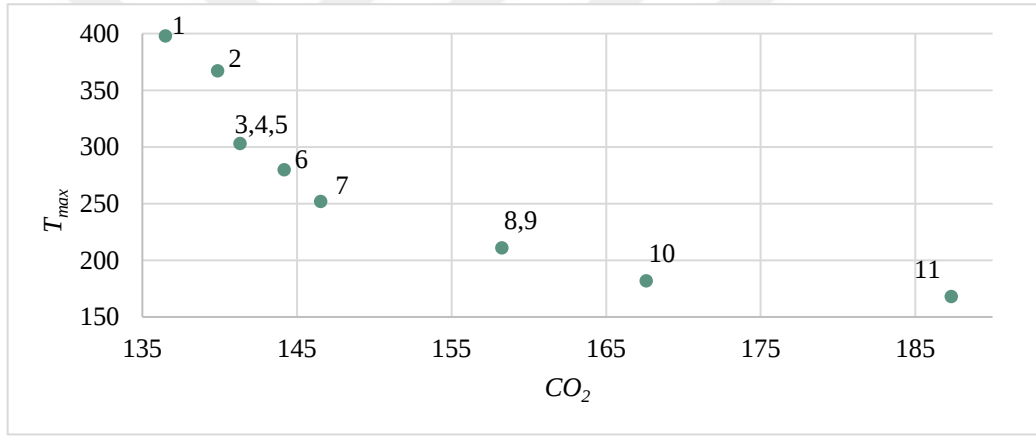
Ardından ele alınan herhangi bir amaç için ödünleşim matrisinden değer aralığı elde edilmiş ve bu aralık eşit alt aralıklara bölünerek ızgara noktaları belirlenmiştir. Belirlenen bu noktalar ilgili amacın sağ taraf sabiti olarak gevşekten sıkıya doğru modele girilerek her bir nokta etrafındaki çözümler taranmıştır.

Buna göre amaç fonksiyonunun sadece toplam CO_2 emisyonu salınımının en küçüklenmesi olarak belirlendiği durumda maksimum gecikmenin en küçüklenmesi için belirlenen 11 adet ızgara noktası gevşekten sıkıya doğru Çizelge 5.5'in ikinci sütununda verilmiştir. Bu değerler sağ taraf sabiti olarak kullanılarak toplam CO_2 emisyonu salınımı en küçüklenmiş ve bu değerler de çizelgenin üçüncü sütununda verilmiştir. Dördüncü sütunda ise ilgili ızgara nokta değerlerine göre elde edilen maksimum gecikme miktarları yer almaktadır. Çizelgeden görüldüğü gibi bu değerler ikinci sütununda yer alan sağ taraf sabitinden küçük eşit durumundadır.

Çizelge 5.5. Örnek probleme ilişkin Pareto optimal çözümlerin elde edilmesi

Sınıf numarası	Izgara nokta değeri	$\sum CO_2$	T_{max}
1	≤ 398	136,50	398
2	≤ 375	139,88	367
3	≤ 352	141,33	303
4	≤ 329	141,33	303
5	≤ 306	141,33	303
6	≤ 283	144,17	280
7	≤ 260	146,55	252
8	≤ 237	158,25	211
9	≤ 214	158,25	211
10	≤ 191	167,59	182
11	≤ 168	187,34	168

Böylece 11 adet sağ taraf sabitinin kısıt olarak modele girilmesiyle toplamda 8 farklı Pareto optimal çözüm Şekil 5.3'teki gibi elde edilmiştir. Problemden yer alan iki amacın birbiriyle çeliştiği Şekil 5.3'ten açıkça görülmektedir.

**Şekil 5.3.** Örnek probleme ait Pareto yüzey

Örnek problem için çözüm süresi 624,07 saniye olarak elde edilmiştir. CPLEX ile elde edilen 8 farklı Pareto çözüme ilişkin grafikler EK-1'de verilmiştir. EK-1'de yer alan grafikler incelendiğinde maksimum gecikmenin en küçük olduğu çözümde sistemde yer alan üç farklı tip araç da eş zamanlı olarak kullanılmış bu sayede atıl beklemler en küçüklenerek gecikme miktarı en küçüklenmiştir. Tam tersi durumda yani CO_2 emisyon miktarının en küçük olduğu çözümde ise araç kapasitesine göre emisyon salınımı artışı göz önünde bulundurularak sadece kapasitesi en küçük olan araç kullanılmıştır.

5.2.3. Pareto yerel arama

Önceki bölümde görüldüğü gibi AEKY yöntemi sadece 7 müşterinin olduğu ve sadece 10 ızgara noktasının olduğu durumda bile 10 dakika gibi bir sürede Pareto çözümleri elde edebilmiştir. Bu nedenle gerçek hayat problemleri için kısa sürede optimal ya da optmale yakın çözümler elde edilebilmesi amacıyla sezgisel bir algoritmanın kullanılması kaçınılmaz olmuştur.

Bu bölümde problemin çözümünde kullanılan iki aşamalı Pareto Yerel Arama (PYA) yaklaşımı incelenmiştir (Costa ve ark., 2018; Lust ve Teghem, 2010). Algoritma iki aşamadan oluşmaktadır. Birinci aşamada problem tek amaçlı problemlere indirgenerek bu problemler için potansiyel etkin çözümler elde edilirken ikinci aşamada elde edilen çözümler PYA ile iyileştirilmeye çalışılarak bu çözümlerden hareketle henüz elde edilmemiş etkin çözümler elde edilmektedir.

Pareto yerel arama yaklaşımının birinci aşaması Çizelge 5.6'da verilmiştir. Başlangıç çözümler elde edilirken amaç fonksiyonu ağırlıklı toplam şeklinde formüle edilir (Adım 0). Problem önce sadece birinci amacı, sonra da sadece ikinci amacı dikkate alarak çözülür (Adım 1). Elde edilen çözümlerle etkin çözüm kümesi oluşturulur (Adım 2). PYA algoritmasının ikinci aşaması başlatılır (Adım 3).

Çizelge 5.6. PYA algoritmasının birinci aşaması

Adımlar	Açıklama
Adım 0:	Problem parametrelerini belirle ve problemin amaç fonksiyonunu $(\lambda_1 f_1 + \lambda_2 f_2)$ şeklinde formüle et.
Adım 1:	Başlangıç çözümleri oluştur.
Adım 1.1:	Problemi sadece birinci amacı dikkate alarak çöz ve x_r çözümünü elde et. $(\lambda_1, \lambda_2) = (1,0)$
Adım 1.2:	Problemi sadece ikinci amacı dikkate alarak çöz ve x_s çözümünü elde et. $(\lambda_1, \lambda_2) = (0,1)$
Adım 2:	Etkin çözüm kümesini x_r ve x_s ile oluştur.
Adım 3:	Yinelemeli sezgiseli uygula.

PYA algoritmasının ikinci aşaması ise yinelemeli olarak çalışmaktadır ve Çizelge 5.7'de verilmiştir. Öncelikle etkin çözüm kümesi içerisindeki x_r ve x_s çözümleri belirlenir (Adım 0). Ele alınan x_r ve x_s çözümlerine göre λ_1 ve λ_2 değerleri güncellenir (Adım 1). Güncellenen ağırlıklarla problem çözülür (Adım 2). Yeni oluşturulan çözümün etkin çözüm kümesindeki diğer çözümlerle karşılaştırılması ve etkin çözüm kümesinin güncellenmesi gerçekleştirilir (Adım 3). Çizelge 5.8'de etkin çözüm kümesi oluşturulurken oluşabilecek durumlar verilmiştir. Eğer etkin çözüm kümesi içerisindeki en az bir çözüm, elde edilen çözüme zayıf bir şekilde baskınsa yeni çözüm dikkate alınmaz ve Çizelge 5.7'de verilen algoritma sonlandırılır. Eğer elde edilen yeni çözüm x_r

ve x_t ile oluşan üçgensel bölgenin içerisindeyse Çizelge 5.7’de verilen algoritma hem x_r ve x_t çözümleri için hem de x_t ve x_s çözümleri için yeniden çalıştırılır (Adım 4).

Çizelge 5.7. PYA algoritmasının ikinci aşaması

Adımlar	Açıklama
Adım 0:	Etkin çözüm kümesi içerisindeki x_r ve x_s çözümlerini belirle.
Adım 1:	λ_1 ve λ_2 değerlerini Adım 0’da ele alınan çözümlere göre aşağıdaki şekilde güncelle. $\lambda_1 = \frac{f_2(x_r) - f_2(x_s)}{f_2(x_r) - f_2(x_s) + f_1(x_s) - f_1(x_r)}, \quad \lambda_2 = 1 - \lambda_1$
Adım 2:	Problemi güncellenen λ_1 ve λ_2 değerleri ile Çizelge 5.9’da verilen prosedür ile çöz ve p çözümünü oluştur.
Adım 3:	Etkin çözüm kümesini Çizelge 5.8’de verilen prosedüre göre güncelle.
Adım 4:	Eğer elde edilen yeni çözüm x_t , x_r ve x_s ile oluşan üçgensel bölgenin içerisindeyse; x_r ve x_t çözümleri için Yinelemeli sezgiseli uygula. x_t ve x_s çözümleri için Yinelemeli sezgiseli uygula.

Çizelge 5.8 Etkin çözüm kümesinin güncellenmesi

Adımlar	Açıklama
Adım 1:	Elde edilen çözümü (p) etkin çözüm kümesi ($\widehat{X}_E$) içerisindeki tüm çözümlerle karşılaştır. Eğer etkin çözüm kümesi içerisindeki en az bir çözüm, elde edilen çözüme zayıf bir şekilde baskınsa yeni çözüm dikkate alınmaz ve algoritma sonlandırılır. Eğer elde edilen yeni çözüm etkin çözüm kümesindeki herhangi bir çözüme baskın geliyorsa etkin çözüm kümesinden bu çözüm çıkartılır. $\widehat{X}_E \leftarrow \widehat{X}_E \setminus \{x\}$ Eğer yukarıdaki her iki durum da sağlanmıyorsa elde edilen yeni çözüm baskın olmayan çözüm olarak etkin çözüm kümesine eklenir. $\widehat{X}_E \leftarrow \widehat{X}_E \cup \{p\}$

İki aşamalı PYA çok amaçlı bir çözüm yöntemi olup problemin yapısına göre algoritmanın çağırdığı, problem çözüm süreci değiştirilerek tüm problemlere kolaylıkla uyarlanabilmektedir. İki aşamalı PYA yaklaşımında, AT_ÇA Problemi için Değişken Komşu Arama Sezgiseli (Variable Neighbour Search) kullanılmıştır. Çizelge 5.9’da verilen çözüm yönteminin daha iyi anlaşılabilmesi için öncelikle üretim ortamına ilişkin gösterimin gerçek bir çözüme dönüştürülmesi süreci açıklanmıştır. Üretim ortamının gösteriminde atölye tipi çizelgeleme problemlerinde sıklıkla kullanılan tekrarlamalı permütasyon (permutation with repetition) yapısı kullanılmıştır. Bierwirth (1995) tarafından önerilen bu gösterim Parametreleri Çizelge 5.2’de verilen problem için Şekil 5.4’te örneklenmiştir. Şekil 5.4’ün ilk satırında sistemdeki işler operasyon sayısı kadar tekrar etmiştir. İkinci satırda ise her bir işin kaçınıcı operasyonunun çizelgeleneceği bilgisi yer almaktadır. Şekil 5.4’teki çözüme göre önce 4,6,3 ve 5 numaralı işlerin ilk operasyonu soldan sağa doğru sırasıyla çizelgelenir ardından 6 numaralı işin ikinci operasyonu (ikinci makinedeki) çizelgelenir. 6 numaralı siparişin ikinci makinede üretime başlayabilmesi için ikinci makinedeki önceki operasyonun (5.iş) üretiminin tamamlanması ve 6 numaralı işin ilk operasyonunun (1.makine) tamamlanması gerekmektedir. Bu şekilde soldan sağa

doğru tüm operasyonlar bu kısıtlara dikkat edilerek çizelgelenir. Bu gösterimin en büyük avantajı uygun çözüm durumu hiçbir zaman bozulmayacağından ekstra bir kontrol mekanizmasına ihtiyaç duyulmamaktadır.

4	6	3	5	6	4	4	1	2	5	1	7	7	7	6	1	3	5	3	2	2
1	1	1	1	2	2	3	1	1	2	2	1	2	3	3	3	2	3	3	2	3

Şekil 5.4. Üretim süreci için örnek bir çözüm gösterimi

Üretim ortamına ilişkin gösterimin çözüme dönüştürülmesi açıklandıktan sonra bu bölümde Roshanaei ve ark. (2009) tarafından atölye tipi çizelgeleme problemleri için önerilen Değişken Komşu Arama Algoritmasının genel uygulama adımları Çizelge 5.9’da verilmiştir. Öncelikle başlangıç çözüm belirli bir kurala göre oluşturulur. AT_ÇA Probleminde amaç fonksiyonlarından biri en büyük gecikmenin en küçüklenmesi olduğu için, En Erken Teslim Tarihi kuralı uygulanmıştır (Adım 0). Başlangıç çözüm oluşturulduktan sonra bu çözüm üzerinde birinci komşu arama stratejisi uygulanır, eğer çözüm iyileştiyse yine birinci komşu arama stratejisine dönülür, eğer bir iyileşme sağlanamadıysa ikinci komşu arama stratejisine geçilir. Bu şekilde çözümün iyileşip iyileşmeme durumlarına göre üç tip komşu arama stratejisi durma kriteri sağlanana kadar Adım 1’deki gibi uygulanır.

Çizelge 5.9. Üretim sürecinde kullanılan algoritmanın adımları

Adımlar	Açıklama
Adım 0:	EDD kuralına göre başlangıç çözümü oluştur ve $k = 1$ atamasını gerçekleştir.
BAŞLA	Durma kriteri sağlanıncaya kadar aşağıdaki adımları uygula.
Adım 1:	k . Komşu arama algoritmasını (NSS_k) uygula.
Adım 2:	Eğer çözüm iyileştiyse $k = 1$ atamasını gerçekleştir ve Adım 1’e dön. Aksi takdirde $k = k + 1$ atamasını gerçekleştir ve Adım 1’e dön. Eğer ($k = 4$) ise $k = 1$ atamasını gerçekleştir ve Adım 1’e dön.
BİTİR	Durma kriteri sağlanınca algoritmayı sonlandır ve üretim ortamını raporla.

Roshanaei ve ark. (2009) tarafından önerilen komşu arama stratejileri temelde aynı prensiple çalışmakta olup, mevcut çizelgeden çıkarılacak olan operasyon sayısı, bu operasyonların nasıl seçildikleri ve çıkarılan operasyonların yeni bir çizelge oluşturmak için nasıl yeniden yerleştirileceği kararlarına göre farklılaşmaktadır. Çizelge 5.10’da verilen ilk komşu arama stratejisinde mevcut çizelgeden rassal olarak ve yinelenmesiz bir şekilde seçilen tek bir operasyon kaldırılmakta (Adım 3) ve bu operasyon rassal olarak

yeniden konumlandırılmaktadır (Adım 4). Süreçte herhangi bir iyileşme gözlemediği anda güncel çizelge üzerinden yeniden aynı süreç tekrarlanmakta gözlenmediği durumda ise mevcut çizelgede daha önce seçilmemiş operasyonlar arasından rassal olarak seçilen bir operasyon için süreç başlatılmaktadır (Adım 5).

