



Çok amaçlı tek sıra tesis düzenleme probleminin çözümü için NSGA-II ve hedef programlama yaklaşımı

Esra Duygu Durmaz*^{ID}, Ramazan Şahin^{ID}

Gazi Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü, 06570, Ankara, Türkiye

Ö N E Ç İ K A N L A R

- Tek sıra tesis düzenleme problemi çok amaçlı olarak modellenmiştir.
- Problemin çözümü için NSGA-II ve hedef programlama yaklaşımı kullanılmıştır.
- Pareto etkin çözümler elde edilmiştir.

Makale Bilgileri

Geliş: 24.06.2016

Kabul: 05.02.2017

DOI:

10.17341/gazimmfd.337647

Anahtar Kelimeler:

Çok amaçlı tek sıra tesis düzenleme problemi, ağırlıklı hedef programlama, baskılanamayan sıralamalı genetik algoritma II

ÖZET

Tek sıra tesis düzenleme problemi, bölümlerin düz bir hat üzerinde yerleşiminin planlanması problemidir. Tek sıra tesis düzenleme probleminde, genellikle bölümler arası toplam taşıma maliyetlerini en küçüklemek hedeflenmektedir. Ancak, gerçek hayatta tesis düzenleme problemini tek bir amaçla ifade etmek mantıklı olmayacaktır. Bunun için nicel ve nitel amaçları ele alan çok amaçlı modeller üzerinde çalışılmaktadır. Bu çalışmada, toplam akış mesafesini en küçüklemeyi ve bölümler arası yakınlık skorlarını en büyüklemeyi hedefleyen çok amaçlı tek sıra tesis düzenleme problemi ele alınmıştır. Bu amaçlar, tek sıra tesis düzenleme problemi için ilk defa bu çalışmada dikkate alınmıştır. Problemin çözümü için, ağırlıklı hedef programlama yaklaşımı ve NSGA-II algoritması önerilmiştir.

NSGA-II and goal programming approach for the multi-objective single row facility layout problem

H I G H L I G H T S

- Single row facility layout problem is formulated as a multi-objective model.
- NSGA-II and goal programming approach are proposed to solve the problem.
- Pareto efficient solutions are obtained.

Article Info

Received: 24.06.2016

Accepted: 05.02.2017

DOI:

10.17341/gazimmfd.337647

Keywords:

Multi-objective single row facility layout problem, weighted goal programming, non-dominated sorting genetic algorithm II

ABSTRACT

Single row facility layout problem is the problem of arranging departments on a straight line. The main objective of the single row facility layout problem is to minimize the total material handling cost. However, it is not explanatory to use one objective. Hence, researchers pay more attention to model facility layout problems with qualitative and quantitative objectives. In this study, a weighted goal programming approach and a NSGA-II algorithm are proposed to solve multi-objective single row facility layout problem with the objectives of minimizing total flow distance and maximizing closeness rating scores. These objectives are taken into account for single row facility layout problem, for the first time, in this paper.

1. GİRİŞ (INTRODUCTION)

Tesis düzenleme problemi (TDP), bir tesis içindeki bölümlerin, üretimi (hizmeti) en iyi destekleyecek şekilde konumlarının belirlenmesini amaçlamaktadır. Bölümlerin tesis içinde doğru şekilde yerleştirilmesi, üretim maliyetleri, devam eden işler, teslim süreleri ve verimlilik açısından büyük öneme sahiptir [1]. Toplam üretim maliyetinin %20 - %50'lik bir kısmı, iş parçalarının taşınmasıyla ilgilidir ve eğer bölümler tesis içinde iyi bir şekilde yerleştirilirse, parça taşımalarında da azalmalar gerçekleşecek ve üretim maliyetlerinde %10 - %30'luk bir iyileşme sağlanabilecektir [2]. Tek sıra tesis düzenleme problemi (TSTDP) ise, belli bir doğrultudaki düz bir hat üzerinde n adet bölümün veya makinanın yerleştirilmesi olarak açıklanabilir. Her i bölümünün uzunluğu l_i olsun. c_{ij} , i ve j bölümleri arasındaki trafik (gidiş-geliş) yoğunluğu olduğu durumda, $n \times n$ boyutlu bir $C = [c_{ij}]$ matrisi verilmiş olsun. İki bölüm arasındaki uzaklık, bölümlerin ağırlık merkezlerinin birbirlerine olan uzaklıklarıyla ifade edilsin. Bu durumda problemin amacı, bütün bölüm çiftleri arasındaki uzaklıkların ağırlıklı toplamını en küçükleyecek şekilde, bir yerleşim düzeni oluşturmak olacaktır [3]. TSTDP, basitliği, verimli akış yapısı, bölümlerin farklı şekillerde düzenlenmesine uygun olması, kısa akış mesafelerine sahip olması, malzeme aktarma sistemlerinin kolay tasarlanabilmesi, üretim süreci kontrolünün etkin olması gibi pek çok nedenle, hücreli imalat sistemlerinde ve esnek imalat sistemlerinde en çok kullanılan yerleşim tiplerinden birisidir [4]. TSTDP, üretim sistemlerinde otomatik yönlendirilmiş aracın hareket ettiği yolun bir tarafına makinelerin yerleştirilmesinde [5], bir marketteki reyonların, ofis veya hastane birimlerinin koridorun bir tarafında düzenlenmesinde [6], disk silindirlerinin dosyalara atanmasında [7], kitapların bir rafa yerleştirilmesinde, depo tasarımında vb. gerçek hayatta karşılaşılan pek çok alanda uygulanabilmektedir. TSTDP, ilk olarak Simmons [6] tarafından tanıtılmış ve sonrasında pek çok araştırmacı tarafından, kesin ve sezgisel çözüm yöntemleriyle ele alınmıştır. TSTDP, NP-zor problem sınıfında yer almaktadır [8]. Problem boyutu arttıkça, kesin çözüm veren yöntemlerle optimal çözüme ulaşmak zorlaşmaktadır. Bu yüzden problemin çözümünde sıklıkla sezgisel yöntemlere başvurulmuştur.

TSTDP için Simmons [6] tarafından dal-sınır algoritmasıyla kesin çözüm aranmış, Love ve Wong [9] ise problemi doğrusal karma tamsayı programlama ile modellemişlerdir. Picard ve Queyranne [7], ilk kez dinamik programlama yaklaşımını TSTDP'nde kullanmışlardır. Heragu ve Kusiak [10] ise, doğrusal olmayan bir matematiksel modelleme yaklaşımı önermişlerdir. Amaral [11], kesme düzlemi yaklaşımıyla, Amaral ve Letchford [12] ise dal kesme algoritması yardımıyla, daha kısa çözüm sürelerinde optimal sonuçlara ulaşmışlardır. Kesin çözüm veren bu yöntemlerle, ancak 35 bölümden oluşan problemlerde optimal çözümlere ulaşılmıştır. Daha büyük problemler için, yarı belirli programlama yaklaşımı kullanan çalışmalar ile 100 bölümden oluşan problemler için alt sınır değerleri

bulunmuştur [13]. Daha büyük boyutlu TSTDP'lerinin makul sürelerde çözümü için Heragu ve Kusiak [5] tarafından bir kurucu sezgisel yöntem önerilmiştir. Kumar vd. [14] ise, tesislerin uzunluklarını göz ardı ederek, ardışık tesisler arası taşıma yoğunluğunu en büyükmeye çalışan, açgözlü bir sezgisel geliştirmişlerdir. Braglia [15], akış tipi çizelgeleme problemlerini çözmek üzere geliştirilen NEH sezgiselini temel alarak, yeni bir kurucu sezgisel algoritma geliştirmiştir. Djellab ve Gourgand [16], ekleme tabanlı iki adımlı bir kurucu sezgiselle, 30 bölümden oluşan problemlere kadar, literatürdeki problemler için bilinen en iyi çözümlere ulaşmışlardır. Literatürde, TSTDP için tavlama benzetimi [17], tabu arama algoritması [18], genetik algoritma [19], dağıtık arama [20], karınca kolonisi algoritması [21], yapay bağışıklık sistemi algoritması [22], yörlüme bağlama [23], emperyalist yarışmacı algoritma [24] gibi meta-sezgisel yöntemlerin yanı sıra melez sezgisel yöntemler de [25] önerilmiştir. TSTDP ile ilgili detaylı bilgi edinmek için Keller ve Buscher [26] tarafından yapılan literatür taraması makalesi incelenebilir. Ancak gerçek hayat problemlerinde, tesis düzenlemesini etkileyen birçok faktör vardır. Bu nedenle, TDP'lerine çözüm aranırken, birden fazla amacın dikkate alınması daha uygundur. Bu makalede, gerçek hayat problemlerine yaklaşmak ve literatürdeki bu alandaki eksikliğe katkıda bulunmak amacıyla, TSTDP çok amaçlı olarak ele alınmıştır. Problemin çözümünde, hızlı bir şekilde pareto etkin kümeyi veren NSGA-II algoritması ve ağırlıklı hedef programlama yaklaşımı kullanılmıştır. Ele alınan problemler için, bu yöntemler ile elde edilen pareto etkin çözümler karşılaştırılmıştır.

2. ÇOK AMAÇLI TEK SIRA TESİS DÜZENLEMESİ PROBLEMİ (MULTI-OBJECTIVE SINGLE ROW FACILITY LAYOUT PROBLEM)

Çok amaçlı problemler, belli kısıtlar altında birden fazla amacın bulunduğu durumlarda, çözümlerin farklı amaçlar için değerlendirilerek, hangi çözümün seçilmesi gerektiğini inceleyen problemlerdir. Çok amaçlı problemlerde, tek bir çözüm bulunabileceği gibi, bazı durumlarda amaçlar arasındaki geçişleri görebilmek ve karar vericiye birden fazla alternatif sunabilmek için, birbirini baskılayamayan çözümlerden oluşan, Pareto optimal küme adı verilen çözümler kümesi de bulunabilir. Pareto optimal kümede, bir amaç kötüleşmeden bir diğer amaçta iyileşme gözlenemez. Bir en küçükleme probleminde, Eş. 1'i sağlayan başka bir x çözümü yok ise, x^* etkin çözüm (pareto optimal çözüm) olarak tanımlanır [27].

$$\begin{aligned} f_i(x) &\leq f_i(x^*) \quad \forall i \text{ için } i = 1, 2, \dots, n \\ f_i(x) &< f_i(x^*) \quad \text{en az bir } i \text{ için} \end{aligned} \quad (1)$$

TDP çok amaçlı olarak ilk kez, Rosenblatt [28] tarafından incelenmiştir. Son yıllarda, TDP'nin çok amaçlı olarak ele alındığı çalışmalarda artış olduğu görülmektedir. Literatürde, çok amaçlı TDP'lerinde genellikle, toplam akış mesafelerinin ve malzeme taşıma maliyetlerinin en

küçüklenmesinin yanı sıra, malzeme toplam işlem zamanlarının en küçüklenmesi, alan gereksinimlerinin dikkate alınması [29], malzeme aktarma araçlarının faydalı kullanımı [30] gibi nicel amaçlar ile yakınlık ilişkisi skorları [31], ardıllık gereksinimlerinin dikkate alınması [32] gibi nitel amaçlar kullanılmıştır. Hem nicel hem de nitel amaçları dikkate alan çalışmalarda problem, gerçek hayata daha yakın bir şekilde modellenebilmekte; günümüz gelişmiş algoritma ve modelleme bilgisi sayesinde araştırmacılar, bu tarzda çalışmalara yönelmektedirler.

TSTDTP literatürde çokça çalışılan bir problem olmasına rağmen, çok amaçlı olarak ele alındığı çok az sayıda çalışma bulunmaktadır. Çok amaçlı TSTDTP ile ilgili ilk çalışma, Na ve diğerleri tarafından 2010 yılında yapılmıştır. Çok ürünli bir sistemin ele alındığı çalışmada, taşıma maliyetlerinin, işlem sürelerinin ve alan kullanım oranlarının en küçüklenmesi hedeflenmiştir [29]. Makalede 10 tezgâhlı bir atölye için uygulama çalışması yapılmıştır. Simülasyon çalışmalarıyla farklı tek sıra tesis düzenleme tiplerinin kullanılması durumunda, bahsi geçen amaç fonksiyonlarının alacağı değerler bulunmuş ve karşılaştırılmıştır. Sonuçlar farklı senaryolarda değerlendirilmiş ve en uygun düzenleme biçimi belirlenmiştir. Lenin ve diğerleri [33] tarafından yapılan çalışmada hedefler, ürünler tarafından kat edilen toplam akış mesafesini, oluşturulan çizelgede kullanılan toplam bölüm sayısını ve oluşturulan çizelgede kullanılan toplam yatırım maliyetlerini en küçüklemektir. Problem, ürünlerin sayıları, ürün talep miktarları, her ürünün hangi bölümlerde hangi sırada işlem göreceği, her bölümün maliyeti ve kullanılabilecek eş bölüm sayılarının bilindiği varsayımıyla ele alınmıştır. Çok ürünli ve ürünlerin farklı işlem sıralarına sahip olduğu bu problemde, bölüm kapasiteleri yüksek tutulmuş, parçalar için geriye dönük hareketler yasaklanmıştır. Ürünlerin farklı sıralamaları için bulunan hedef değerler normalize edilmiş, normalize edilen hedef değerlerinin ortalaması alınarak, bir ortalama uygunluk faktörü hesaplanmıştır. Problemin çözümü genetik algoritmayla gerçekleştirilmiştir. Lenin ve diğerleri [4] tarafından yapılan bir diğer çalışmada ise, bir önceki çalışmalarındaki problem aynı varsayımlar altında, yeni bir hedef daha eklenerek ele alınmıştır. Toplam malzeme aktarma maliyetini en küçüklemeyi de hedefleyen bu yeni çalışmada, tabu arama algoritması kullanılmıştır. Ortalama uygunluk faktörü, hedef değerlerinin normalize edilip ortalamalarının alınmasıyla bulunmuştur.

2.1. Matematiksel Model (Mathematical Model)

Ele alınan çok amaçlı TSTDTP'nin varsayımları şu şekildedir:

- Bölüm uzunlukları, bölümler arası parça akış miktarları ve bölümler arası yakınlık skorları bilinmektedir.
- Bölümler arası boşluklar göz ardı edilmiştir.
- Sistemde tek ürün bulunmaktadır.
- Parçalar, bölüm sıralamasında geriye doğru hareket edememektedir. Özdeş bölümlere izin verilmemiştir.

Ele alınan problemin ilk amacı, toplam akış mesafesini en küçüklemek şeklinde belirlenmiştir. Bunun için bölümler

arası parça akış miktarlarıyla, bölümler arası uzaklıklar çarpılmakta, bu işlem her bölüm çifti için gerçekleştirilmektedir. Toplam akış mesafesi, her bölüm çifti için hesaplanan akış mesafelerinin toplanmasıyla bulunmaktadır. Problemin ikinci amacı ise, toplam yakınlık skorlarını en büyükmektir. Her bölüm çiftine ait yakınlık skorları da aynı şekilde, ele alınan bölümler arasında olması gereken yakınlık değerleri ve bu bölümlerin arasındaki uzaklıklar çarpılarak hesaplanmaktadır. Toplam yakınlık değerleri de her bölüm çifti için hesaplanan yakınlık değerlerinin toplanmasıyla bulunmaktadır. Bu toplam en küçüklenerek, yakınlık skorlarının en büyüklenmesi sağlanmaktadır.

İncelenen çok amaçlı TSTDTP'nin matematiksel modelde, Love ve Wong [9] tarafından tek amaçlı TSTDTP için geliştirilen matematiksel model temel alınmıştır. Bu çalışmada ise, Love ve Wong [9] tarafından verilen model, ağırlıklı hedef programlama modeli olarak revize edilmiştir. Revize edilen modele ait parametreler, karar değişkenleri, amaç fonksiyonu ve kısıtlar aşağıda sunulmuştur.