Çizelge 5.11’de verilen ikinci komşu arama stratejisinde ise mevcut çizelgeden rassal bir şekilde ardışık olarak seçilen iki operasyon kaldırılmakta (Adım 3) ve yeni çizelgeyi oluşturmak için rassal olarak yeniden konumlandırılmaktadır (Adım 4). İlk iyileşme gözlemediği anda mevcut çizelge güncellenerek bu çizelgeye ilk komşu arama stratejisi uygulanır. İyileşme gözlenmediği durumda ise bir sonraki ardışık operasyonları seçmek üzere algoritma yeniden başlatılır. Çizelge 5.12’de verilen son komşu arama stratejisinde ise mevcut çizelgeden rassal olarak seçilen üç operasyon yine rassal olarak yeniden konumlandırılır. Üçüncü komşu arama stratejisinde bu süreç belirli bir jenerasyon sayısı (ρ) kadar tekrarlanır. Benzer şekilde ilk iyileşme gözlemediği anda çizelge güncellenerek ilk komşu arama stratejisi başlatılır. İyileşme olmadığı durumda ise tüm jenerasyon boyunca elde edilen en iyi çözüm mevcut çözümden daha kötü olsa bile mevcut çözüm olarak ele alınır. Bunun sebebi arama boyunca yerel optimale takılmayı önlemektir. İlk komşu arama stratejisi algoritmanın yoğunlaşma stratejisini desteklerken, ikinci ve üçüncü komşu arama stratejisi ile de çözüm uzayının farklı bölgeleri taranarak çeşitlendirme stratejisi desteklenmektedir. Çizelge 5.13’te ise üretim ortamı oluşturulan çözüme ait dağıtım ortamı belirlenmektedir.

Çizelge 5.10. Birinci komşu arama stratejisi(NSS_1)

Adımlar	Açıklama
Adım 0:	$iyileşme = true$ atamasını gerçekleştir.
BAŞLA	$iyileşme = true$ olduğu sürece aşağıdaki adımları uygula.
Adım 1:	$iyileşme = false$ atamasını gerçekleştir. $i = 1$ değeri ata.
Adım 2:	$(i \leq toplam\ operasyon\ sayısı)$ olduğu sürece aşağıdaki adımları uygula.
Adım 3:	Mevcut sıralamadan (π) rassal olarak seçilen h . operasyonunu kaldır.
Adım 4:	Kaldırılan operasyonu mevcut sıralamada rassal bir pozisyona yerleştir (π').
Adım 5:	Eğer yeni oluşturulan çözüm mevcut çözümden daha iyi ise $\pi = \pi'$ olarak güncelle. $i = toplam\ operasyon\ sayısı$ atamasını gerçekleştir. $iyileşme = true$ atamasını gerçekleştir. Değilse herhangi bir değişiklik yapma. $i = i + 1$ atamasını gerçekleştir ve Adım 2’ye dön.
Adım 6:	$(i > toplam\ operasyon\ sayısı)$ olunca Adım 1’e dön.
BİTİR:	$iyileşme = false$ olunca algoritmayı sonlandır.

Çizelge 5.11. İkinci komşu arama stratejisi(NSS_2)

Adımlar	Açıklama
Adım 0:	$i = 1$ atamasını gerçekleştir.
BAŞLA	$(i \leq \text{toplam operasyon sayısı})$ olduğu sürece aşağıdaki adımları uygula.
Adım 1:	$ii = i + 1$ atamasını gerçekleştir.
Adım 2:	$(ii \leq \text{toplam operasyon sayısı})$ olduğu sürece aşağıdaki adımları uygula.
Adım 3:	Mevcut sıralamadan i . ve ii . operasyonları kaldır.
Adım 4:	Kaldırılan operasyonları mevcut sıralamada rassal iki pozisyona yerleştir (π').
Adım 5:	Eğer yeni oluşturulan çözüm mevcut çözümden daha iyi ise $\pi = \pi'$ olarak güncelle. $ii = \text{toplam operasyon sayısı}$ atamasını gerçekleştir. $i = \text{toplam operasyon sayısı}$ atamasını gerçekleştir. Değilse herhangi bir değişiklik yapma. $ii = ii + 1$ atamasını gerçekleştir ve Adım 2'ye dön.
Adım 6:	$(ii > \text{toplam operasyon sayısı})$ olunca $i = i + 1$ atamasını gerçekleştir ve Adım 1'e dön.
BİTİR	$(i > \text{toplam operasyon sayısı})$ olunca algoritmayı sonlandır.

Çizelge 5.12. Üçüncü komşu arama stratejisi(NSS_3)

Adımlar	Açıklama
Adım 0:	ρ parametresini belirle. $i = 0$ atamasını gerçekleştir.
BAŞLA	$(i \leq \rho)$ olduğu sürece aşağıdaki adımları uygula.
Adım 1:	Mevcut sıralamadan rassal olarak seçilen üç operasyonu kaldır.
Adım 2:	Kaldırılan operasyonları mevcut sıralamada rassal üç pozisyona yerleştir (π').
Adım 3:	$i = i + 1$ atamasını gerçekleştir ve Adım 1'e dön.
BİTİR	$(i > \rho)$ olunca algoritmayı sonlandır ve $\pi = eniyi(\pi')$ olarak güncelle.

Çizelge 5.13. Dağıtım sürecinde kullanılan algoritmanın adımları

Adımlar	Açıklama
Adım 1:	Değişken komşu arama algoritması sonucu elde edilen üretim çizelgesinde işleri üretim bitiş zamanlarına göre sırala.
Adım 2:	Bu dev turu araç kapasitesini göz önünde bulundurarak CO ₂ emisyonu açısından optimal bir şekilde alt turlara böl ve başlangıç çözümü elde et.
BAŞLA	Durma kriteri sağlanana kadar aşağıdaki adımları uygula.
Adım 3:	Komşu arama stratejilerini uygula.
Adım 4:	En iyi iyileşmeyi sağlayan komşu çözümü seçerek başlangıç çözümü güncelle.
BİTİR	Aynı üretim ortamı için iyileştirilen dağıtım ortamını raporla.

Buna göre Çizelge 5.9 sonlandığında elde edilen çözümde işler üretim bitiş zamanlarına göre sıralanır (Adım 1). Ardından bu dev tur CO₂ emisyonu açısından optimal bir şekilde Prins (2004) algoritması ile alt turlara bölünür (Adım 2). Sistemde üç tip araç olduğundan tur bölünmesi sürecinde ilgili alt turda en az CO₂ emisyonunu sağlayan araç dikkate alınmaktadır. Elde edilen başlangıç çözüm üzerinde farklı komşu arama stratejileri uygulanır (Adım 3) ve iyileşme olması durumunda başlangıç çözüm güncellenerek (Adım 4) güncel çözüm üzerinde komşu arama stratejileri uygulanır.

5.2.4. PYA yaklaşımının örnek test verisine uygulanması

İki aşamalı PYA yaklaşımının önceki bölümde incelenen örnek probleme uygulama adımları Çizelge 5.14'te verilmiştir. Çizelgenin ilk sütununda etkin çözüm kümesi içerisinde incelenecek olan çözümler yer almaktadır. İkinci sütunda birinci sütunda yer alan çözümlere göre güncellenen λ değerleri verilirken üçüncü sütunda bu değerlere göre yapılan arama prosedürü sonucu elde edilen çözüm verilmiştir. Dördüncü sütunda yeni üretilen çözüme ilişkin bilgi yer alırken son sütunda güncellenen etkin çözüm kümesi verilmiştir.

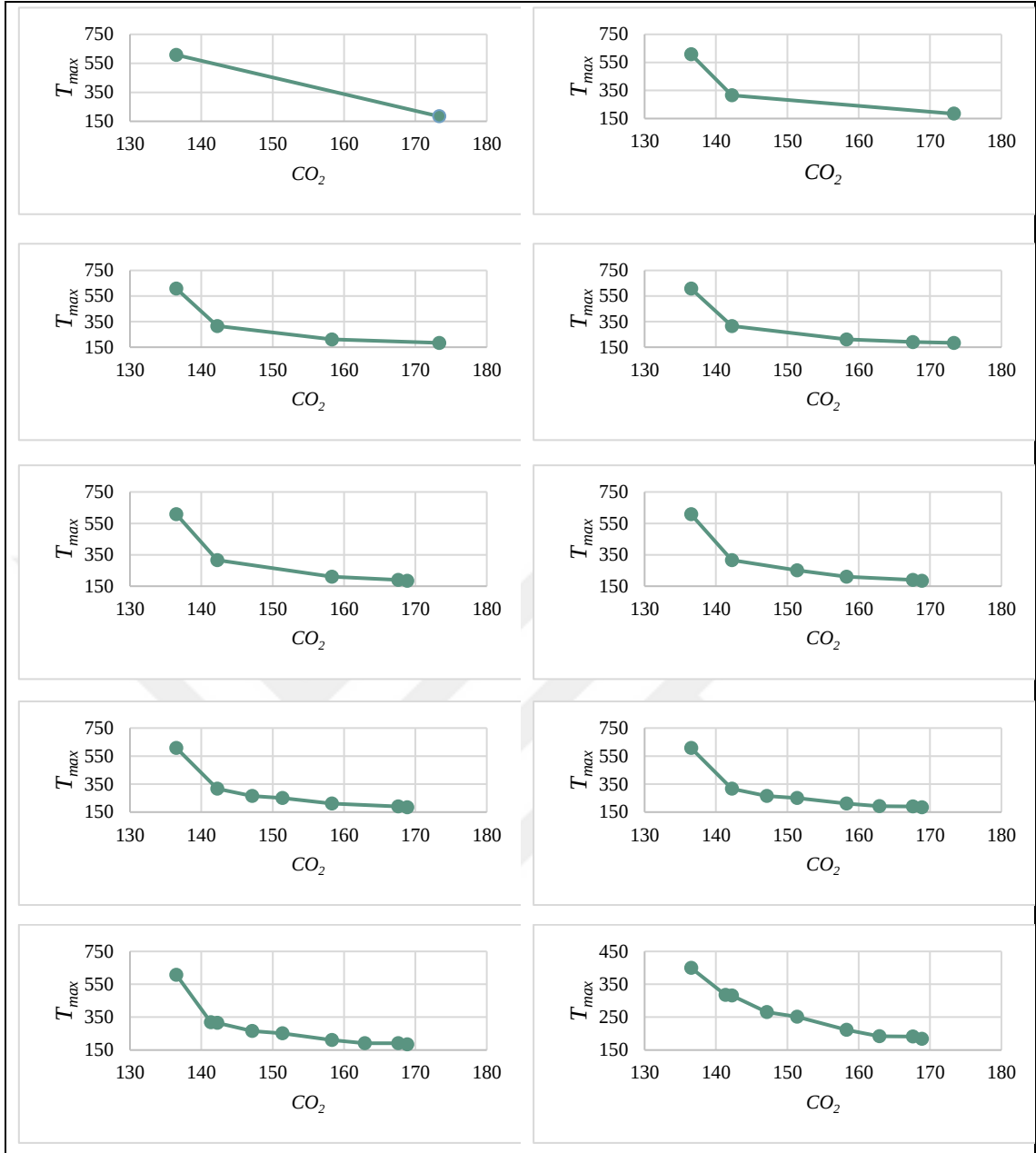
Çizelge 5.14. PYA yaklaşımının örnek probleme uygulanması

x_R ve x_S	λ	x_T	Durum	$\widehat{X}_E$
$x_R = \{184, 173, 340\}$ $x_S = \{608, 136, 500\}$	$\lambda = \{0,0799, 0,9201\}$	$x_T = \{316, 142, 232\}$	Δ	184, 173, 340 316, 142, 232 608, 136, 500
$x_R = \{184, 173, 340\}$ $x_S = \{316, 142, 232\}$	$\lambda = \{0,1907, 0,8093\}$	$x_T = \{211, 158, 299\}$	Δ	184, 173, 340 211, 158, 299 316, 142, 232 608, 136, 500
$x_R = \{184, 173, 340\}$ $x_S = \{211, 158, 299\}$	$\lambda = \{0,3578, 0,6422\}$	$x_T = \{191, 167, 588\}$	Δ	184, 173, 340 191, 167, 588 211, 158, 299 316, 142, 232 608, 136, 500
$x_R = \{184, 173, 340\}$ $x_S = \{191, 167, 588\}$	$\lambda = \{0,4511, 0,5489\}$	$x_T = \{184, 168, 837\}$	Baskın	184, 168, 837 191, 167, 588 211, 158, 299 316, 142, 232 608, 136, 500
$x_R = \{184, 168, 837\}$ $x_S = \{191, 167, 588\}$	$\lambda = \{0,1514, 0,8486\}$	$x_T = \{211, 158, 299\}$		184, 168, 837 191, 167, 588 211, 158, 299 316, 142, 232 608, 136, 500
$x_R = \{191, 167, 588\}$ $x_S = \{211, 158, 299\}$	$\lambda = \{0,3172, 0,6828\}$	$x_T = \emptyset$		184, 168, 837 191, 167, 588 211, 158, 299 316, 142, 232 608, 136, 500
$x_R = \{211, 158, 299\}$ $x_S = \{316, 142, 232\}$	$\lambda = \{0,1327, 0,8673\}$	$x_T = \{251, 151, 374\}$	Δ	184, 168, 837 191, 167, 588 211, 158, 299 251, 151, 374 316, 142, 232 608, 136, 500

Çizelge 5.14. PYA yaklaşımının örnek probleme uygulanması (devam)

x_R ve x_S	λ	x_T	Durum	$\widehat{X}_E$
$x_R = \{211, 158, 299\}$ $x_S = \{251, 151, 374\}$	$\lambda = \{0,1476, 0,8524\}$	$x_T = \emptyset$		184, 168,837 191, 167,588 211, 158,299 251, 151,374 316, 142,232 608, 136,500
$x_R = \{251, 151, 374\}$ $x_S = \{316, 142, 232\}$	$\lambda = \{0,1233, 0,8767\}$	$x_T = \{265, 147, 128\}$	Δ	184, 168,837 191, 167,588 211, 158,299 251, 151,374 265, 147,128 316, 142,232 608, 136,500
$x_R = \{251, 151, 374\}$ $x_S = \{265, 147, 128\}$	$\lambda = \{0,2327, 0,7673\}$	$x_T = \{192, 162, 883\}$		184, 168,837 191, 167,588 192, 162,883 211, 158,299 251, 151,374 265, 147,128 316, 142,232 608, 136,500
$x_R = \{265, 147, 128\}$ $x_S = \{316, 142, 232\}$	$\lambda = \{0,0876, 0,9124\}$	$x_T = \{318, 141, 328\}$		184, 168,837 191, 167,588 192, 162,883 211, 158,299 251, 151,374 265, 147,128 316, 142,232 318, 141,328 608, 136,500
$x_R = \{316, 142, 232\}$ $x_S = \{608, 136, 500\}$	$\lambda = \{0,0193, 0,9807\}$	$x_T = \{400, 136, 500\}$	Baskın	184, 168,837 191, 167,588 192, 162,883 211, 158,299 251, 151,374 265, 147,128 316, 142,232 318, 141,328 400, 136,500
$x_R = \{316, 142, 232\}$ $x_S = \{400, 136, 500\}$	$\lambda = \{0,0639, 0,9361\}$	$x_T = \emptyset$		184, 168,837 191, 167,588 192, 162,883 211, 158,299 251, 151,374 265, 147,128 316, 142,232 318, 141,328 400, 136,500

Çizelge 5.14'te yer alan çözüm sürecinin grafiksel gösterimi Şekil 5.5'te verilmiştir.