Parametreler:

D : bölüm sayısı

f_{ij} : i ve j bölümleri arası malzeme akışı

c_{ij} : i ve j bölümleri arası yakınlık skorları

h_i : i . bölümünün uzunluğu

w_1, w_2 : 1. ve 2. amaçların ağırlıkları

$hedef_1, hedef_2$: 1. ve 2. amaçlar için hedef değerleri

M : büyük sayı

Karar değişkenleri:

$$R_{ij} = \begin{cases} i \text{ ve } j \text{ bölümlerinin merkezleri arası uzaklık, } i \text{ sağdaysa} \\ 0, \text{ diğer durumda} \end{cases}$$

$$L_{ij} = \begin{cases} i \text{ ve } j \text{ bölümlerinin merkezleri arası uzaklık, } j \text{ sağdaysa} \\ 0, \text{ diğer durumda} \end{cases}$$

x_i : i bölümünün bitiş noktası

$$\alpha_{ij} = \begin{cases} 1, & i \text{ bölümü } j \text{ bölümünün solundaysa} \\ 0, & \text{diğer durumda} \end{cases}$$

Amaç fonksiyonu:

$$\text{enk } w_1 d_1^+ + w_2 d_2^+ \quad (2)$$

Kısıtlar:

$$R_{ij} - L_{ij} = x_i - x_j + 0.5 * (h_j - h_i) \quad (3)$$

$$x_i - x_j + M * \alpha_{ij} \geq h_i \quad (4)$$

$$x_j - x_i + M * (1 - \alpha_{ij}) \geq h_j \quad (5)$$

$$h_i \leq x_i \leq \sum_{i=1}^D h_i \quad (6)$$

$$\sum_{i=1}^{D-1} \sum_{j=i+1}^D f_{ij} + d_1^- - d_1^+ = hedef_1 \quad (7)$$

$$\sum_{i=1}^{D-1} \sum_{j=i+1}^D c_{ij} + d_2^- - d_2^+ = hedef_2 \quad (8)$$

$$\alpha_{ij} \in \{0,1\} \quad \forall i, j \text{ için} \quad (9)$$

$$R_{ij}, L_{ij} \geq 0 \quad \forall i, j \text{ için} \quad (10)$$

$$x_i \geq 0 \quad \forall i \text{ için} \quad (11)$$

Geliştirilen problemde, her iki amaç da en küçüklemeyi gerektirmektedir. Ağırlıklı hedef programlamadaysa, bu amaçlardan pozitif yöndeki sapmalar (d_1^+ ve d_2^+) en küçüklenmeye çalışılmaktadır. Amaç fonksiyonu (Eş. 2), pozitif yöndeki sapmaları en küçüklemeyi hedeflemektedir. Eş. 3 ile bölümler arası uzaklıklar hesaplanmaktadır. Eş. 4 ve Eş. 5 ile gösterilen kısıtlar, i ve j bölümlerinin konumlarını sabitlemekte ve çakışmalarını engellemeye çalışmaktadır. Eş. 6, bölümlerin, 0 referans noktası ile tüm bölümlerin toplam uzunluğu içinde yer almasını sağlamaktadır. Eş. 7 ve Eş. 8, amaç fonksiyonlarının belirlenen hedef değerini en az düzeyde aşmasını sağlamak için eklenmiştir. Eş. 9, Eş. 10 ve Eş. 11 ise, karar değişkenlerinin alabileceği değerleri ifade etmektedir. Öncelikle, Love ve Wong'un, Eş. 7 ve Eş. 8 numaralı kısıtları içermeyen tek amaçlı matematiksel modeli, 2 ayrı amaç için çözülmüş ve $hedef_1$ ve $hedef_2$ değerleri elde edilmiştir. Elde edilen bu hedef değerler, ele alınan amaçlar için ulaşılmaya çalışılan değerleri olarak alınmıştır. Amaç fonksiyonlarına ait veri setleri farklı aralıklarda değişmektedir. Normalize edilmeden alınan amaçlarda, herhangi biri diğerlerine baskın gelmekte ve problemin çözümünde elde edilen sonuçlar sadece baskın olan amaca dayalı olarak (tek amaçlı bir problem gibi) oluşmaktadır. Dikkate alınan amaçların yerleşim üzerindeki etkilerinin aynı düzeyde olmasını sağlamak için, amaçların normalize edilmesine gereksinim duyulmuştur. Amaçların normalize edilmesi için alt ve üst sınır değerleri kullanılmıştır. Çalışmada bir amaç fonksiyonunun alacağı alt sınır değeri, matematiksel modelin tek amaçlı olarak çözülmesiyle elde edilmiştir. Bir amacın üst sınır değerinin belirlenmesinde ise, Malakooti [34] tarafından verilen normalize yöntemi kullanılmıştır. Bu yöntemle göre, bir amaç fonksiyonunun aldığı değer, ancak diğer amaç fonksiyonu en iyilendiğinde ortaya çıkacak değer kadar kötüleşecektir. Yalnızca 1. amaç fonksiyonunu en küçüklemeyi hedefleyen tek amaçlı matematiksel model, Model1; yalnızca 2. amaç fonksiyonunu en küçüklemeyi hedefleyen tek amaçlı matematiksel model ise, Model2 olarak adlandırılmıştır. Model1'de elde edilen en iyi sıralamanın, Model2'deki karşılığı, yani bu sıralamanın toplam yakınlık

derecelendirme değeri bulunur (f_2). Aynı şekilde, Model2'de elde edilen en iyi sıralamanın, Model1'deki karşılığı, yani bu sıralamanın toplam akış mesafesi değeri bulunur (f_1).

Tablo 1. Amaçların normalizasyonu için gerekli veriler
(Data for normalization of objectives)

	1. amaç fonksiyonu	2. amaç fonksiyonu
Model1	$hedef_1$	f_2
Model2	f_1	$hedef_2$

Tablo 1'de bulunan değerlere göre, Eş. 6 ve Eş. 7 şu şekilde güncellenmiştir (Eş. 12 ve Eş. 13);

$$\left(\sum_{i=1}^{D-1} \sum_{j=i+1}^D f_{ij} - hedef_1 \right) / (f_1 - hedef_1) + d_1^- - d_1^+ = 0 \quad (12)$$

$$\left(\sum_{i=1}^{D-1} \sum_{j=i+1}^D f_{ij} - hedef_2 \right) / (f_2 - hedef_2) + d_2^- - d_2^+ = 0 \quad (13)$$

Böylece, hedef programlama modelinde, normalize edilmiş amaç fonksiyonlarından gerçekleşen pozitif yönde sapmalar en küçüklenmeye çalışılmıştır. Sonuçlar elde edildikten sonra normalize edilmiş amaç fonksiyonu değerleri tekrar eski haline çevrilmiştir.

3. NSGA-II ALGORİTMASI (NSGA-II ALGORITHM)

NSGA-II algoritması, Deb ve diğerleri tarafından [35] ortaya atılan, çok amaçlı bir evrimsel algoritmadır. Algoritma, Srinivas ve Deb [36]'ın geliştirdiği NSGA algoritmasının eksikliklerinin giderilmesiyle oluşturulmuştur. NSGA-II, Pareto-en iyi kümeyi araştırmak için, genetik algoritma tabanlı olarak tasarlanmıştır. Genetik algoritmanın adımlarına ek olarak, baskınlık derecelendirmesi ve yığılma uzaklığı hesaplaması işlemleri de uygulanmaktadır. NSGA-II, düşük hesaplama karmaşıklığına sahip, hızlı ve seçkinliği dikkate alan bir algoritma olduğu için, literatürde pek çok alanda uygulaması bulunmaktadır. İşçi atama problemi [37], proje yönetimi alanında karar alternatiflerinin elde edilmesi [38], tedarik zinciri dağıtım problemi [39], çok dönemli stok kontrol problemi [40], kablosuz algılayıcı ağlarda uygun yerleşimin belirlenmesi [41] gibi pek çok problemde Pareto etkin kümenin bulunmasında NSGA-II algoritmasından yararlanılmış, algoritma sonucunda elde edilen Pareto etkin küme üzerindeki domine edilemeyen noktalar, karar vericilere çözüm alternatifleri olarak sunulmuştur. Ancak, TDP'lerinde, NSGA-II algoritmasının kullanıldığı herhangi bir çalışmaya rastlanılmamıştır. NSGA-II algoritması, hızlı baskılanamayan sıralama yaklaşımı, yığılma uzaklığı ataması ve bir ana döngüden oluşmaktadır.

3.1. Hızlı Baskılanamayan Sıralama Yaklaşımı (Fast Nondominated Sorting Approach)

Bu aşamada, bireyler, baskın olma derecelerine göre, farklı baskınlık kümelerinde sınıflandırılır.

N : yığın büyüklüğü

n_p : baskınlık sayısı

S_p : p bireyinin baskın olduğu bireylerin kümesi olmak üzere;

her $p \in P$ için

$$S_p = \{ \}$$

$n_p = 0$

her $q \in P$ için

eğer $(p < q)$ ise

$$S_p = S_p \cup \{q\}$$

eğer $(q < p)$ ise

$$n_p = n_p + 1$$

eğer $n_p = 0$ ise

$$p_{derece} = 1$$

$$F_1 = F_1 \cup \{p\}$$

$i = 1$

$F_i \neq \{ \}$ iken

$$Q = \{ \}$$

her $p \in F_i$ için

her $q \in S_p$ için

$$n_q = n_q - 1$$

eğer $n_q = 0$ ise

$$q_{derece} = i + 1$$

$$Q = Q \cup \{q\}$$

$i = i + 1$

$$F_i = Q$$

Yığındaki bireyler her amaç için kıyaslanarak, bireyin baskın olduğu bireylerin sayısı ve bireye baskın olan bireyler hafızada tutulur. Eğer bireye baskın olan bireylerin sayısı 0 ise, o birey 1. baskılanamayan bireyler kümesine eklenir. $n_p = 0$ olan her p bireyine ait S_p kümesindeki her q bireyi ziyaret edilir ve n_p değeri 1 azaltılır. Bu işlemler gerçekleştirilirken, q bireyine baskınlık sayısı (n_p) 0 olursa, q bireyi Q kümesine eklenir. Q kümesindeki bireyler, 2. baskılanamayan bireyler kümesini oluşturur. Yukarıda anlatılan süreç, Q kümesinin tüm elemanlar için uygulanarak, bir sonraki baskılanamayan bireyler kümesi oluşturulur. Her küme birbirlerine baskın olmayan bireylerden oluşmaktadır. Bu işlemler tüm bireyler kümelerine atanıp, tüm kümeler belirleninceye kadar devam ettirilir.

3.2. Yığılma Uzaklığı Ataması (Crowding Distance Assignment)

Her baskılanamayan küme, kendi içinde yığılma uzaklığına göre sıralanır. Bu sıralama, belli bir noktanın (çözümün) etrafındaki çözümlerin yoğunluğuna göre belirlenir. Her amaç fonksiyonu için, sınır çözümlere (en küçük ve en büyük değere sahip çözümlere) sonsuz uzaklık değeri atanır. Böylece, uç değerlerin seçilmesi garantilenmiş olur. Arada kalan tüm diğer çözümler için, tüm amaç fonksiyonları için bir uzaklık değeri hesaplanır. Toplam yığılma uzaklığı değeri, her amaç için hesaplanan bireysel uzaklıkların toplamıdır.

$$L = |I|$$

Her i için

$$I[i]_{uzaklık} = 0$$

Her m amacı için

$$I = \text{siralala}(I, m)$$

$$I[1]_{uzaklık} = I[L]_{uzaklık} = \infty$$

$i = 2$ 'den $(L-1)$ 'e kadar her i için

$$I[i]_{uzaklık} = I[i]_{uzaklık} + (I[i + 1].m - I[i - 1].m) / (f_m^{max} - f_m^{min})$$

Yığındaki i ve j bireyleri yoğunluk karşılaştırma operatörü (α_n) ile karşılaştırılır ve farklı baskılanamayan sıralardaki iki çözümden, daha düşük (daha iyi) olanı tercih edilir. Eğer iki çözüm de aynı baskılanamayan kümedeyse, yığılma uzaklığı daha yüksek olan, başka bir deyişle çevresindeki çözüm kalabalığı daha düşük olan çözüm tercih edilir.

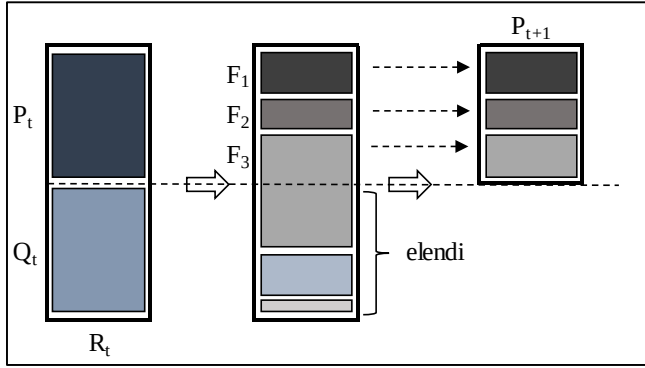
$i \alpha_n j$ eğer ($i_{derece} < j_{derece}$) ise veya ($(i_{derece} = j_{derece})$ ve ($i_{uzaklık} > j_{uzaklık}$))

3.3. Ana Döngü (Main Loop)

Önerilen algoritmada, çözüm gösterimi olarak permütasyon tipi gösterim seçilmiştir. Yığını oluşturan her bir birey, bölümlerin bir sıralamasını ifade etmektedir. Bireyin her elemanı, yani kromozomun her geni, bir bölüme karşılık gelmektedir. P_0 başlangıç yığını rasgele biçimde oluşturulur. Yapılan denemeler sonucunda yığın büyüklüğü ($N = 5 \times \text{bölüm sayısı}$) olarak belirlenmiştir. Yığın, baskılanamama derecelerine göre sıralanır. Her çözüme, baskılanamama derecesine göre (1, en iyi seviye olmak üzere) bir derece atanır. Genetik algoritmalarda bireylerin kalitesi, uygunluk değerlendirme fonksiyonlarıyla belirlenmektedir ve uygunluk değerlendirme fonksiyonu, genellikle amaç fonksiyonu olarak seçilmektedir. Bu çalışmada toplam akış mesafesi ve toplam yakınlık skorları olmak üzere iki uygunluk değerlendirme fonksiyonu bulunmaktadır ve bireylerin birbirlerine olan baskınlıkları bu uygunluk değerlendirme fonksiyonlarına göre belirlenmiştir. Genetik algoritmadaki gibi, yeniden üretim operatörü, çaprazlama ve mutasyon operatörleri kullanılarak, N büyüklüğünde (Q_0) çocuk yığını üretilir. Burada kullanılan genetik operatörler bir sonraki bölümde anlatılacaktır. NSGA-II'de seçkinlik ve seçim, mevcut yığının bir önceki en iyi baskın olmayan çözümlerle karşılaştırılmasıyla yapıldığı için, prosedür ilk nesilden sonra tek amaçlı genetik algoritmadan farklılaşır.

R_t , ebeveyn ve çocuk yığınların birleşiminden oluşan, $2 \times N$ büyüklüğündeki bir yığını ifade etmektedir ve bu yığın baskılanamama derecelerine göre sıralanır. F_1 , en iyi baskılanamayanlar kümesini ifade etmektedir. Eğer $F_1 < N$ ise, F_1 'deki tüm çözümler, bir sonraki ebeveyn yığını (P_{t+1}) oluşturmak üzere seçilir. P_{t+1} 'in kalan elemanları, baskılanamama dereceleri sırasına göre, bir sonraki sıradaki baskın olmayan kümeden seçilir. Bu prosedür, başka küme kalmayana dek devam eder. F_1 'den F_L 'ye kadar tüm kümelerdeki çözümlerin toplam sayısı, yığın büyüklüğünden fazla olacaktır. Yeni oluşturulan yığına N adet birey seçmek için, son baskılanamama derecesindeki çözümler, yığılma uzaklığına göre büyüktür küçüğe sıralanır ve yığındaki boşlukları dolduracak şekilde, en yüksek yığılma uzaklığına sahip çözümden başlayarak, yığına dahil edilir. Yeni oluşturulan N büyüklüğündeki P_{t+1} yığınının seçim, çaprazlama ve mutasyon işlemleri uygulanarak, N büyüklüğünde bir Q_{t+1} yığını oluşturulur. Daha sonra P_{t+1} ebeveyn yığınıyla birleştirilerek $2 \times N$ büyüklüğünde bir R_{t+1} yığını oluşturulur ve yukarıda anlatılan işlemler uygulanır.