Şekil 5.5. Örnek probleme ilişkin PYA sürecinin işleyişi

5.2.3. Baskın olmayan sıralama genetik algoritma-II

Baskın-olmayan Sıralama Genetik Algoritma-II (BSGA-II), Deb ve ark. (2002) tarafından geliştirilen çok amaçlı evrimsel bir algoritmadır. Algoritma, Srinivas ve Deb (1994)'in daha önce geliştirmiş oldukları Baskın-olmayan Sıralama Genetik Algoritma (BSGA) algoritmasının dezavantajlarını gidermek amacıyla ortaya çıkmıştır. BSGA algoritmasında çeşitlilik, uyum değerleri paylaşımı ile gerçekleştirilirken bu değer paylaşım parametresi ile belirlenmektedir. Bu parametre ise algoritmanın performansını önemli ölçüde etkilemektedir. Bu nedenle BSGA-II algoritmasında parametresiz bir çeşitlilik mekanizması geliştirilmek istenmiştir. BSGA-II algoritmasında bu çeşitlilik

yoğunluk mesafesi (crowding distance) kavramı ile sağlanmaktadır. Çizelge 5.15'te BSGA-II algoritmasının uygulama adımları verilmiştir. Çizelge 5.15'te görüldüğü gibi öncelikle başlangıç popülasyon oluşturulur (Adım 1) ve popülasyonda yer alan bireylerin uyum değerleri hesaplanır (Adım 2). Ardından bireylerin uyum değerlerine göre Pareto seviyeleri belirlenir (Adım 3). Algoritma başlatıldıktan sonra çocuk popülasyonu yeniden üretim ve mutasyon operatörleri uygulanarak oluşturulur (Adım 4). Ebeveyn ve çocuk popülasyonunda yer alan bireylerin uyum değerleri hesaplanarak (Adım 5) bu değerlere göre bireylerin Pareto seviyeleri ve yoğunluk mesafeleri hesaplanır (Adım 6). En düşük Pareto seviyesi ve en yüksek yoğunluk mesafeli bireylerden başlanarak popülasyon güncellenir (Adım 7) ve durma kriteri sağlanınca popülasyonda yer alan Pareto çözümler raporlanır.

Çizelge 5.15. BSGA-II'nin temel adımları

Adımlar	Açıklama
Adım 0:	Algoritmada kullanılan parametreleri belirle.
Adım 1:	Başlangıç popülasyonunu P_k oluştur.
Adım 2:	Popülasyonda yer alan bireylerin uyum değerlerini hesapla.
Adım 3:	Uyum değerlerine göre bireylerin Pareto seviyelerini belirle.
BAŞLA	Durma kriteri sağlanana kadar aşağıdaki adımları uygula.
Adım 4:	Çocuk popülasyonunu Q_k oluştur. Turnuva seçim yöntemi uygula. Çaprazlama ve mutasyon operatörlerini uygula.
Adım 5:	Ebeveyn ve çocuk popülasyonunda yer alan bireylerin $(P_k + Q_k)$ uyum değerlerini hesapla.
Adım 6:	Uyum değerlerine göre tüm bireylerin Pareto seviyelerini belirle ve yoğunluk mesafesi değerlerini hesapla.
Adım 7:	En düşük Pareto seviyesi ve en yüksek yoğunluk mesafeli bireylerden başlayarak popülasyonu P_{k+1} güncelle.
BİTİR	
Adım 8	Pareto çözümleri raporla.

BSGA-II algoritmasına ilişkin genel yapı verildikten sonra bu bölümde algoritmada kullanılan genetik gösterim, kromozomun genetik kodunun çözümü, seçim yöntemi, genetik operatörler ve yerel arama prosedürü detaylı olarak açıklanmıştır.

Bütünleşik problemin üretim aşaması için genetik gösterim atölye tipi çizelgeleme problemlerinde sıklıkla kullanılan tekrarlamalı permütasyon şeklinde belirlenmiştir. Bierwirth (1995) tarafından önerilen bu gösterimde, N iş M de işlerin operasyon sayısını ifade etmek üzere bir kromozom toplam $M \times N$ sayıda genden oluşmaktadır. Dağıtım ortamı ise tur bölünmesi olmaksızın dev bir turla ifade edilmektedir. Dağıtım ortamı için permütasyon gösterim yapısı kullanılmıştır ve N müşteri sayısını ifade etmek üzere bir dağıtım aşaması N sayıda genden oluşmaktadır. Sonuç olarak tam bir çözüme ait

kromozom yapısı, üretim ve dağıtım ortamlarının birleştirilmesiyle $(M \times N) + N$ sayıda genden oluşmaktadır.

Başlangıç popülasyonu oluşturulurken ilk bireyin üretim ortamı en yakın teslim tarihi kuralına göre, ikinci bireyin üretim ortamı ise en yakın komşu sezgiseline göre elde edilen sıralamaya göre belirlenmiş diğer bireylerin üretim ortamları tamamen rassal şekilde oluşturulmuştur. Popülasyonda yer alan bireylerin dağıtım ortamları ise işlerin üretim bitiş zamanları dikkate alınarak belirlenmiştir. Buna göre müşteriler üretim bitiş zamanlarına göre küçükten büyüğe doğru sıralanmıştır.

Popülasyonda yer alan bireylerin çözüm kaliteleri değerlendirilirken öncelikle bireylerin Pareto seviyeleri belirlenmektedir. Pareto seviyeleri ilgili çözüme baskın gelen çözüm sayısına göre belirlenmektedir. Buna göre bir birey diğer bireyler tarafından domine edilemiyorsa bu birey baskın olmayan çözüm olarak adlandırılmakta ve Pareto seviyesi “0” olarak belirlenmektedir. Bireylerin Pareto seviyeleri belirlendikten sonra çözümler Pareto seviyelerine göre küçükten büyüğe sıralanır. Seçim sürecinde aynı Pareto seviyesinde yer alan iki birey karşılaştırılırken ise bireylerin yoğunluk mesafelerine bakılır. Yoğunluk mesafesi büyük olan birey mevcut çözümlerden mümkün olduğunca uzağa giderek nispeten keşfedilmemiş bölgelerde arama yapmaya olanak sağlamaktadır. Bir çözümün yoğunluk mesafesi hesaplanırken seçilen bir amaç için alt ve üst sınır değerlerine sahip olan çözümlere mesafe olarak sonsuz değeri atanır. m . amaca ilişkin j çözümün yoğunluk mesafesi d_m^j olmak üzere, bu değer m . amaca ilişkin tüm çözümlerin en büyük f_m^{max} ve en küçük f_m^{min} uyum değerleri kullanılarak Eşitlik (5.31) ile hesaplanmaktadır. d^j ile ifade edilen toplam yoğunluk mesafesi ise tüm amaçlar için ayrı ayrı hesaplanan mesafelerin toplamı ile Eşitlik (5.32) de görüldüğü gibi elde edilmektedir.

$$d_m^j = \frac{f_m^{j+1} - f_m^{j-1}}{f_m^{max} - f_m^{min}} \quad \forall m \in P; j \in S \quad (5.31)$$

$$d^j = \sum_m d_m^j \quad \forall j \in S \quad (5.32)$$

Önerilen algorithmda turnuva seçim yöntemi kullanılmıştır. Turnuva büyüklüğü iki olarak belirlendiğinde popülasyondan rassal olarak seçilen iki birey arasından Pareto seviyelerine göre diğerine galip gelen birey ebeveyn olarak seçilmiştir. Bireylerin Pareto

seviyeleri aynı ise bu kez yoğunluk mesafesi en büyük olan birey diğerine galip gelerek ebeveyn olarak seçilmektedir.

Çaprazlama operatörü olarak tekrarlı permütasyon gösterimi için önerilen üç farklı çaprazlama stratejisi kullanılmıştır. Bunlardan ilki Bierwirth (1995) tarafından önerilen genelleştirilmiş sıra çaprazlama (GSC) operatörüdür. İkincisi Bierwirth ve ark. (1996) tarafından önerilen genelleştirilmiş parça çaprazlama (GPC) operatörüdür. Üçüncüsü ise yine Bierwirth ve ark. (1996) tarafından önerilen öncelik koruyucu çaprazlama (ÖKÇ) operatörüdür. GPC operatörü uygulanırken öncelikle bir numaralı ebeveynin belirli bir bölümü seçilir. Bu alt dizinin büyüklüğü yoğunlukla kromozomda yer alan genlerin sayısının bir bölü üçü ile yarısı arasında seçilmektedir. Bir numaralı ebeveyne ait kromozomda yer alan genlerin tekrar etme sayıları belirlenir. Ardından iki numaralı ebeveynde bir numaralı ebeveynde belirlenen alt dize bu dizedeki genlerin ranklarına göre silinir. Çocuk kromozom oluşturulurken bir numaralı ebeveynde seçilen alt dizinin ilk geni dikkate alınarak yerleştirme yapılır. Buna göre çocuk gende bir numaralı ebeveyndeki alt dize başlangıç noktasından itibaren kopyalanır. Diğer genler iki numaralı ebeveynden sırasıyla alınır. GSC operatörü uygulanırken ise ilk adımlar GPC operatörü ile aynı olup çocuk kromozom oluşturulurken bir numaralı ebeveynde seçilen alt dizinin ilk geninin rankı dikkate alınarak yerleştirme yapılır. ÖKÇ operatörü uygulanırken ise çocuk kromozomdaki gen sayısı kadar gen içeren bir seçim kromozomu oluşturulur. Seçim kromozomundaki genler rassal olarak 0 ve 1 değerlerini alır. Genler 0 değerini almışsa çocuk kromozomun geni bir numaralı ebeveynden, 1 değerini almışsa iki numaralı ebeveynden alınır. Her seferinde alınan gen ebeveynlerden silinir.

Yeni bireyin üretim ortamı belirlendikten sonra dağıtım ortamı belirlenmektedir. Atıl bekleme zamanlarını azaltmak amacıyla işler üretim bitiş zamanlarına göre küçükten büyüğe doğru sıralanmaktadır. Ardından oluşturulan dev tur için CO₂ emisyonu açısından optimal bölünmeyi veren Prins (2004) algoritmasıyla alt turlar oluşturulmaktadır. Bu çözüm verilen üretim sıralaması için oluşturulan başlangıç dağıtım sıralamasıdır. Aynı üretim ortamına sahip farklı dağıtım ortamlarını değerlendirmek amacıyla başlangıç çözüm üzerinden dağıtım ortamı için yerel arama algoritması uygulanmaktadır. Yerel arama algoritmasında kullanılan komşuluk yapıları swap, insert, 2-opt ve araç değişimi operatörleridir. Swap operatörü rassal olarak seçilen iki müşterinin birbiriyle yer değiştirmesi yoluyla uygulanmaktadır. Insert operatöründe rassal olarak seçilen bir müşteri rassal olarak seçilen bir konuma yerleştirilir. 2-opt operatörü aynı turda ve farklı turda olmak üzere iki şekilde uygulanmaktadır. Aynı turda 2-opt operatörü uygulanırken

ardışık olmayan iki ark silinerek bu arklar arasında yer alan müşteriler yer değiştirir. Farklı turda 2-opt operatörü uygulanırken, her bir tur rassal olarak seçilen arklara göre iki parçaya ayrılır. Başlangıç ve bitiş şeklinde farklı turların parçaları bağlanarak yeni turlar oluşturulur. Araç değişim operatörü ise rassal olarak seçilen bir turda hizmet veren aracın değiştirilmesi yoluyla uygulanmaktadır. Tüm operatörler uygulanırken araç kapasitesi kısıtı göz önünde bulundurularak uygun çözüm yapısı her zaman korunmaktadır.

Mutasyon operatörü olarak kromozomun üretim kısmında sıra bazlı ve pozisyon bazlı olmak üzere iki operatör kullanılmıştır. Sıra bazlı mutasyonda rassal olarak seçilen iki gen birbiriyle yer değiştirmekte, pozisyon bazlı mutasyonda ise rassal olarak seçilen bir gen rassal olarak seçilen farklı bir pozisyona yerleştirilmektedir. Kromozomun dağıtım kısmında ise mutasyon operatörü yine sıra bazlı ve pozisyon bazlı şekilde dev tur üzerinde uygulanmıştır.

Popülasyonun güncellenmesi aşamasında, yeni üretilen bireylerle birlikte popülasyon büyüklüğünün iki katı kadar birey değerlendirilmektedir. Bireyler yine öncelikle Pareto seviyelerine göre sıralanır. Ardından en düşük seviyeden en yükseğe doğru popülasyon güncellenir. Pareto seviyeleri aynı olan bireyler arasından seçim yaparken yine yoğunluk mesafesi ölçütü kullanılır.

5.2.4. BSGA-II yaklaşımının örnek test verisine uygulanması

Bu bölümde BSGA-II yaklaşımının daha iyi anlaşılabilmesi için Çizelge 5.2’de parametreleri verilen örnek üzerinde algoritmanın adımları bir jenerasyon uygulanmıştır.

Adım 0: Yapılan test koşulları sonucunda önerilen BSGA-II için algoritma parametreleri Çizelge 5.16’daki gibi belirlenmiştir.