Bu işlem, belirlenen sayıda nesil oluşturuluncaya kadar devam ettirilir. NSGA-II algoritması ile ilgili yukarıda yapılan açıklamalar, Deb ve diğerleri [35] tarafından Şekil 1'deki gibi gösterilmiştir.



Şekil 1. NSGA-II prosedürü (NSGA-II Procedure)

NSGA-II'de ana döngüye ait uygulama adımları aşağıda verilmiştir.

t. nesil için

$$R_t = P_t \cup Q_t$$

F = hızlı baskın olmayan sıralama (R_t)

$$P_{t+1} = \{ \} \text{ ve } i = 1$$

$|P_{t+1}| + |F_i| \leq N$ olana kadar yığılma uzaklığı ataması (F_i)

$$P_{t+1} = P_{t+1} \cup F_i$$

$$i = i + 1$$

Sırala (F_i , a_n)

$$P_{t+1} = P_{t+1} \cup F_i[1 : (N - |P_{t+1}|)]$$

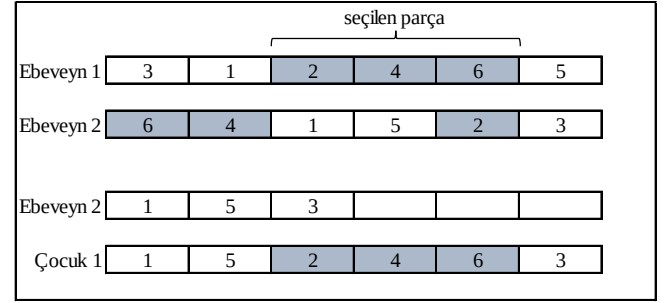
Q_{t+1} = yeni yığın oluştur (P_{t+1})

$$t = t + 1$$

3.4. Genetik Operatörler (Genetic Operators)

Yeniden üretim operatörleri, kalitesi daha yüksek olan bireyin, genetik operatörlerde işlem görecektir. Daha şansının daha yüksek olması prensibine göre çalışır. Daha kaliteli ebeveyn birey seçimi, gelecek nesillerin de daha kaliteli olmasını sağlayacaktır. NSGA-II algoritmasında eşleşme havuzuna alınacak bireylerin seçim işlemi, yalnızca ilk iterasyonda klasik yeniden üretim operatörleri kullanılarak yapılmaktadır. İlerleyen iterasyonlarda seçim işlemi, NSGA-II algoritmasının kendi mekanizmalarıyla gerçekleştirilmektedir. Bu çalışmada, ilk iterasyonda kullanılmak üzere ikili turnuva seçimi yöntemi tercih edilmiştir. İkili turnuva seçiminde, yığından rastgele olarak seçilen iki birey, uygunluk değerlendirme fonksiyonu açısından karşılaştırılmakta ve daha iyi olan birey genetik operatörlerde işlem görmek üzere seçilmektedir. Bu işlem, eşleşme havuzunda yığın büyüklüğü sayısı kadar birey olana kadar devam ettirilmektedir. Bireylerin kalitesi değerlendirilirken, her iki amaç fonksiyonu da dikkate alınmış ve uygunluk değeri olarak %50 olasılıkla bir amaç fonksiyonunun rastgele seçilmesi sağlanmıştır. Yeniden üretim operatörüyle ve NSGA-II algoritmasının seçim mekanizmasıyla oluşturulan eşleşme havuzuna, belirli oranda çaprazlama işlemi uygulanmıştır. Bu oran yapılan

denemeler sonucunda 0,6 olarak belirlenmiştir. Çaprazlama yöntemi olarak, sıralı çaprazlama yöntemi kullanılmıştır (Şekil 2).



Şekil 2. Sıralı Çaprazlama (Order Crossover)

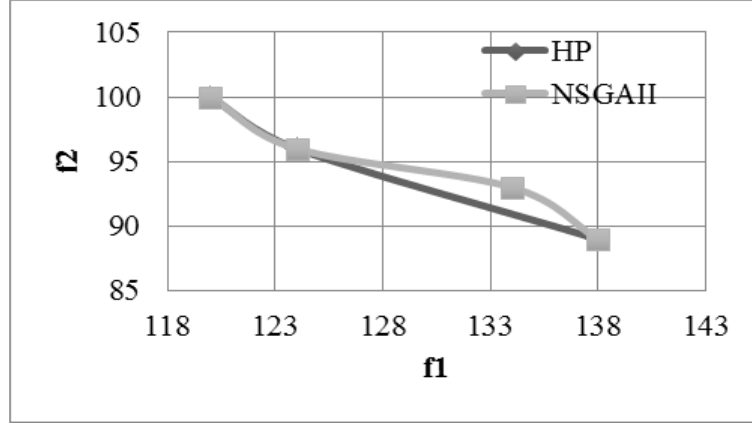
Mutasyon operatörü, yığından seçilen bireylerde, küçük değişiklikler yapılmasını sağlar ve işlem yalnızca seçilen bireyi etkiler. Sıralama çözüm gösterimine sahip problemlerde genellikle, ikili yer değiştirme, araya ekleme veya tersine çevirme yöntemleriyle mutasyon işlemi gerçekleştirilmektedir. Bu çalışmada, araya ekleme mutasyon yöntemi kullanılmıştır. Bu yöntemde, bireyde rastgele seçilen bir konuma rastgele seçilen bir bölüm yerleştirilmektedir. Yığındaki her birey, önceden belirlenen bir mutasyon oranında, mutasyon işlemine tabi tutulur. Mutasyon oranının, gerekenden yüksek belirlenmesi, genetik algoritmada aramanın olasılıklı aramaya yakın şekilde gerçekleşeceği anlamına geldiği için, bu oran küçük tutulmalıdır. Mutasyon oranı, denemeler sonucunda 0,05 olarak seçilmiştir.

4. SONUÇLAR VE TARTIŞMALAR (RESULTS AND DISCUSSIONS)

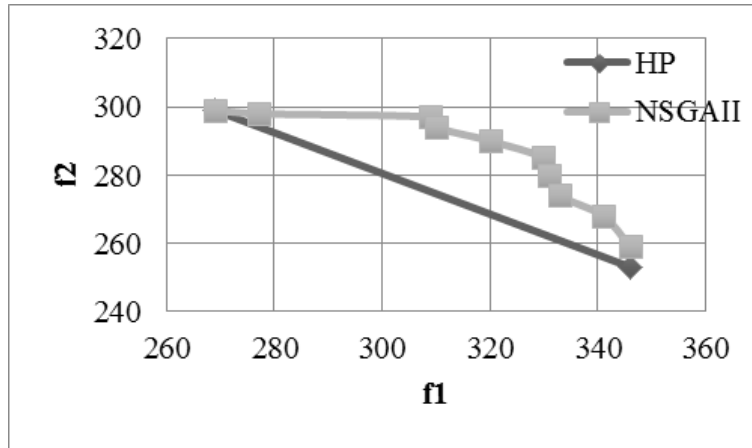
TSTDP'lerinde, daha önce toplam akış mesafesini ve toplam yakınlık skorlarını en küçükleme amaçları birlikte ele alınmadığı için, veri seti olarak farklı tesis düzenleme problemlerinin verilerinden yararlanılmıştır. Veri setlerinden 8 bölüme kadar olan problemler için, matematiksel modellerle en iyi çözümler bulunmuştur. Ancak daha büyük problemler için, hedef programlama modelleri LINGO 9.0 programında 6 saat çalıştırılmış, elde edilen sonuçlar, NSGA-II sonuçlarıyla karşılaştırılmıştır. Ağırlıklı hedef programlama yaklaşımı kullanılarak, farklı ağırlıklar için amaç fonksiyonu değerleri ile pareto etkin küme oluşturulmuştur. Tablo 2'de, Dutta ve Sahu [42] tarafından verilmiş olan 6 bölümlü bir tesis için hedef programlama yaklaşımı ve NSGA-II algoritması ile elde edilen çözüm değerleri ve Şekil 3'te bu çözüm değerlerine ait Pareto diyagramları verilmiştir. Yine Dutta ve Sahu [42]'dan alınan verilerle, Tablo 3'te ve Şekil 4'te 8 tesis için elde edilen sonuçlar yer almaktadır. Tablo 4, Şekil 5, Tablo 5 ve Şekil 6'da, Urban [43] tarafından verilmiş olan iki farklı 6 bölümlü tesis problemi için elde edilen çözümler ve pareto diyagramları sunulmuştur. Fortenberry ve Cox [44] tarafından verilmiş olan 12 bölümlü tesis düzenleme problemi için ulaşılan sonuçlar Tablo 6 ve Şekil 7'de sunulmuştur.