Çizelge 5.16. Önerilen algoritmada kullanılan parametreler

Algoritma parametresi	Değer
Popülasyon büyüklüğü	= 30
Turnuva büyüklüğü	= 2
Çaprazlama oranı	= 0,90
Mutasyon oranı	= 0,10
Maksimum jenerasyon sayısı	= 500 (Küçük ve orta boyutlu) = 10 dakika (Büyük boyutlu)

Adım 1: Örnek test verisi dikkate alınarak ilk bireyin üretim ortamı en yakın teslim tarihi kuralına göre, ikinci bireyin üretim ortamı ise en yakın komşu sezgiseline göre elde edilen sıralamaya göre belirlenmiş diğer bireylerin üretim ortamları tamamen rassal şekilde oluşturulmuştur. Popülasyonda yer alan bireylerin dağıtım ortamları ise

işlerin üretim bitiş zamanları dikkate alınarak belirlenmiştir. Örnek olarak *Birey 1*'de yer alan işler en yakın teslim tarihine göre çizelgelendiğinde üretim bitiş zamanları sırasıyla 339, 867, 616, 157, 507, 149 ve 656 olarak elde edilmiş böylece dağıtım ortamı üretim bitiş zamanlarına göre 6,4,1,5,3,7 ve 2 şeklinde elde edilmiştir. Dağıtım ortamında yer alan dev turun bölünmesi aşamasında CO₂ emisyonu dikkate alınarak Prins (2004) algoritması uygulanmıştır. Çizelge 5.17'de olası alt turlara ilişkin CO₂ miktarı hesaplaması verilmiştir. Çizelgenin ilk sütununda sistemde yer alan en büyük kapasiteli aracın kapasitesi baz alınarak olası tüm tur kombinasyonları oluşturulmuştur. İkinci sütunda ilk sütunda yer alan iki düğüm arasında seyahat ederken ilgili aracın üzerindeki yük miktarı üçüncü sütunda ise bu düğümler arası mesafe verilmiştir. Dördüncü sütunda ilk sütundaki tur için kapasite kısıtına göre hizmet verebilecek araçlar verilirken son sütunda ise uygun araçlara göre toplam emisyon miktarları hesaplanmıştır.

4	4	4	6	6	6	1	1	1	5	5	5	3	3	3	7	7	7	2	2	2	6	4	1	5	3	7	2
Üretim aşaması																		Dağıtım aşaması									

Şekil 5.6. Örnek bir kromozom gösterimi

Çizelge 5.17. Olası alt turlara ilişkin emisyon hesaplanması

Uygun Turlar		Toplam yük	Toplam mesafe	Uygun araçlar	CO ₂
0-6-0	0-6	50	45	V1, V2, V3	18,96 27,96 36,96
	6-0	0	45		
	0-6	102	45		
0-6-4-0	6-4	52	102	V1, V2, V3	45,44 66,04 86,64
	4-0	0	59		
	0-6	162	45		
0-6-4-1-0	6-4	112	102	V2, V3	103,82 134,82
	4-1	60	109		
	1-0	0	54		
	0-6	247	45		
0-6-4-1-5-0	6-4	197	102	V3	185,06
	4-1	145	109		
	1-5	85	80		
	5-0	0	69		
	0-6	247	45		
0-4-0	0-4	52	59	V1, V2, V3	24,91 36,71 48,51
	4-0	0	59		
0-4-1-0	0-4	112	59	V1, V2, V3	50,02 72,23 94,43
	4-1	60	109		
	1-0	0	54		
0-4-1-5-0	0-4	197	59	V2, V3	109,77 141,47
	4-1	145	109		
	1-5	85	80		
	5-0	0	69		
0-1-0	0-1	60	54	V1, V2, V3	22,99 33,79 44,59
	1-0	0	54		
0-1-5-0	0-1	145	54	V2, V3	67,17 87,47
	1-5	85	80		
	5-0	0	69		
0-1-5-3-0	0-1	240	54	V3	118,35
	1-5	180	80		
	5-3	95	94		
	3-0	0	29		
0-5-0	0-5	85	69	V1, V2, V3	30,11 43,91 57,71
	5-0	0	69		
0-5-3-0	0-5	180	69	V2, V3	66,75 85,95
	5-3	95	94		
	3-0	0	29		
0-5-3-7-0	0-5	264	69	V3	132,45
	5-3	179	94		
	3-7	84	73		
	7-0	0	51		
0-3-0	0-3	95	29	V1, V2, V3	12,78 18,58 24,38
	3-0	0	29		
0-3-7-0	0-3	179	29	V2, V3	50,75 66,05
	3-7	84	73		
	7-0	0	51		
0-3-7-2-0	0-3	259	29	V3	120,74
	3-7	164	73		
	7-2	80	93		
	2-0	0	78		
0-7-0	0-7	84	51	V1, V2, V3	22,24 32,44 42,64
	7-0	0	51		
0-7-2-0	0-7	164	51	V2, V3	73,37 95,57
	7-2	80	93		
	2-0	0	78		
0-2-0	0-2	80	78	V1, V2, V3	33,87 49,47 65,07
	2-0	0	78		

Ardından Prins (2004) tarafından önerilen algoritma maliyet fonksiyonu yerine emisyon miktarı kullanılarak ilgili dev tura Çizelge 5.18’ deki gibi uygulanmıştır. Sistemdeki araçlar CO₂ emisyonu açısından değerlendirildiğinde optimal bölünmeyi sağlamak için her bir olası alt turda CO₂ emisyonunu en küçükleyen araç dikkate alınmıştır.

Çizelge 5.18. Prins algoritmasının uygulanması

$V_0 = 0;$ $V_1 = V_2 = V_3 = V_4 = V_5 = V_6 = \infty$	Eğer $V_{i-1} + z_{i-1,j} < V_j$ ise $V_j \leftarrow V_{i-1} + z_{i-1,j}$
$V_0 + 18,96 < V_1$	$V_1 = 18,96$
$V_0 + 45,44 < V_2$	$V_2 = 45,44$
$V_0 + 103,82 < V_3$	$V_3 = 103,82$
$V_0 + 185,06 < V_4$	$V_4 = 185,06$
$V_1 + 24,91 < V_2$	$V_2 = 43,87 (18,96+24,91)$
$V_1 + 50,02 < V_3$	$V_3 = 68,98$
$V_1 + 109,77 < V_4$	$V_4 = 128,73$
$V_2 + 22,99 < V_3$	$V_3 = 66,86 (43,87+22,99)$
$V_2 + 67,17 < V_4$	$V_4 = 111,04$
$V_2 + 118,35 < V_5$	$V_5 = 162,22$
$V_3 + 30,11 < V_4$	$V_4 = 96,97 (66,86+30,11)$
$V_3 + 66,75 < V_5$	$V_5 = 133,61$
$V_3 + 132,45 < V_6$	$V_6 = 199,31$
$V_4 + 12,78 < V_5$	$V_5 = 109,75 (96,97+12,78)$
$V_4 + 50,75 < V_6$	$V_6 = 147,72$
$V_4 + 120,74 < V_7$	$V_7 = 217,71$
$V_5 + 22,24 < V_6$	$V_6 = 131,99 (109,75+22,24)$
$V_5 + 73,37 < V_7$	$V_7 = 183,12$
$V_6 + 33,87 < V_7$	$V_7 = 165,86 (131,99+33,87)$

Sonuç olarak (0-6-4-1-5-3-7-0) dev turu için toplam CO₂ miktarı dikkate alınarak Prins (2004) algoritması uygulandığında optimal bölünme her turda tek bir müşteri olacak şekilde ve her müşteriye en küçük kapasiteli araç hizmet verecek şekilde sağlanmıştır. Bu durumda toplam CO₂ miktarı 165,86 olarak elde edilmiştir. Popülasyondaki diğer bireyler de aynı yaklaşımla oluşturulmuştur.

Adım 2: Popülasyonda yer alan bireylerin uyum değerleri için Eşitlik (5.1) ve Eşitlik (5.2)’de verilen amaç fonksiyonları hesaplanır.

Adım 3: Popülasyonda yer alan bireylerin uyum değerlerine göre Pareto seviyeleri belirlenir.

Adım 4: Çaprazlama operatörü olarak üç farklı çaprazlama stratejisi önerilmiş olup bu operatörler bireylerin üretim ortamına ilişkin kromozom parçaları üzerinde uygulanmaktadır. Şekil 5.7’de yeniden üretim aşamasında GSC ve GPÇ operatörlerine ilişkin birer örnek verilmiştir.

Ebeveyn1:	5	6	4	3	6	4	4	1	2	5	1	5	7	7	2	3	1	7	3	6	2
Rank	1	1	1	1	2	2	3	1	1	2	2	3	1	2	2	2	3	3	3	3	3

Ebeveyn2:	4	4	6	4	2	3	5	6	7	5	7	7	6	4	4	1	3	5	3	2	2
Rank	1	1	1	2	1	1	1	2	1	2	2	3	3	3	2	3	2	3	3	2	3

GSC:	4	6	6	4	4	1	2	5	1	3	5	7	7	7	6	1	3	5	3	2	2
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

GPC:	4	6	3	5	6	4	4	1	2	5	1	7	7	7	6	1	3	5	3	2	2
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Şekil 5.7. GSC ve GPC operatörlerinin uygulanışı

Şekil 5.8’de ise son çaprazlama operatörü olan OKÇ operatörü örneklenmiştir.

Ebeveyn1:	5	6	4	3	6	4	4	4	2	5	1	5	7	7	2	3	1	7	3	6	2
-----------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Ebeveyn2:	4	1	6	4	2	3	5	6	7	5	7	7	6	4	4	4	3	5	3	2	2
-----------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Rassal	1	0	0	1	1	1	0	0	0	1	1	0	1	0	0	0	1	1	0	1	0
--------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

OKÇ:	4	5	6	1	4	2	3	6	4	7	5	1	7	5	2	3	7	6	1	3	2
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Şekil 5.8. OKÇ operatörünün uygulanışı

Çaprazlama operatörü sadece üretim aşamasında gerçekleştirilip yeni oluşturulan bireylerin başlangıçtaki dağıtım ortamları üretim bitiş zamanlarına göre belirlenmektedir. Ardından aynı üretim ortamı için daha kaliteli dağıtım ortamlarının araştırılması amacıyla yerel arama yaklaşımı uygulanmaktadır. Buna göre dağıtım ortamına ilişkin komşu çözümler Şekil 5.9’da örneklendirilmiştir. Kromozomun üretim ortamına göre çizelgelenen işler üretim bitiş zamanlarına göre sıralandıklarında 6,4,7,1,5,3,2 şeklinde dev tur oluşmaktadır. Bu dev tur Prins algoritmasına göre alt turlara bölünmüş ve dağıtım ortamı için başlangıç çözümü oluşturulmuştur.

4	6	3	5	6	4	4	1	2	5	1	7	7	7	6	1	3	5	3	2	2	6	4	7	1	5	3	2
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Başlangıç çözümü	Turlar	CO ₂	T _{max}
6-4-7-1-5-3-2	6 4,7 1 5 3 2	155,488	492

Şekil 5.9. Dağıtım ortamı için başlangıç çözümü

Çizelge 5.19. Dağıtım ortamı için komşu çözümlerin amaç fonksiyonları

Operatör	Turlar	CO ₂	T _{max}
Swap	6 7,4 1 5 3 2	155,022	492
İnsert	4,7 6,1 5 3 2	143,267	518
2-opt	6 4,7 1 5 3,2	163,166	439
Araç değişimi	6 4,7 1 5 3 2	169,288	354

Çizelge 5.19’da görüldüğü gibi turlar oluşturulurken uygun çözüm durumu korunmuş ve kapasite kısıtı dikkate alınmıştır. Komşu çözümler incelendiğinde swap operatörü ile elde edilen çözüm başlangıç çözüme baskındır. Komşu çözümler değerlendirildiğinde ise dört komşu çözüm baskın olmayan çözüm durumundadır. Dağıtım ortamı için PYA yaklaşımında kullanılan ve Çizelge 5.13’te verilen algoritma BSGA-II için de kullanılmıştır. Ancak çözüm kalitesi değerlendirilirken PYA yaklaşımında iki amaç fonksiyonunun ağırlıklı toplamı dikkate alınırken BSGA-II yaklaşımında çözümlerin Pareto seviyeleri ve yoğunluk mesafeleri değerlendirilmektedir.

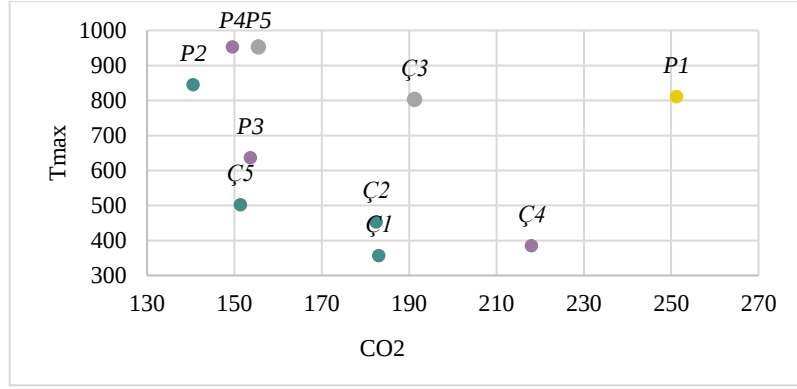
Mutasyon operatörü uygulanırken sıra bazlı mutasyonda rassal olarak seçilen iki gen birbiriyle yer değiştirmekte, pozisyon bazlı mutasyonda ise rassal olarak seçilen bir gen rassal olarak seçilen farklı bir pozisyona yerleştirilmektedir. Mutasyon operatörü dağıtım ortamında dev tur üzerinde de uygulanmaktadır.

Adım 5: Yeni üretilen bireylerle birlikte popülasyon büyüklüğünün iki katı kadar bireyin her iki amaç fonksiyonu değeri belirlenir.

Adım 6: Amaç fonksiyonlarına göre popülasyonda yer alan ve yeni oluşturulan bireylerin Pareto seviyeleri ve yoğunluk mesafeleri hesaplanır. Çizelge 5.20’deki popülasyon büyüklüğünün 5 olduğu bir durum için Pareto seviyeleri ve yoğunluk mesafelerinin hesaplanması örneklendirilmiştir.

Çizelge 5.20. Pareto seviyelerinin belirlenmesi

Bireyler	CO ₂	T _{max}	Domine eden çözümler	Pareto seviyesi	Yoğunluk mesafesi
P1	251,27	811	P3, Ç1, Ç2, Ç3, Ç4, Ç5	6	-
P2	140,59	845	-	0	∞
P3	153,70	636	Ç5	1	2
P4	149,54	953	P2	1	∞
P5	155,49	953	P2, P3, P4, Ç5	4	∞
Ç1	183,09	357	-	0	∞
Ç2	182,45	453	-	0	1,04
Ç3	191,27	803	P3, Ç1, Ç2, Ç5	4	∞
Ç4	218,01	385	Ç1	1	∞
Ç5	151,37	502	-	0	1,79



Şekil 5.10. Popülasyondaki bireylerin Pareto seviyeleri

Şekil 5.10 incelendiğinde Pareto seviyesi en düşük bireyler yani baskın olmayan bireyler $P2$, $\Ç1$, $\Ç2$ ve $\Ç5$ 'tir. Buna göre yeni popülasyonda bu dört birey yer alır. Beşinci birey ise bu bireylerden sonra Pareto seviyesi en düşük olan bireyler arasından yoğunluk mesafesine göre seçilir. Pareto seviyesi 1 olan $P3$, $P4$ ve $\Ç4$ bireyleri için yoğunluk mesafeleri önceki bölümde verilen Eşitlik (5.31) ve (5.32) ile hesaplanmıştır. Buna göre yoğunluk mesafesi en yüksek olan $P4$ veya $\Ç4$ beşinci birey olarak seçilerek popülasyon güncellenir. Örnekte baskın olmayan çözümlere ilişkin yoğunluk mesafesi hesaplaması Çizelge 5.21'de verilmiştir.

Çizelge 5.21. Yoğunluk mesafelerinin hesaplanması

Bireyler	CO_2	T_{max}	Yoğunluk mesafesi
$P2$	140,59	845	∞
$\Ç1$	183,09	357	∞
$\Ç2$	182,45	453	$\frac{183,09 - 151,37}{183,09 - 140,59} + \frac{502 - 357}{845 - 357} = 1,04$
$\Ç5$	151,37	502	$\frac{182,45 - 140,59}{183,09 - 140,59} + \frac{845 - 453}{845 - 357} = 1,79$

Adım 7: Popülasyon en düşük Pareto seviyesi ve en yüksek yoğunluk mesafesi kuralına göre güncellenir son durumda güncel popülasyonda $P2$, $\Ç1$, $\Ç5$, $\Ç2$ ve $P4$ bireyleri yer almaktadır.

5.2.5. AT_ÇA Problemi için test problemlerinin üretilmesi

Bu bölümde önerilen algoritmaların performanslarını değerlendirmek amacıyla kullanılan test problemlerinin üretilme mekanizması açıklanmıştır. Çizelgede 5.22'de görüldüğü gibi üretilen test problemleri müşteri ve makine sayılarına göre küçük, orta ve

büyük boyutlu problemler olmak üzere üç grupta incelenmiştir. Bunun dışında problem üzerinde teslim tarihlerinin etkisini araştırmak amacıyla teslim tarihleri sıkı, gevşek ve tüm müşterilerin aynı teslim tarihine sahip olduğu ortak sıkı ve ortak gevşek olmak üzere dört farklı seviye incelenmiştir. Araç kapasitesinin etkisi ise sıkı ve gevşek olmak üzere iki farklı seviyede incelenmiştir. Her bir kombinasyona ilişkin 3 örnek üretilmiştir. Böylece 144 küçük boyutlu, 288 orta boyutlu ve 192 büyük boyutlu olmak üzere toplamda 624 örnek incelenmiştir.

Çizelge 5.22. İncelenen örneklem sayıları

Örnek	N	M	V	dd	Q
Küçük boyutlu	5,6,7	2,3	3	Sıkı, gevşek, ortak sıkı, ortak gevşek	Sıkı, gevşek
Orta boyutlu	10,11,12,13,14,15	3,4	3	Sıkı, gevşek, ortak sıkı, ortak gevşek	Sıkı, gevşek
Büyük boyutlu	20,30,40,50	4,5	3	Sıkı, gevşek, ortak sıkı, ortak gevşek	Sıkı, gevşek

Çizelge 5.23. İncelenen problem parametrelerinin üretilmesi

Parametre	Değer	
d_i	$\sim UNIF\left(\frac{\rho}{2}, \rho\right)$	$\rho = 100$ (5.33)
p_{im}	$\sim UNIF(1 - \mu, 1 + \mu)d_i$	$\mu = 0,2$ (5.34)
$Q_{sıkı}$	$= 20 \times n, 25 \times n, 30 \times n$	(5.35)
$Q_{gevşek}$	$= 20 \times n, 30 \times n, 40 \times n$	(5.36)
$\rho_{sıkı}^0; \rho_{sıkı}^*$	$= 0,2, 0,25, 0,30; 0,26, 0,315, 0,30$	(5.37)
$\rho_{gevşek}^0; \rho_{gevşek}^*$	$= 0,2, 0,30, 0,40; 0,26, 0,39, 0,52$	(5.38)
α	$= \frac{\sum_i \sum_m p_{im}}{n}$	(5.39)
ω	$= \sqrt{\frac{\alpha^2}{2}}$	(5.40)
(X_0, Y_0)	$= \left(\frac{\omega}{2}, \frac{\omega}{2}\right)$	(5.41)
(X_i, Y_i)	$\sim UNIF(0, \omega)$	(5.42)
$dd_i(gevşek)$	$\sim UNIF\left(\left(\sum_m p_{im} + t_{oi}\right), \left(2 \times \sum_m p_{im} + t_{oi}\right)\right)$	(5.43)
$dd_i(sıkı)$	$\sim UNIF\left(\left(\sum_m p_{im} + t_{oi}\right), \left(1,5 \times \sum_m p_{im} + t_{oi}\right)\right)$	(5.44)
$dd_i(ortakgevşek)$	$= \left(2 \times \max_i \left(\sum_m p_{im} + t_{oi}\right)\right)$	(5.45)
$dd_i(ortaksıkı)$	$= \left(1,5 \times \max_i \left(\sum_m p_{im} + t_{oi}\right)\right)$	(5.46)

Çizelge 5.23'te ise incelenen problem parametrelerinin üretilme mekanizması verilmiştir. Test problemlerinin üretilme süreci, parametre üretiminde kullanılan genel bir ρ değerinin belirlenmesiyle başlamıştır ($\rho = 100$). Eşitlik (5.33)'de görüldüğü gibi sistemdeki müşteri talepleri bu ρ değeri kullanılarak düzgün dağılıma göre belirlenmiştir. Ardından her bir ürünün her bir makinedeki birim üretim zamanları parametreleri $(1 - \mu)$ ve $(1 + \mu)$ olan düzgün dağılıma uyacak şekilde belirlenerek, üretim zamanları ilgili talep ve birim üretim zamanının çarpılmasıyla Eşitlik (5.34) ile elde edilmiştir. İşlerin makinelerdeki operasyon sıraları rassal olarak belirlenmiştir. Sistemde yer alan üç tip araç için kapasiteler müşteri sayısının bir fonksiyonu olarak sıkı ve gevşek seviyede sırasıyla Eşitlik (5.35) ve Eşitlik (5.36)'ya göre üretilmiştir. Araç kapasitelerine göre araçların yakıt tüketim oranları araçlar boşken ve tam kapasite doluyken sıkı ve gevşek seviyede sırasıyla Eşitlik (5.36) ve Eşitlik (5.37)'ye göre üretilmiştir. Bu değerler Ubeda ve ark. (2011)'nin yapmış olduğu çalışma baz alınarak belirlenmiştir. Müşteriler X ve Y koordinatlarına yerleştirilirken iki nokta arasındaki en büyük mesafenin belirli bir değeri (α) aşmaması sağlanmıştır. Bu değer Eşitlik (5.39)'da görüldüğü gibi tüm işlerin tüm makinelerdeki işlem sürelerinin toplamının toplam iş sayısına bölünmesiyle elde edilmiştir. Eşitlik (5.41)'de depo düğümün koordinatları belirlenmiştir. Ardından sistemde yer alan müşteriler arasındaki öklit uzaklık bu eşik değerini geçmeyecek şekilde müşteriler rassal olarak X ve Y koordinatlarına Eşitlik (5.42) ile konumlandırılmıştır. Son olarak teslim tarihleri belirlenirken alt sınır olarak i işe ait tüm makinelerdeki üretim zamanları toplamı ile depodan i . düğüme ulaşım süresi toplamı alınmıştır. Üst sınırlar ise sıkı, gevşek, ortak sıkı ve ortak gevşek teslim tarihi düzeylerine göre Eşitlik (5.43) – Eşitlik (5.46)'ya göre belirlenmiştir.

5.3. AT_ÇA Problemi için Karşılaştırmalı Sonuçlar

Bu bölüm önerilen çok amaçlı KTDP modelinin performansı ile önerilen PYA ve BSGA-II yaklaşımlarının performanslarının karşılaştırılması olmak üzere iki başlıkta incelenmiştir. Geliştirilen matematiksel model GAMS 28.2 ara yüzünde kodlanmış ve matematiksel model çözücüsü olarak CPLEX kullanılmıştır. CPLEX için verilen süre sınırı 7200 sn olarak ayarlanmıştır.

Çizelge 5.24. Geliştirilen yöntemler için kullanılan performans kriterleri

İfade	Açıklama
PÇS	: Verilen süre sınırı içerisinde elde edilen Pareto çözüm sayısı
OÇS	: Matematiksel model ve önerilen algoritmaya ait ortalama saniye cinsinden koşum süresi
HV	: Hipervolüm değeri
PÇUPS	: Pareto çözüme ulaşılan problem sayısı

Çizelge 5.24'te ifade edilen HV değeri ve hesaplanması Bölüm 5.3.2'de detaylı bir şekilde açıklanmıştır.

5.3.1. Önerilen çok amaçlı KTDP modelinin sonuçları

Bu bölümde tez kapsamında incelenen problem için CPLEX sonuçları değerlendirilmiştir. Çizelge 5.25'te görüldüğü gibi müşteri sayısının artması hem üretim hem de dağıtım ortamını etkileyerek çözüm süresinde üstel bir artışa sebep olmaktadır. Sistemde 5 müşterinin olduğu durumda çözüm süresi ortalama 11 saniye iken bu süre 6 müşteride 84 saniyeye kadar ulaşmıştır. Müşteri sayısının 7 olduğu durumda bazı örnekler için CPLEX yetersiz gelerek verilen süre sınırları içerisinde ödünleşim matrisini oluşturamamış, diğer örnekler için ise ortalama çözüm süresi yaklaşık olarak 9 dakika olarak bulunmuştur.

Çizelge 5.25. Müşteri sayısının etkisi

N	PÇUPS	PÇS	OÇS
5	48	3,02	11,00
6	48	3,38	84,38
7	31	3,77	524,30

Çizelge 5.26'da ise CPLEX' in performansı makine, sayısı, teslim tarihi ve araç kapasiteleri açısından değerlendirilmiştir. Çizelge 5.26'nın ilk sütununda müşteri sayıları yer almaktadır. İkinci sütunda ise performans kriteri olarak sırasıyla Pareto çözüm sayısı, saniye cinsinden ortalama çözüm süresi ve verilen süre sınırları içerisinde çözüm elde edilebilen problem sayıları verilmiştir. Üçüncü, dördüncü ve beşinci sütunda ise sırasıyla makine sayısı, teslim tarihi ve kapasite seviyeleri yer almaktadır.

Çizelge 5.26. CPLEX' in değerlendirilmesi

<i>N</i>		<i>M</i>		<i>dd</i>				<i>Q</i>	
		<i>2</i>	<i>3</i>	<i>Gevşek</i>	<i>Sıkı</i>	<i>Ortak gevşek</i>	<i>Ortak sıkı</i>	<i>Gevşek</i>	<i>Sıkı</i>
<i>5</i>	PÇS	3,42	2,63	3,5	3,08	2	3,5	3	3,04
	OÇS	12,36	9,64	10,99	11,05	10,44	11,53	11,31	10,7
	PÇUPS	24	24	12	12	12	12	24	24
<i>6</i>	PÇS	3,38	3,38	4	3,42	2,33	3,75	3,42	3,33
	OÇS	99,23	69,53	46,06	54,71	81,62	155,13	103,14	65,62
	PÇUPS	24	24	12	12	12	12	24	24
<i>7</i>	PÇS	3	4,41	4,75	3,7	2,88	3,8	3,69	3,87
	OÇS	411,15	617,48	600,82	303,3	413,32	1021,44	417,53	638,18
	PÇUPS	14	17	8	10	8	5	16	15
<i>Ort.</i>	PÇS	3,27	3,47	4,08	3,40	2,40	3,68	3,37	3,41
	OÇS	174,25	232,22	219,29	123,02	168,46	396,03	177,33	238,17
	PÇUPS	62	65	32	34	32	29	64	63

Saniye cinsinden çözüm süresi problem parametrelerine bağlı olarak belirgin bir şekilde değişmektedir. Makine sayısındaki artışın OÇS değerine etkisi müşteri sayısındaki artışın etkisine göre nispeten daha az olmuştur. Teslim tarihlerinin ortak olduğu durumda çözüm uzayındaki alternatif çözümlerin artması beklendiğinden OÇS değeri artırmıştır. Kapasitenin daha sıkı olduğu durumda yine OÇS değeri artış göstermiştir.

5.3.2. Sezgisel algoritmaların değerlendirilmesi

Bu bölümde öncelikle çok amaçlı optimizasyon yöntemleri için kullanılan performans kriterleri hesaplanması örnek bir test verisi üzerinde açıklanmış ardından önerilen yöntemlerin bu kriterlere göre değerlendirilmesi yapılmıştır.

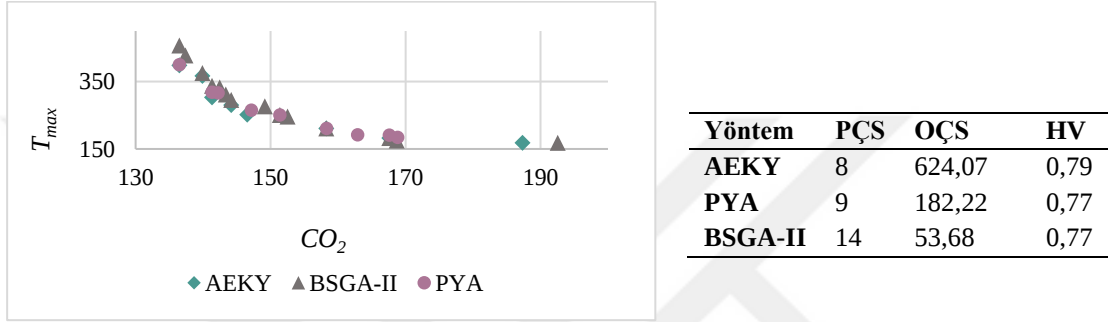
5.3.2.1. Çok amaçlı optimizasyon performans kriterleri

Riquelme ve ark. (2015) yapmış oldukları kaynak araştırması çalışmasına göre literatürde en fazla kullanılan çok amaçlı optimizasyon performans ölçütü hipervolum (HV)'dür. HV, üstten referans noktası ile sınırlandırılan baskın alanın boyutu olarak tanımlanmıştır ve Pareto yüzeyin baskın olduğu çözümlerin boyutunu vermektedir. HV ne kadar yüksekse Pareto yüzeye yaklaşım o kadar iyidir. *N* baskın olmayan çözüm sayısını ifade etmek üzere, iki amaçlı bir problem için HV hesaplama adımları Çizelge 5.27'de verilmiştir(Ferrández ve ark., 2017).

HV değeri hesaplanırken amaçlar arasında ölçekleme problemi oluşmaması için *i*. amaç fonksiyonuna ait *p*. çözüm Eşitlik (5.47)'de görüldüğü gibi normalleştirilir.

$$f_i^{norm p} = \frac{f_i^p - f_i^{min}}{f_i^{max} - f_i^{min}} \quad (5.47)$$

Buna göre önceki bölümde incelenen örnek probleme ilişkin Şekil 5.11'de verilen Pareto yüzeylere ait HV değerleri Çizelge 5.28'de hesaplanmıştır.



Şekil 5.11. Örnek probleme ilişkin Pareto yüzeyler

Şekil 5.11'de görüldüğü gibi örnek problem için önerilen algoritmalarla elde edilen Pareto yüzeyler AEKY ile elde edilen gerçek Pareto yüzeye oldukça yakın olup Pareto çözüm sayısı açısından BSGA-II yaklaşımıyla daha kısa sürede daha fazla baskın olmayan çözüm elde edilmiştir. AEKY' nin çözüm süresi açısından başarısız olduğu görülmektedir. HV değerleri incelendiğinde ise örnek problem için PYA ve BSGA-II algoritmalarının HV değerleri AEKY'ye oldukça yakın bulunmuştur.