Tablo 2. 6 bölüm için elde edilen çözümler (Obtained solutions for 6 department)

Hedef Programlama		NSGA-II	
f1	f2	f1	f2
120	100	120	100
124	96	124	96
138	89	134	93
		138	89

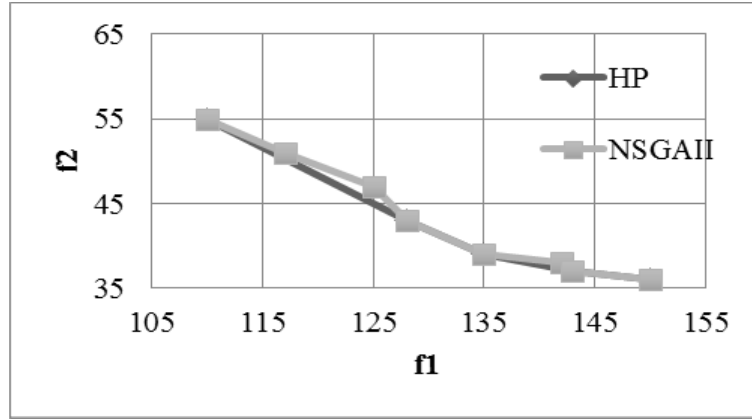
**Şekil 3.** 6 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 6 department)**Tablo 3.** 8 bölüm için elde edilen çözümler (Obtained solutions for 8 department)

Hedef Programlama		NSGA-II	
f1	f2	f1	f2
269	299	269	299
346	253	277	298
		309	297
		310	294
		320	290
		330	285
		331	280
		333	274
		341	268
		346	259

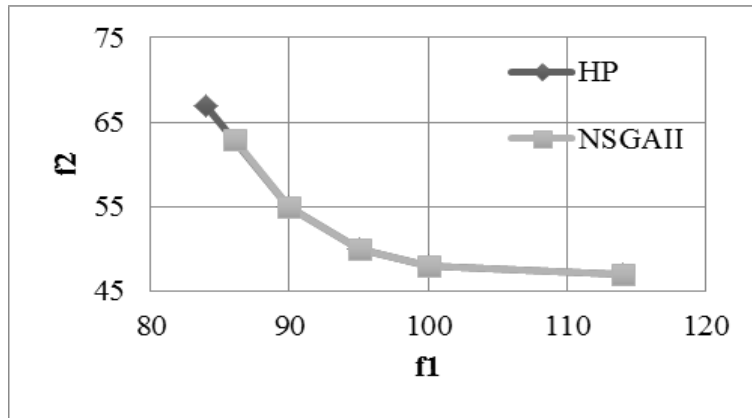
**Şekil 4.** 8 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 8 department)

Tablo 4. 6 bölüm için elde edilen çözümler (Obtained solutions for 6 department)

Hedef Programlama		NSGA-II	
f1	f2	f1	f2
110	55	110	55
128	43	117	51
135	39	125	47
143	37	128	43
150	36	135	39
		142	38
		143	37
		150	36

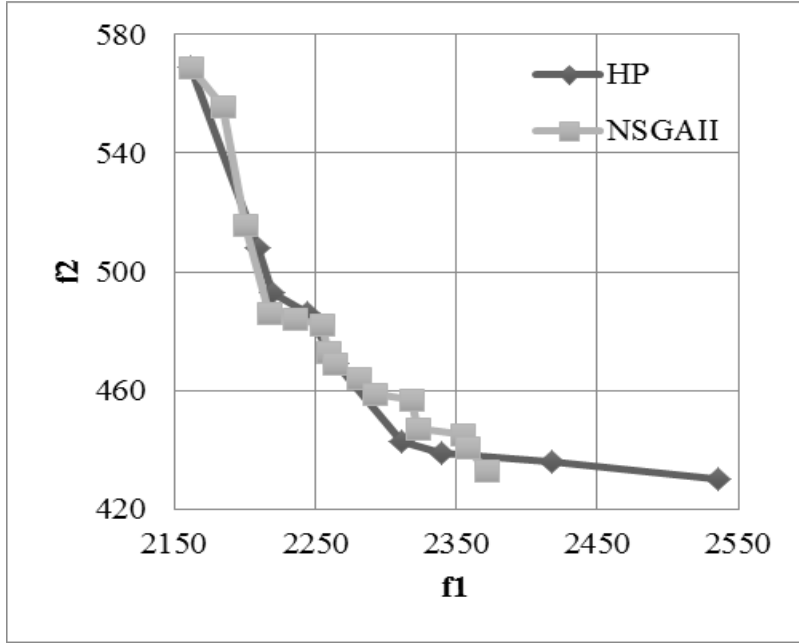
**Şekil 5.** 6 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 6 department)**Tablo 5.** 6 bölüm için elde edilen çözümler (Obtained solutions for 6 department)

Hedef Programlama		NSGA-II	
f1	f2	f1	f2
84	67	86	63
90	55	90	55
95	50	95	50
100	48	100	48
114	47	114	47

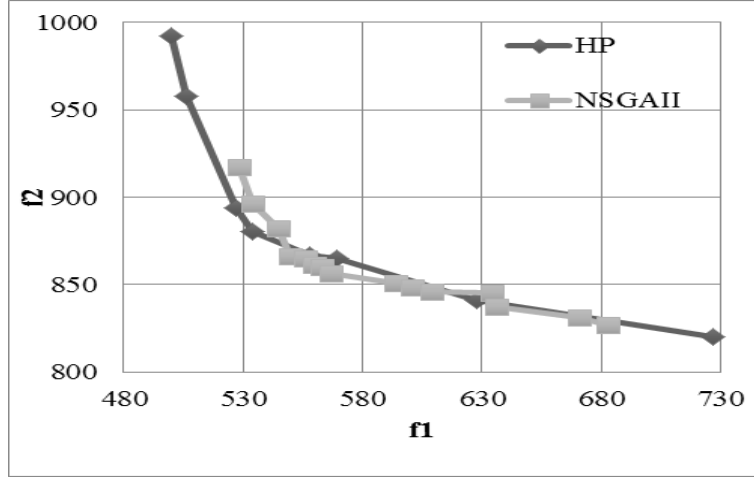
**Şekil 6.** 6 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 6 department)

Tablo 6. 12 bölüm için elde edilen çözümler (Obtained solutions for 12 department)

Hedef Programlama		NSGA-II	
f1	f2	f1	f2
2162	569	2162	569
2209	508	2184	556
2219	493	2200	516
2244	486	2217	486
2265	469	2235	484
2311	443	2254	482
2339	439	2259	473
2418	436	2264	469
2536	430	2281	464
		2292	459
		2318	457
		2323	447
		2354	445
		2358	441
		2371	433