5.3.2.2. Sezgisel algoritmaların performansları

Bu bölümde tez kapsamında incelenen küçük, orta ve büyük boyutlu örnekler için önerilen yöntemlerin performansları değerlendirilmiştir. Çizelge 5.29'un ilk sütununda problem karmaşıklığını doğrudan etkilediği ispatlanan müşteri sayısı parametresi verilmiştir. İkinci, üçüncü ve dördüncü sütunda ise sırasıyla PÇS, HV ve OÇS değerleri yer almaktadır. Buna göre önceki bölümde verilen test problemleri küçük, orta ve büyük boyutlu olmak üzere üç kısımda incelenmiştir. Küçük boyutlu örneklerde önerilen iki algoritma kesin yöntemle karşılaştırılmıştır. Küçük boyutlu örneklerde BSGA-II ile elde edilen Pareto çözüm sayısı, kesin yöntemle 10 aralık için elde edilen çözüm sayısına göre

daha fazladır. Genel olarak her üç yöntemle elde edilen Pareto çözüm sayısı ise küçük boyutlu örnekler için birbirine yakındır. HV değerleri incelendiğinde her iki algoritmada küçük boyutlu problemlerde gerçek Pareto yüzeye oldukça yaklaşmıştır. Algoritmalar kendi içlerinde değerlendirildiklerinde ise BSGA-II algoritması PYA algoritmasına göre daha fazla alan taramıştır. Çözüm sürelerine bakıldığında 5 müşteri için kesin yöntem oldukça kısa sürede çözüm üretebilirken müşteri sayısının 6 ve 7 olduğu örneklerde çözüm süresi açısından önerilen algoritmalara karşı başarısız bulunmuştur. Sistemde 7 müşterinin olduğu örneklerde ise kesin yöntemin Pareto yüzey elde etme süresi oldukça artmıştır. Çözüm süresi açısından algoritmalar değerlendirildiğinde yine en etkin algoritma BSGA-II olmuştur. PYA algoritması kesin yöntem kadar olmasa da çözüm süresi açısından müşteri sayısının artışından etkilenmektedir.

Sistemde 10 ile 15 müşteri arasında müşterinin olduğu orta boyutlu örnekler için BSGA-II algoritması 500 iterasyon için ortalama olarak 5 dakikada çözüm elde ederken PYA algoritması ortalama olarak 10 dakikada çözüme ulaşmıştır. Orta boyutlu örnekler için BSGA-II algoritması PYA algoritmasına göre daha kısa sürede daha fazla sayıda Pareto çözüm elde edebilmiştir. PYA algoritmasının müşteri sayısındaki artışa oldukça duyarlı olduğu orta boyutlu örnekler incelendiğinde görülebilir. Orta boyutlu örnekler için HV değerleri karşılaştırıldığında ise BSGA-II algoritması PYA algoritmasına göre daha fazla alan taramıştır.

Büyük boyutlu örnekler için PYA algoritmasının çözüm süresi açısından performansı oldukça zayıf bulunmuştur. Bu nedenle PYA algoritması durma kriteri 10 dakika ve 1 saat şeklinde belirlenerek çalıştırılmıştır. BSGA-II algoritması ise durma kriteri 10 dakika olacak şekilde çalıştırılmıştır. Çizelge 5.29'da görüldüğü gibi 10 dakika içerisinde 20 müşteri için PYA algoritması ile elde edilen Pareto yüzeyin BSGA-II ile elde edilen Pareto yüzeye oldukça yakın olduğu görülmektedir. Ancak müşteri sayısının 30, 40 ve 50 olduğu durumlarda PYA için verilen süre oldukça yetersiz gelmekte ve her iki algoritmayla elde edilen Pareto yüzeyler birbirinden uzaklaşmaktadır. Özellikle müşteri sayısının 50 olduğu durumda PYA algoritmasının HV değeri BSGA-II algoritmasına göre oldukça düşük bulunmuştur. Büyük boyutlu örnekler için PYA algoritmasının çözüm süresi 1 saat olarak verildiğinde ise Çizelge 5.30'da görüldüğü gibi algoritma performansı oldukça artmıştır. Şekil 5.12'de büyük boyutlu problemlerden rassal olarak seçilen dört farklı örnek için her iki algoritma ile elde edilen Pareto yüzeyler verilmiştir. X eksenini toplam CO₂ emisyonu değerini Y eksenini ise maksimum gecikme miktarını ifade etmektedir. Şekil 5.12'de görüldüğü gibi 20 müşterinin olduğu durumda

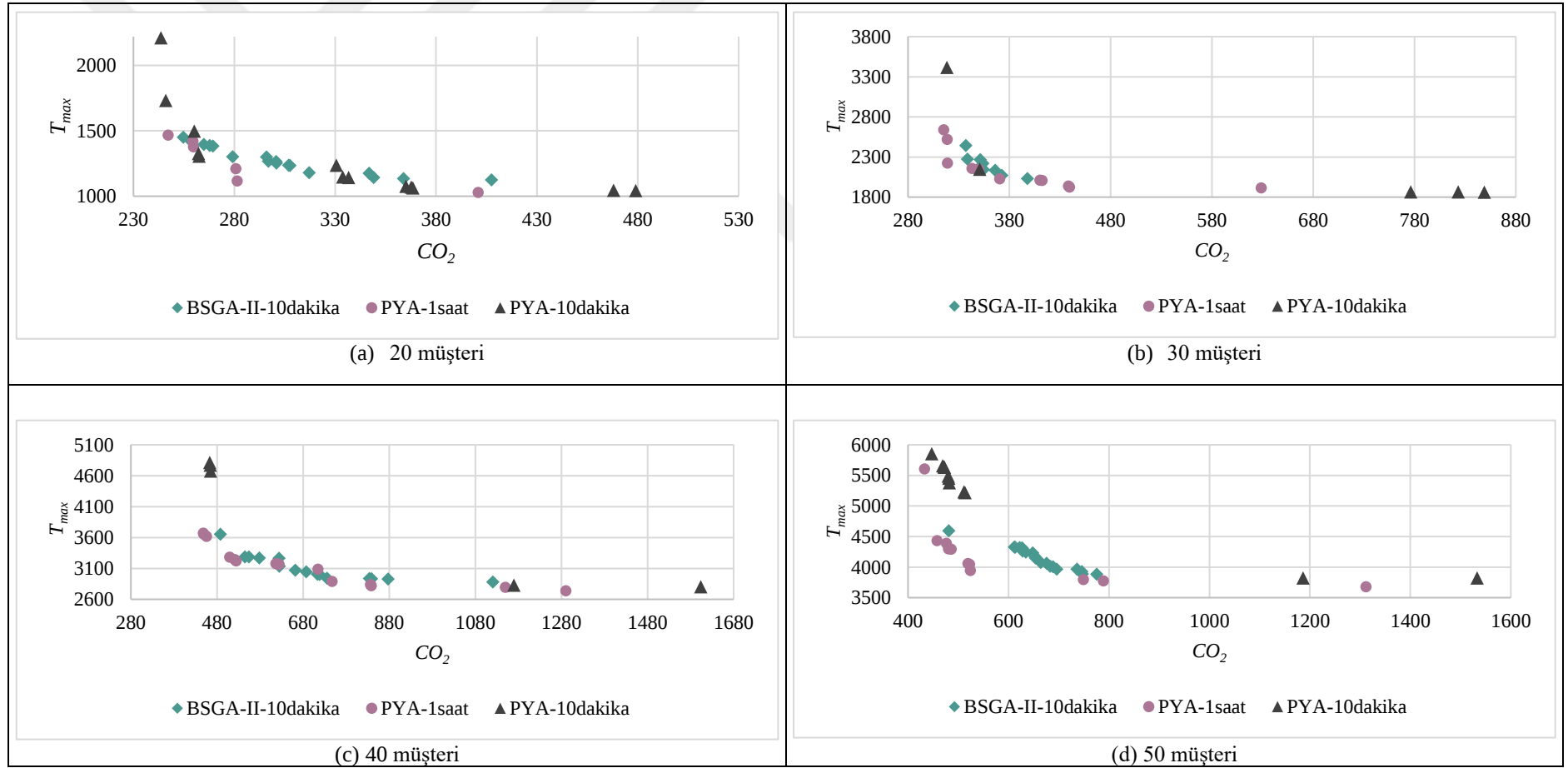
her iki algoritma ile maksimum 10 dakikada elde edilen Pareto yüzeyler birbirine oldukça yakındır. Müşteri sayısı arttıkça PYA algoritmasının 10 dakika içerisinde uç noktalardan Pareto çözüm elde etme becerisi azalmaktadır. Sonuç olarak PYA algoritmasının performansı çözüm süresinden oldukça etkilenmekte olup küçük ve orta boyutlu örnekler için başarılı bulunurken büyük boyutlu örnekler için operasyonel düzeyde kararların hızlı alınması gerektiği varsayımıyla yetersiz bulunmuştur. BSGA-II algoritması ise en büyük boyutlu örneklerde bile 10 dakika gibi kısa bir sürede etkin Pareto yüzeyler elde edebilmiştir.

Çizelge 5.29. Önerilen yöntemlerin karşılaştırılması

	<i>N</i>	PÇS			HV			OÇS		
		AEKY	PYA	BSGA-II	AEKY	PYA	BSGA-II	AEKY	PYA	BSGA-II
<i>Küçük</i>	5	3,02	2,63	3,23	0,420	0,367	0,374	11,00	25,44	40,55
	6	3,38	3,00	3,54	0,507	0,389	0,490	84,38	52,30	45,43
	7	3,77	3,26	4,61	0,653	0,499	0,545	524,30	95,48	48,97
<i>Orta</i>	10	-	4,79	8,67	-	0,705	0,830	-	444,48	306,14
	11	-	4,63	8,73	-	0,640	0,823	-	467,93	325,58
	12	-	4,98	10,31	-	0,733	0,867	-	569,61	332,68
	13	-	4,73	10,67	-	0,690	0,829	-	609,79	344,97
	14	-	5,02	12,21	-	0,718	0,807	-	790,67	355,34
	15	-	5,17	12,02	-	0,682	0,827	-	813,04	363,15
<i>Büyük</i>	20	-	6,06	11,42	-	0,768	0,785	-	600,00	600,00
	30	-	5,92	11,79	-	0,658	0,838	-	600,00	600,00
	40	-	5,71	12,13	-	0,391	0,867	-	600,00	600,00
	50	-	4,98	12,44	-	0,185	0,879	-	600,00	600,00

Çizelge 5.30. OÇS'nin PYA'nın performansına etkisi

	<i>N</i>	PÇS		HV		OÇS	
		PYA	BSGA-II	PYA	BSGA-II	PYA	BSGA-II
<i>Büyük</i>	20	6,15	11,42	0,762	0,723	3600,00	600,00
	30	6,96	11,79	0,779	0,691	3600,00	600,00
	40	7,73	12,13	0,788	0,658	3600,00	600,00
	50	6,77	12,44	0,780	0,707	3600,00	600,00



Şekil 5.12. Büyük boyutlu örnekler için elde edilen Pareto yüzeyler

6. SONUÇLAR VE TARTIŞMA

6.1 Sonuçlar

Günümüzde birçok işletme çeşitli kısıtlamalardan ötürü üretim ve dağıtım operasyonlarının kararlarını eş zamanlı olarak vermek zorundadır. Bu nedenler arasında müşterilerin belirttikleri teslim tarihinden önce teslimatlarının yapılmasını istemeleri, üretimi bitmiş parçalar için yeterli ara stok alanının bulunmaması, TZÜ politikasından kaynaklanan baskı, bozulabilir ürünlerin üretimi ve siparişe göre üretim politikası sayılabilir. Üretim kararları mevcut makine ortamında makinelerdeki siparişlerin operasyon sıralarının ve tamamlanma zamanlarının ne olacağını belirlerken, dağıtım kararları farklı müşterilere hangi dağıtım sırası ve zamanda teslimatların yapılacağını ifade etmektedir. Böylece BÜDÇ Probleminde, üretim çizelgeleme ve araç rotalama problemleri birlikte ele alınmaktadır. Her iki problemde literatürde polinomsal zamanda optimal çözümleri bulan bir algoritmanın olmadığı NP-Zor problem sınıfında olduğundan bütünlüklük problemin de NP-Zor yapıda olduğu söylenebilir.

Yapılan tez çalışmasında operasyonel düzey BÜDÇ Problemleri için literatürdeki boşluklar araştırılarak literatürde daha önce çalışılmamış olan üç farklı BÜDÇ Problemi incelenmiştir. Problemler incelenirken daha basit yapıdan daha karmaşık bir yapıya doğru gidilerek problemler aşamalı bir şekilde farklılaştırılmıştır.

PAT_TA Problemi olarak ifade edilen en temel problemde permütasyon akış tipi üretim ortamı ile sınırlı kapasiteli tek aracın dağıtım yaptığı dağıtım ortamı incelenmiştir. PAT_TA Problemi için amaç fonksiyonu toplam tur zamanı ile toplam gecikme toplamının en küçüklenmesi olarak ele alınmıştır. Dağıtım ortamında sistemde sınırlı kapasiteye sahip tek bir araç olduğundan tüm müşteri siparişlerinin karşılanabilmesi için araç birden fazla kez kullanılabilir. Böylece sistemde iki farklı durumda atıl bekleme oluşabilmektedir. Birinci durumda üretimi tamamlanan parti, dağıtımını yapacak olan aracın önceki turunu tamamlayarak depoya dönmesini beklemektedir. İkinci durumda ise depoda hazır durumda olan araç dağıtımını yapacağı partideki ürünlerin üretimlerinin tamamlanmasını beklemektedir. Atıl süreler açısından bir durumun diğerine baskın gelmesi BÜDÇ Probleminde bir problemin diğerine göre baskın gelmesi anlamına gelmektedir. Bu durum BÜDÇ Probleminin temel mantığına aykırı olduğundan test problemlerinin üretimi titizlikle gerçekleştirilmiştir. Literatürde hazır bir veri seti olmadığından test problemlerinin üretilmesi aşamasında bir problemin diğerine baskın gelmemesi adına üretim ve dağıtım süreleri arasında bir denge kurulmaya çalışılmıştır.