**Şekil 7.** 12 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 12 department)**Tablo 7.** 12 bölüm için elde edilen çözümler (Obtained solutions for 12 department)

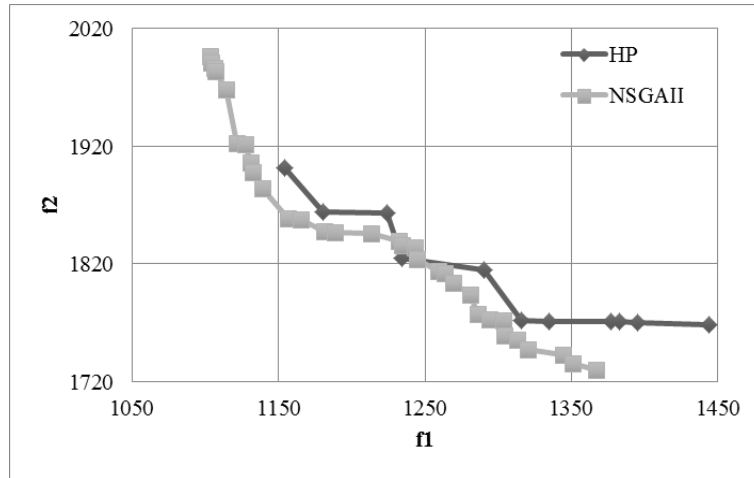
Hedef Programlama		NSGA-II			
f1	f2	f1	f2	f1	f2
500	992	528	917	594	851
506	958	534	896	601	848
527	894	545	882	609	846
534	880	550	866	634	845
558	867	556	865	636	837
569	865	560	861	656	832
628	841	563	860	671	831
727	820	567	856	683	827



Şekil 8. 12 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 12 department)

Tablo 8. 15 bölüm için elde edilen çözümler (Obtained solutions for 15 department)

Hedef Programlama		NSGA-II					
f1	f2	f1	f2	f1	f2	f1	f2
1154	1902	1103	1997	1165	1858	1281	1794
1180	1864	1104	1991	1181	1848	1286	1777
1224	1863	1106	1987	1188	1847	1294	1773
1234	1825	1107	1984	1213	1846	1303	1772
1290	1815	1114	1968	1232	1839	1304	1759
1316	1772	1121	1923	1234	1836	1313	1755
1335	1771	1127	1922	1243	1834	1320	1747
1377	1771	1131	1906	1244	1824	1344	1743
1383	1771	1132	1898	1259	1814	1351	1735
1395	1770	1139	1884	1263	1812	1367	1730
1444	1768	1156	1859	1269	1804		



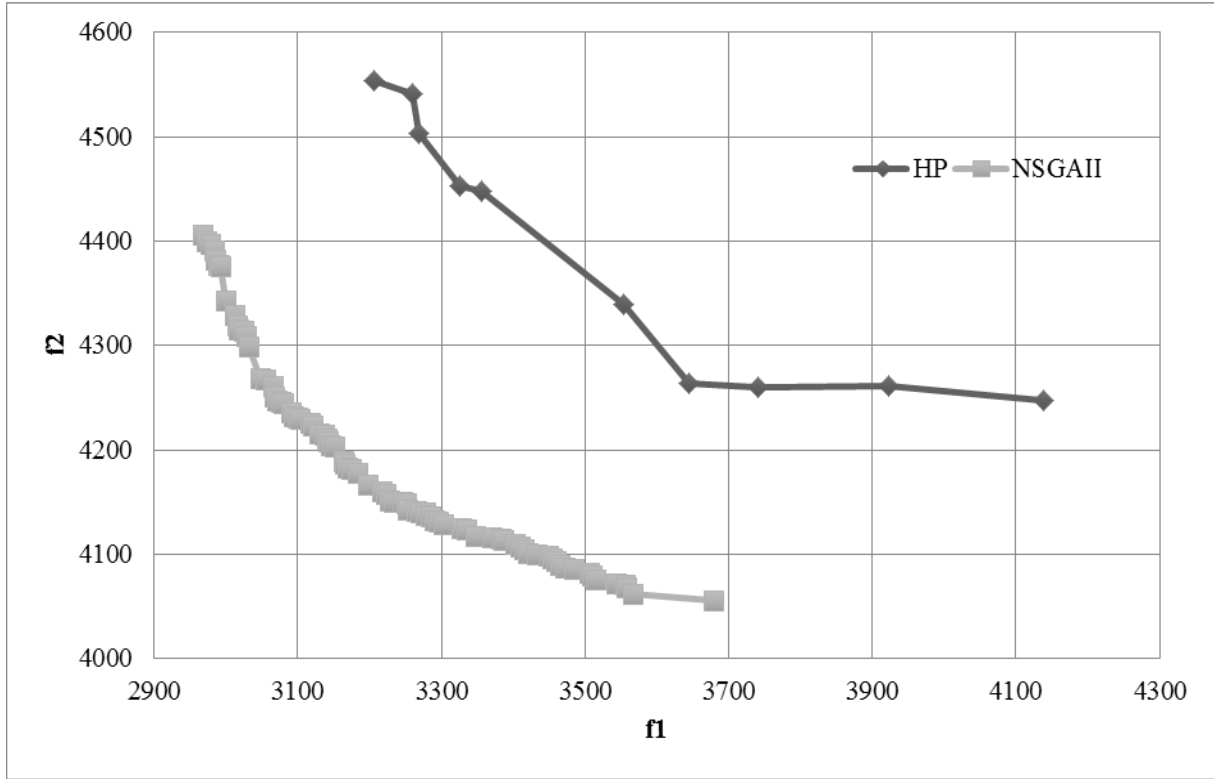
Şekil 9. 15 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 15 department)

Suresh ve Sahu [45] tarafından literatürde sunulmuş olan problemlere ilişkin sonuçlar Tablo 7-10 ve Şekil 8-11'de verilmiştir. Tablo 7 ve Şekil 8 12 bölümlü tesis düzenleme problemine ait sonuçları; Tablo 8 ve Şekil 9 15 bölümlü tesis düzenleme problemine ait sonuçları; Tablo 9 ve Şekil 10 20 bölümlü tesis düzenleme problemine ait sonuçları; Tablo 10 ve Şekil 11 30 bölümlü tesis düzenleme problemine ait

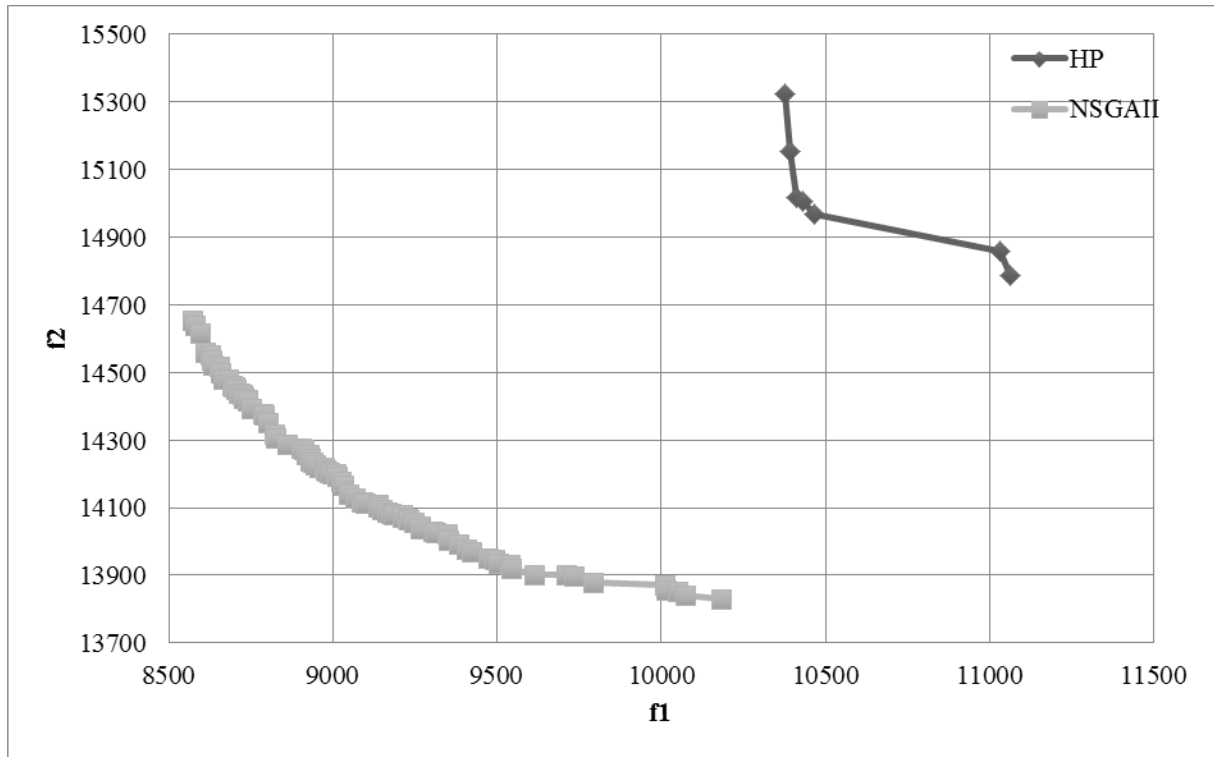
sonuçları içermektedir. Ağırlıklı hedef programlama yönteminin belirli süre çalıştırılmasıyla, büyük boyutlu problemlerde en iyi çözüme ulaşamadığı için, pareto optimal eğrisinden uzakta sonuçlar elde edildiği, aynı problemler için NSGA-II yöntemiyle, çok daha kısa sürede daha etkin bir pareto kümesi ve daha fazla sayıda çözüm bulunmuştur.