PAT_TA Problemi ilk olarak KTDP modeli ile formüle edilmiş ardından modelin CPLEX ile çözümü raporlanmıştır. Müşteri sayısına göre küçük, orta ve büyük boyutlu problemlerden oluşan test problemleri üzerinde CPLEX'in 10800 sn süre sınırı içerisinde performansı değerlendirildiğinde şu çıkarımlar ortaya çıkmıştır: CPLEX 180 küçük boyutlu örneğin hepsinde verilen süre sınırı içerisinde optimal çözümlere ulaşabilmiştir. Müşteri sayısındaki artışla birlikte CPLEX'in çözüm süresi üstel bir şekilde artmış orta boyutlu örneklerde bile verilen süre optimal çözümlerin elde edilmesi açısından yetersiz gelmiştir. Orta boyutlu 540 örneğin sadece 76 tanesinde (%14) CPLEX ile optimal çözümler elde edilmiştir. Büyük boyutlu örneklerde ise CPLEX tamamen yetersiz gelerek hiçbir örnekte optimal çözümü elde edememiştir. Yüzde sapma değerleri değerlendirildiğinde orta boyutlu örnekler için yüzde sapma değeri ortalama olarak 47,74 iken büyük boyutlu örneklerde bu değer 86,94'e kadar çıkmıştır. Böylece en temel yapıdaki PAT_TA Problemi için CPLEX yetersiz bulunmuştur. Ardından PAT_TA Problemi için makul sürelerde optimal ya da optimale yakın çözümler elde etmek amacıyla MA yaklaşımı önerilmiştir. Önerilen MA ile 180 küçük boyutlu örneğin 173 (%96) tanesinde ortalama 1,5 saniyede CPLEX ile elde edilen optimal çözümlere ulaşılmıştır. Orta boyutlu optimali bilinen 76 örneğin 70 tanesinde (%92) önerilen MA ile ortalama 3,8 saniyede bu çözümler elde edilmiştir. Önerilen MA orta boyutlu örneklerde yüzde sapma değeri açısından CPLEX ile yakın performans göstermiştir. Büyük boyutlu örneklerde ise önerilen MA'nın CPLEX'e göre performansı oldukça iyi bulunmuştur. Çözüm süreleri değerlendirildiğinde ise MA için en büyük boyutlu örnekte bile bir dakikadan kısa bir sürede CPLEX'in 10800 saniyede bulduğu çözüme göre oldukça başarılı çözümler elde edilmiştir. PAT_TA Problemi için ayrıca teslim tarihlerinin etkisi ve üretim sürecindeki değişkenliğin etkisi araştırılmıştır. Bu parametrelerin CPLEX'in başarısı üzerindeki etkisini araştırmak için orta boyutlu problemler incelenmiştir. Bunun sebebi CPLEX'in küçük boyutlu örneklerin hepsinde optimal çözüme ulaşması, büyük boyutlu problemlerde ise hiçbirinde optimal çözüm elde edememesidir. Buna göre müşterilerin belirledikleri son teslim tarihleri daraldıkça CPLEX ile elde edilen optimal çözüm sayılarının ciddi oranda azaldığı ve ortalama çözüm süresinin hızla arttığı görülmüştür. Üretim sürecindeki değişkenlik arttıkça orta boyutlu örneklerde yine CPLEX ile elde edilen optimal çözüm sayısında azalma görülmüştür ancak bu oran çok sınırlı olup belirgin bir etki gözlenmemiştir.

PAT_ÇA Problemi olarak ifade edilen ikinci problemde PAT_TA Problemindeki üretim ortamı bu kez sistemde heterojen kapasiteli sınırlı sayıda aracın olduğu durum için

ele alınmıştır. Problemin basitleştirilmesi amacıyla üretim ve dağıtım partilerinin aynı olduğu varsayımı yapılmıştır. Böylece PAT_ÇA Problemi için önerilen KTP modeli doğrusal yapıda değildir. Bu nedenle öncelikle model yardımcı değişkenlerle doğrusal bir hale getirilmiş ardından modelin CPLEX ile çözümü raporlanmıştır. CPLEX için süre sınırı yine 10800 saniye olarak verilmiştir. Test problemlerinin üretilmesinde ise PAT_TA Problemleri için önerilen yaklaşım aynen kullanılmış sadece üretim ve dağıtım sürelerinin belirlenmesinde sistemde birden fazla aracın olması durumu dikkate alınarak bu süreler güncellenmiştir. Müşteri sayısına göre küçük, orta ve büyük boyutlu problemlerden oluşan test problemleri üzerinde CPLEX'in 10800 sn süre sınırı içerisinde performansı değerlendirildiğinde şu çıkarımlar ortaya çıkmıştır: CPLEX 180 küçük boyutlu örneğin 145 tanesinde verilen süre sınırı içerisinde optimal çözümlere ulaşabilmiştir. PAT_TA Probleminde küçük boyutlu örnekler için CPLEX'in çözüm süresi ortalama 15 saniye iken PAT_ÇA Probleminde bu süre ortalama 2683 saniye olarak bulunmuştur. Buradan dağıtımda sürecinde birden fazla araç olması durumunun problem karmaşıklığını büyük ölçüde artırdığı görülmektedir. CPLEX, PAT_ÇA Problemi için orta ve büyük boyutlu örneklerin hiçbirinde verilen süre sınırı içerisinde optimal çözüm elde edememiştir. Küçük, orta ve büyük boyutlu örnekler için yüzde sapma değerleri ise ortalama %7, %73 ve %95 olarak elde edilmiştir. Buradan da müşteri sayısındaki artışın problem karmaşıklığı üzerinde üstel bir etkisi olduğu çıkarımı yapılabilir. Ardından PAT_ÇA Problemi için makul sürelerde optimal ya da optimale yakın çözümler elde etmek amacıyla MA yaklaşımı önerilmiştir. Önerilen MA ile optimal bilinen 145 küçük boyutlu örneğin hepsinde ortalama 1,2 saniyede optimal çözümlere ulaşılmıştır. Orta boyutlu örneklerde yüzde sapma değerleri değerlendirildiğinde önerilen MA'ya göre CPLEX'in yüzde sapma değeri müşteri sayısındaki artışa bağlı olarak artmış ve ortalama %12 olarak elde edilmiştir. Büyük boyutlu örneklerde ise CPLEX bazı örnekler için verilen süre sınırı içerisinde bir üst sınır üretememiş ve önerilen algoritma saniyelerle ifade edilen sürelerde CPLEX'e göre büyük üstünlük sağlamıştır. PAT_ÇA Problemi için ayrıca teslim tarihlerinin etkisi ve üretim sürecindeki değişkenliğin etkisi araştırılmıştır. CPLEX'in küçük boyutlu örnekler için gevşek orta ve sıkı teslim tarihli problemlerde verilen süre sınırı içerisinde optimali elde ettiği örnek sayıları sırasıyla 57, 45 ve 43 olarak elde edilmiştir. Bunun yanı sıra küçük boyutlu problemlerde teslim tarihlerinin gevşek olduğu durumdan sıkı olduğu duruma geçerken ortalama çözüm süreleri sırasıyla 1052, 3198, ve 3798 olarak bulunmuştur. Buradan çıkarılan sonuç problemin teslim tarihlerinin sıkı olduğu durumun gevşek

olduğu duruma göre çözümünün daha zor olduğudur. Üretim sürecindeki değişkenliğin ise PAT_ÇA Problemi için yine bariz bir etkisi gözlenmemiştir.

AT_ÇA Problemi olarak ifade edilen son problemde ise üretim ortamı diğer iki problemden farklı olacak şekilde atölye tipi üretim ortamı olarak ele alınmıştır. AT_ÇA Probleminin dağıtım ortamında ise hem kapasite hem de yakıt tüketim oranı birbirinden farklı olan heterojen araçlar incelenmiştir. AT_ÇA Probleminin amaç fonksiyonu da diğer problemlerden farklı olarak çok amaçlı yapıda incelenmiştir. Birinci amaç fonksiyonu ile dağıtım aşamasında araçlar tarafından havaya salınan toplam CO₂ emisyon miktarı en küçüklenirken, ikinci amaç fonksiyonu ile müşterilerin belirlemiş oldukları teslim tarihine göre oluşan en büyük gecikme en küçüklenmektedir. AT_ÇA Problemi için öncelikle çok amaçlı KTDP modeli önerilmiş ardından probleme epsilon kısıt yönteminin geliştirilmiş versiyonu olan AEKY uygulanmıştır. Kesin yöntemle elde edilen sonuçlar değerlendirildiğinde şu çıkarımlar ortaya çıkmıştır: AEKY'nin 7200 saniye süre sınırı içerisinde performansı küçük boyutlu örnekler üzerinde değerlendirilmiş ve müşteri sayısındaki artışın problem karmaşıklığının üstel bir şekilde artırdığı görülmüştür. Sistemde 5 ve 6 müşterinin olduğu durumda AEKY ile tüm örnekler için Pareto çözümler üretilebilirken, 7 müşterinin olduğu durumda 48 örneğin 17 tanesinde AEKY ile verilen süre sınırı içerisinde ödünleşim matrisi oluşturulamamıştır. AEKY'nin problem boyutu arttıkça yetersiz gelmesi üzerine çok amaçlı optimizasyon için geliştirilen iki farklı sezgisel yöntem, BSGA-II ve PYA, AT_ÇA Problemine uygulanmıştır. PYA algoritması içerisinde, problemin üretim ortamında değişken komşu arama algoritması uygulanırken dağıtım aşamasında yerel arama yaklaşımına başvurulmuştur. AT_ÇA Problemi için rassal olarak üretilen 624 adet test problemi üzerinde önerilen algoritmaların hem kesin yöntemle hem de birbirleriyle karşılaştırılması yapılmıştır. Çok amaçlı optimizasyon performans metriği olarak PÇS, HV değerleri ve OÇS kullanılmıştır. Sonuçlar değerlendirildiğinde küçük boyutlu örneklerde hem BSGA-II hem de PYA ile elde edilen Pareto yüzeyler AEKY ile elde edilen gerçek Pareto yüzeye oldukça yakın bulunmuştur. Orta ve büyük boyutlu problemlerde ise AEKY ile verilen süre sınırı içerisinde ödünleşim matrisi oluşturulamadığından yalnızca BSGA-II ve PYA'nın birbirine göre performansı değerlendirilmiştir. Sonuçlara göre her üç performans metriği açısından da on dakikalık süre sınırı içerisinde elde edilen Pareto çözümlere göre BSGA-II, PYA'ya göre daha başarılı bulunmuştur. PYA'nın müşteri sayısındaki artışa oldukça duyarlı olduğu gözlenerek PYA 1 saat çalıştırılmıştır. Elde edilen sonuçlar incelendiğinde çözüm

süresinin artırılmasının PYA performansını ciddi oranda artırdığı hatta bazı örneklerde PYA'nın BSGA-II'yi geçtiği görülmüştür. Ayrıca AT_ÇA Problemi için teslim tarihlerinin ve araç kapasitelerinin etkisi araştırılmıştır. Teslim tarihlerinin ortak sıkı olduğu durum hem verilen süre sınırı içerisinde AEKY ile Pareto çözüm elde edilen örnek sayısı açısından hem de ortalama çözüm süresi açısından problemin en çok zorlandığı durumdur. Araç kapasitelerinin daha sıkı olduğu durumda yine AEKY'nin çözüm süresinde artış gözlenmiştir.

6.2 Tartışma

Tez kapsamında ele alınan operasyonel düzey BÜDÇ Problemleri adına ileride yapılacak olan çalışmalarda aşağıdaki hususlar dikkate alınabilir:

- (i) Tez kapsamında incelenen problemler deterministik yapıdadır ancak gerçek hayatta bazı problem parametrelerini önceden kestirmek imkansızdır. Bu nedenle BÜDÇ Problemi için stokastik modellerin de incelenmesi uygun olacaktır. Stokastik çalışmalarda belirsizlikle baş edebilecek çözüm yöntemlerinin geliştirilmesi gerekmektedir. Bu nedenle ileride yapılacak olan çalışmalarda hızlı ve gürbüz (robust) çözüm yaklaşımlarına odaklanmalıdır.
- (ii) BÜDÇ Probleminde incelenen üretim ortamı ve dağıtım ortamı daha karmaşık yapıda ele alınabilir. Üretim ortamı açısından birden fazla üretim seviyesine sahip daha karmaşık yapıdaki üretim ortamları araştırılabilir. Dağıtım ortamında ise depodaki ve müşteri lokasyonlarındaki hizmet süreleri göz önünde bulundurulabilir.
- (iii) Tez kapsamında incelenen çalışmalarda stok kararları göz önünde bulundurulmamıştır. Özellikle maliyet bazlı amaç fonksiyonlarını inceleyen BÜDÇ Problemlerinde her ne kadar stok kararları taktik seviyede alınsa da toplam maliyeti etkileyebileceğinden stok kapasitesi ve stokta tutma maliyetleri dikkate alınabilir.
- (iv) Gerçek hayatta firmalar çok sayıda müşteri ve siparişe ilgilenmek zorunda olduğundan büyük boyutlu problemler için operasyonel düzeyde etkin kararların verilmesi oldukça zor bir görevdir. Bu nedenle BÜDÇ Problemleri için kısa sürede optimale yakın çözümler verebilen farklı meta-sezgisel yaklaşımlar araştırılabilir.
- (v) Yapılan tez çalışmasında PAT_TA ve PAT_ÇA Problemleri için teslim tarihi, ve üretim sürecindeki değişkenliğin etkisi AT_ÇA Problemi içinse araç

kapasitelerinin ve teslim tarihlerinin etkisi araştırılmıştır. Bundan sonraki çalışmalarda problem varsayımlarına bağlı olarak farklı parametreler araştırılarak duyarlılık analizleri yapılabilir.



KAYNAKLAR

- Amorim, P., Belo-Filho, M., Toledo, F. d., Almeder, C., and Almada-Lobo, B., 2013, Lot sizing versus batching in the production and distribution planning of perishable goods, *International Journal of Production Economics*, 146(1), 208-218.
- Armstrong, R., Gao, S., and Lei, L., 2008, A zero-inventory production and distribution problem with a fixed customer sequence, *Annals of Operations Research*, 159(1), 395-414.
- Basseur, M., Seynhaeve, F., and Talbi, E.-g., 2002, Design of multi-objective evolutionary algorithms: Application to the flow-shop scheduling problem, *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, 1151-1156.
- Beasley, J. E., 1983, Route first—cluster second methods for vehicle routing, *Omega*, 11(4), 403-408.
- Belo-Filho, M., Amorim, P., and Almada-Lobo, B., 2015, An adaptive large neighbourhood search for the operational integrated production and distribution problem of perishable products, *International Journal of Production Research*, 53(20), 6040-6058.
- Bierwirth, C., 1995, A generalized permutation approach to job shop scheduling with genetic algorithms, *Operations-Research-Spektrum*, 17(2), 87-92.
- Bierwirth, C., Mattfeld, D. C., and Kopfer, H., 1996, On permutation representations for scheduling problems, *International Conference on Parallel Problem Solving from Nature*, 310-318.
- Chang, Y.-C., and Lee, C.-Y., 2004, Machine scheduling with job delivery coordination, *European Journal of Operational Research*, 158(2), 470-487.
- Chang, Y.-C., Li, V. C., and Chiang, C.-J., 2014, An ant colony optimization heuristic for an integrated production and distribution scheduling problem, *Engineering Optimization*, 46(4), 503-520.
- Chen, H.-K., Hsueh, C.-F., and Chang, M.-S., 2009, Production scheduling and vehicle routing with time windows for perishable food products, *Computers & operations research*, 36(7), 2311-2319.
- Chen, Z.-L., 2010, Integrated production and outbound distribution scheduling: review and extensions, *Operations research*, 58(1), 130-148.
- Chen, Z.-L., and Vairaktarakis, G. L., 2005, Integrated scheduling of production and distribution operations, *Management Science*, 51(4), 614-628.
- Cheref, A., Artigues, C., and Billaut, J.-C., 2016, Online recoverable robustness based on groups of permutable jobs for integrated production scheduling and delivery routing.
- Chiang, W.-C., Russell, R., Xu, X., and Zepeda, D., 2009, A simulation/metaheuristic approach to newspaper production and distribution supply chain problems, *International Journal of Production Economics*, 121(2), 752-767.
- Clarke, G., and Wright, J. W., 1964, Scheduling of vehicles from a central depot to a number of delivery points, *Operations research*, 12(4), 568-581.
- Condotta, A., Knust, S., Meier, D., and Shakhlevich, N. V., 2013, Tabu search and lower bounds for a combined production–transportation problem, *Computers & Operations Research*, 40(3), 886-900.

- Costa, L., Lust, T., Kramer, R., and Subramanian, A., 2018, A two-phase Pareto local search heuristic for the bi-objective pollution-routing problem, *Networks*, 72(3), 311-336.
- Croes, G. A., 1958, A method for solving traveling-salesman problems, *Operations research*, 6(6), 791-812.
- Davis, L., 1985, Applying adaptive algorithms to epistatic domains, *IJCAI*, 162-164.
- Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., 2002, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE transactions on evolutionary computation*, 6(2), 182-197.
- Devapriya, P., Ferrell, W., and Geismar, N., 2017, Integrated production and distribution scheduling with a perishable product, *European Journal of Operational Research*, 259(3), 906-916.
- Farahani, P., Grunow, M., and Günther, H.-O., 2012, Integrated production and distribution planning for perishable food products, *Flexible services and manufacturing journal*, 24(1), 28-51.
- Ferrández, M., Redondo, J., Ivorra, B., Ramos, A., and Ortigosa, P., 2017, A decision tool based on a multi-objective methodology for designing high-pressure thermal treatments in food industry, *Preprint version (September 2017)*.
- Fischer, T., and Merz, P., 2007, A memetic algorithm for the optimum communication spanning tree problem, *International Workshop on Hybrid Metaheuristics*, 170-184.
- Fu, L.-L., Aloulou, M. A., and Triki, C., 2017, Integrated production scheduling and vehicle routing problem with job splitting and delivery time windows, *International Journal of Production Research*, 55(20), 5942-5957.
- Ganji, M., Kazemipoor, H., Molana, S. M. H., and Sajadi, S. M., 2020, A green multi-objective integrated scheduling of production and distribution with heterogeneous fleet vehicle routing and time windows, *Journal of Cleaner Production*, 259, 120824.
- Garcia, J., Lozano, S., and Canca, D., 2004, Coordinated scheduling of production and delivery from multiple plants, *Robotics and Computer-Integrated Manufacturing*, 20(3), 191-198.
- Garg, P., 2010, A Comparison between Memetic algorithm and Genetic algorithm for the cryptanalysis of Simplified Data Encryption Standard algorithm, *arXiv preprint arXiv:1004.0574*.
- Geismar, H. N., Dawande, M., and Sriskandarajah, C., 2011, Pool-point distribution of zero-inventory products, *Production and Operations Management*, 20(5), 737-753.
- Geismar, H. N., Laporte, G., Lei, L., and Sriskandarajah, C., 2008, The integrated production and transportation scheduling problem for a product with a short lifespan, *INFORMS Journal on Computing*, 20(1), 21-33.
- Gharaei, A., and Jolai, F., 2018, A multi-agent approach to the integrated production scheduling and distribution problem in multi-factory supply chain, *Applied Soft Computing*, 65, 577-589.
- Golberg, D. E., 1989, Genetic algorithms in search, optimization, and machine learning, *Addison Wesley*, 1989(102), 36.
- Gupta, V., Grossmann, I. E., Pathak, S., and André, J., 2012, Assessing the benefits of production-distribution coordination in an industrial gases supply chain.
- Hajiaghaei-Keshteli, M., Aminnayeri, M., and Ghomi, S. F., 2014, Integrated scheduling of production and rail transportation, *Computers & Industrial Engineering*, 74, 240-256.

- Hall, N. G., and Potts, C. N. J. O. R., 2003, Supply chain scheduling: Batching and delivery, 51(4), 566-584.
- Hasan, S. K., Sarker, R., Essam, D., and Cornforth, D., 2009, Memetic algorithms for solving job-shop scheduling problems, *Memetic Computing*, 1(1), 69-83.
- He, P., Li, K., and Kumar, P. R., 2021, An enhanced branch-and-price algorithm for the integrated production and transportation scheduling problem, *International Journal of Production Research*, 1-16.
- Hurter, A. P., and Van Buer, M. G., 1996, The newspaper production/distribution problem, *Journal of Business Logistics*, 17(1), 85.
- Jamili, N., Ranjbar, M., and Salari, M., 2016, A bi-objective model for integrated scheduling of production and distribution in a supply chain with order release date restrictions, *Journal of Manufacturing Systems*, 40, 105-118.
- Johar, F., Nordin, S., and Potts, C., 2016, Coordination of production scheduling and vehicle routing problem with due dates, *AIP Conference Proceedings*, 030035.
- Kang, H.-Y., Pearn, W., Chung, I.-P., and Lee, A. H., 2016, An enhanced model for the integrated production and transportation problem in a multiple vehicles environment, *Soft Computing*, 20(4), 1415-1435.
- Karaoğlu, İ., and Kesen, S. E., 2017, The coordinated production and transportation scheduling problem with a time-sensitive product: a branch-and-cut algorithm, *International Journal of Production Research*, 55(2), 536-557.
- Kergosien, Y., Gendreau, M., and Billaut, J.-C., 2017, A Benders decomposition-based heuristic for a production and outbound distribution scheduling problem with strict delivery constraints, *European Journal of Operational Research*, 262(1), 287-298.
- Kesen, S. E., and Bektaş, T., 2019, Integrated production scheduling and distribution planning with time windows, In *Lean and Green Supply Chain Management*, Springer, 231-252.
- Krasnogor, N., and Smith, J., 2005, A tutorial for competent memetic algorithms: model, taxonomy, and design issues, *IEEE Transactions on Evolutionary Computation*, 9(5), 474-488.
- Lacomme, P., Moukrim, A., Quilliot, A., and Vinot, M., 2018, Supply chain optimisation with both production and transportation integration: multiple vehicles for a single perishable product, *International Journal of Production Research*, 56(12), 4313-4336.
- Lee, J., Kim, B.-I., Johnson, A. L., and Lee, K., 2014, The nuclear medicine production and delivery problem, *European Journal of Operational Research*, 236(2), 461-472.
- Li, C.-L., Vairaktarakis, G., and Lee, C.-Y., 2005, Machine scheduling with deliveries to multiple customer locations, *European Journal of Operational Research*, 164(1), 39-51.
- Li, C. L., and Ou, J. J. N. R. L., 2005, Machine scheduling with pickup and delivery, 52(7), 617-630.
- Li, K., Zhou, C., Leung, J. Y., and Ma, Y., 2016, Integrated production and delivery with single machine and multiple vehicles, *Expert Systems with Applications*, 57, 12-20.
- Li, W., and Ferrell, W. G., 2011, Integrated production and distribution schedule problem with a perishable product, *IIE Annual Conference. Proceedings*, 1.
- Li, W., and Ling, Z., 2015, A Heuristic method for the integrated production distribution problem, *IIE Annual Conference. Proceedings*, 3027.

- Liu, L., Li, W., Li, K., and Zou, X., 2020, A coordinated production and transportation scheduling problem with minimum sum of order delivery times, *Journal of Heuristics*, 26(1), 33-58.
- Low, C., Chang, C.-M., Gao, B.-Y. J. I. J. o. S. S. O., and Logistics, 2017, Integration of production scheduling and delivery in two echelon supply chain, 4(2), 122-134.
- Low, C., Chang, C.-M., Li, R.-K., and Huang, C.-L., 2014, Coordination of production scheduling and delivery problems with heterogeneous fleet, *International Journal of Production Economics*, 153, 139-148.
- Low, C., Li, R.-K., and Chang, C.-M., 2013, Integrated scheduling of production and delivery with time windows, *International Journal of Production Research*, 51(3), 897-909.
- Lust, T., and Teghem, J., 2010, Two-phase Pareto local search for the biobjective traveling salesman problem, *Journal of Heuristics*, 16(3), 475-510.
- Marandi, F., and Fatemi Ghomi, S., 2019, Integrated multi-factory production and distribution scheduling applying vehicle routing approach, *International Journal of Production Research*, 57(3), 722-748.
- Marandi, F., and Zegordi, S., 2017, Integrated production and distribution scheduling for perishable products, *Scientia Iranica*, 24(4), 2105-2118.
- Marchiori, E., and Steenbeek, A., 2000, An evolutionary algorithm for large scale set covering problems with application to airline crew scheduling, *Workshops on Real-World Applications of Evolutionary Computation*, 370-384.
- Mavrotas, G., 2009, Effective implementation of the ϵ -constraint method in multi-objective mathematical programming problems, *Applied mathematics and computation*, 213(2), 455-465.
- Merz, P., and Freisleben, B., 2001, Memetic algorithms for the traveling salesman problem, *complex Systems*, 13(4), 297-346.
- Mohammadi, S., Al-e-Hashem, S. M., and Rekik, Y., 2020, An integrated production scheduling and delivery route planning with multi-purpose machines: A case study from a furniture manufacturing company, *International Journal of Production Economics*, 219, 347-359.
- Moscato, P., 1989, On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms, *Caltech concurrent computation program, C3P Report*, 826, 1989.
- Moscato, P., and Norman, M. G., 1992, A memetic approach for the traveling salesman problem implementation of a computational ecology for combinatorial optimization on message-passing systems, *Parallel computing and transputer applications*, 1, 177-186.
- Naso, D., Surico, M., Turchiano, B., and Kaymak, U., 2007, Genetic algorithms for supply-chain scheduling: A case study in the distribution of ready-mixed concrete, *European Journal of Operational Research*, 177(3), 2069-2099.
- Park, Y.-B., and Hong, S.-C., 2009, Integrated production and distribution planning for single-period inventory products, *International Journal of Computer Integrated Manufacturing*, 22(5), 443-457.
- Potts, C. N. J. O. R., 1980, Analysis of a heuristic for one machine sequencing with release dates and delivery times, 28(6), 1436-1441.
- Prins, C., 2004, A simple and effective evolutionary algorithm for the vehicle routing problem, *Computers & operations research*, 31(12), 1985-2002.
- Prins, C., Labadi, N., and Reghioui, M., 2009, Tour splitting algorithms for vehicle routing problems, *International Journal of Production Research*, 47(2), 507-535.

- Ramezani, R., Mohammadi, S., and Cheraghalikhani, A., 2017, Toward an integrated modeling approach for production and delivery operations in flow shop system: trade-off between direct and routing delivery methods, *Journal of Manufacturing Systems*, 44, 79-92.
- Riquelme, N., Von Lücken, C., and Baran, B., 2015, Performance metrics in multi-objective optimization, *2015 Latin American Computing Conference (CLEI)*, 1-11.
- Rosenkrantz, D. J., Stearns, R. E., and Lewis, I., Philip M, 1977, An analysis of several heuristics for the traveling salesman problem, *SIAM journal on computing*, 6(3), 563-581.
- Roshanaei, V., Naderi, B., Jolai, F., and Khalili, M., 2009, A variable neighborhood search for job shop scheduling with set-up times to minimize makespan, *Future Generation Computer Systems*, 25(6), 654-661.
- Russell, R., Chiang, W.-C., and Zepeda, D., 2008, Integrating multi-product production and distribution in newspaper logistics, *Computers & Operations Research*, 35(5), 1576-1588.
- Srinivas, N., and Deb, K., 1994, Multiobjective optimization using nondominated sorting in genetic algorithms, *Evolutionary computation*, 2(3), 221-248.
- Stecke, K. E., Zhao, X. J. M., and Management, S. O., 2007, Production and transportation integration for a make-to-order manufacturing company with a commit-to-delivery business mode, 9(2), 206-224.
- Tang, J., Lim, M. H., Ong, Y. S., and Er, M. J., 2006, Parallel memetic algorithm with selective local search for large scale quadratic assignment problems, *International Journal of Innovative Computing, Information and Control*, 2(6), 1399-1416.
- Thomas, D. J., and Griffin, P. M., 1996, Coordinated supply chain management, *European journal of operational research*, 94(1), 1-15.
- Ubeda, S., Arcelus, F. J., and Faulin, J., 2011, Green logistics at Eroski: A case study, *International Journal of Production Economics*, 131(1), 44-51.
- Ullrich, C. A., 2013, Integrated machine scheduling and vehicle routing with time windows, *European Journal of Operational Research*, 227(1), 152-165.
- Van Buer, M. G., Woodruff, D. L., and Olson, R. T., 1999, Solving the medium newspaper production/distribution problem, *European Journal of Operational Research*, 115(2), 237-253.
- Viergutz, C., and Knust, S., 2014, Integrated production and distribution scheduling with lifespan constraints, *Annals of Operations Research*, 213(1), 293-318.
- Wang, J., Yao, S., Sheng, J., and Yang, H., 2019, Minimizing total carbon emissions in an integrated machine scheduling and vehicle routing problem, *Journal of Cleaner Production*, 229, 1004-1017.
- Wang, S., Wu, R., Chu, F., and Yu, J., 2019, Variable neighborhood search-based methods for integrated hybrid flow shop scheduling with distribution, *Soft Computing*, 1-20.
- Xiao, Y., Zhao, Q., Kaku, I., and Xu, Y., 2012, Development of a fuel consumption optimization model for the capacitated vehicle routing problem, *Computers & operations research*, 39(7), 1419-1431.
- Yağmur, E., and Kesen, S. E., 2020, A memetic algorithm for joint production and distribution scheduling with due dates, *Computers & Industrial Engineering*, 142, 106342.
- Yağmur, E., and Kesen, S. E., 2020, Comparison of Integrated and Sequential Decisions on Production and Distribution Activities: New Mathematical Models, *Logistics 4.0*, CRC Press, 202-225.

- Zhang, J., Wang, X., and Huang, K., 2016, Integrated on-line scheduling of order batching and delivery under B2C e-commerce, *Computers & Industrial Engineering*, 94, 280-289.
- Zhang, S., Lee, C., Choy, K., Ho, W., and Ip, W., 2014, Design and development of a hybrid artificial bee colony algorithm for the environmental vehicle routing problem, *Transportation Research Part D: Transport and Environment*, 31, 85-99.
- Zou, X., Liu, L., Li, K., and Li, W., 2018, A coordinated algorithm for integrated production scheduling and vehicle routing problem, *International Journal of Production Research*, 56(15), 5005-5024.
- Zu, L., Li, W., and Kurz, M. E., 2014, Integrated Production and distribution problem with pickup and delivery and multiple trips, *IIE Annual Conference. Proceedings*, 1587.