Tablo 9. 20 bölüm için elde edilen çözümler (Obtained solutions for 20 department)

Hedef Programlama		NSGA-II			
f1	f2	f1	f2	f1	f2
3206	4554	2968	4406	3188	4172
3259	4541	2973	4400	3191	4168
3269	4504	2978	4398	3197	4167
3325	4453	2983	4391	3217	4161
3356	4448	2986	4382	3222	4158
3553	4339	2989	4378	3227	4152
3644	4264	2990	4377	3233	4151
3741	4260	2993	4376	3246	4150
3923	4261	2999	4343	3250	4149
4138	4247	3012	4329	3252	4143
		3015	4319	3265	4142
		3018	4315	3271	4141
		3024	4314	3277	4140
		3028	4309	3278	4138
		3031	4299	3284	4137
		3034	4295	3290	4133
		3040	4294	3296	4132
		3042	4291	3303	4129
		3045	4273	3328	4125
		3048	4269	3334	4124
		3054	4268	3347	4118
		3065	4261	3370	4116
		3068	4251	3382	4115
		3071	4247	3387	4114
		3077	4246	3402	4110
		3080	4245	3409	4107
		3091	4236	3415	4105
		3094	4232	3420	4101
		3100	4231	3433	4100
		3103	4230	3449	4099
		3117	4226	3454	4096
		3120	4224	3459	4094
		3130	4216	3464	4090
		3136	4215	3472	4087
		3140	4211	3485	4086
		3142	4209	3506	4082
		3145	4205	3510	4080
		3151	4204	3515	4076
		3163	4189	3543	4072
		3165	4187	3556	4071
		3168	4183	3557	4068
		3174	4182	3567	4062
		3183	4178	3678	4056
		3186	4174		



Şekil 10. 20 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 20 department)



Şekil 11. 30 bölüm için Pareto diyagramı karşılaştırması (Pareto diagrams comparison for 30 department)

Tablo 10. 30 bölüm için elde edilen çözümler (Obtained solutions for 30 department)

Hedef Programlama		NSGA-II			
f1	f2	f1	f2	f1	f2
10377	15323	8570	14655	9010	14199
10393	15155	8580	14638	9011	14192
10413	15018	8594	14615	9022	14179
10430	15006	8612	14561	9031	14168
10464	14970	8627	14551	9049	14141
11030	14858	8629	14537	9067	14131
11063	14785	8636	14521	9068	14129
		8653	14519	9088	14119
		8658	14498	9092	14117
		8665	14481	9097	14113
		8682	14480	9136	14111
		8692	14463	9137	14103
		8698	14457	9150	14095
		8699	14456	9166	14087
		8703	14451	9175	14082
		8712	14440	9183	14079
		8717	14438	9209	14078
		8724	14435	9210	14073
		8728	14429	9225	14069
		8729	14424	9230	14063
		8738	14421	9245	14057
		8741	14416	9266	14044
		8751	14393	9267	14038
		8767	14387	9312	14028
		8786	14377	9347	14023
		8790	14376	9351	14004
		8800	14353	9366	14001
		8823	14316	9381	13994
		8827	14306	9406	13978
		8860	14287	9423	13970
		8904	14276	9473	13951
		8912	14271	9493	13947
		8919	14259	9504	13934
		8927	14258	9539	13932
		8929	14245	9541	13922
		8930	14239	9612	13902
		8940	14235	9708	13901
		8945	14227	9733	13899
		8956	14220	9792	13879
		8972	14218	10008	13870
		8977	14213	10013	13856
		8982	14207	10050	13851
		8988	14206	10074	13841
		8998	14202	10182	13831

5. SONUÇLAR (CONCLUSIONS)

Tesis düzenlemesi, işletmelerin üretim maliyetlerini etkileyen en önemli unsurlardan birisidir. Tek sıra tesis düzenlemesi ise, esnek üretim sistemlerinde ve hücreli imalat sistemlerinde en çok kullanılan, tesis düzenleme yöntemlerinden birisidir ve literatürde bu problemi inceleyen pek çok çalışma mevcuttur. Literatürde TSTD P'ni çok amaçlı olarak ele alan, çok az sayıda çalışma vardır. Bu çalışmada, gerçek hayat problemlerine yaklaşmak ve bu alandaki eksikliğe katkıda bulunmak amacıyla, TSTD P çok amaçlı olarak ele alınmıştır. Problemin çözümü için NSGA-II algoritması ve ağırlıklı hedef programlama yaklaşımı önerilmiş, her iki yöntemle elde edilen pareto etkin çözümler karşılaştırılmıştır. Karşılaştırma sonucunda, NSGA-II algoritmasının, ağırlıklı hedef programlamaya göre çok daha kısa hesaplama sürelerinde, daha iyi sonuçlar verdiği görülmüştür. Küçük boyutlu problemlerde, her iki yöntemin de birbirine daha yakın sonuçlar verdiği gözlemlenirken, problem boyutu attıkça NSGA-II algoritmasının daha iyi pareto etkin çözümler bulduğu görülmektedir. Aynı zamanda, NSGA-II algoritması ile daha fazla sayıda pareto etkin çözüm elde edilmiş ve karar vericinin çözümler içinden, ele alınan amaçlara verdiği önem derecesine göre seçim yapmasına imkân tanınmıştır.

KAYNAKLAR (REFERENCES)

- Drira A., Pierreval H., Hajri-Gabouj S., Facility layout problems: A survey, *Annu. Rev. Control*, 31 (2), 255-267, 2007.
- Tompkins J.A., White J.A., Bozer Y.A., Tanchoco J.M., *Facilities Planning*, John Wiley & Sons, ABD, 2010.
- Amaral A.R.S., On the exact solution of a facility layout problem, *Eur. J. Oper. Res.*, 173 (2), 508-518, 2006.
- Lenin N., Kumar M.S., Ravindran D., Islam M.N., A tabu search for multiobjective single row facility layout problem, *Journal of Advanced Manufacturing Systems*, 13 (1), 17-40, 2014.
- Heragu S.S., Kusiak A., Machine layout problem in flexible manufacturing systems, *Oper. Res.*, 36 (2), 258-268, 1988.
- Simmons D.M., One-dimensional space allocation: An ordering algorithm, *Oper. Res.*, 17 (5), 812-826, 1969.
- Picard J.C., Queyranne M., On the one-dimensional space allocation problem, *Oper. Res.*, 29 (2), 371-391, 1981.
- Beghin-Picavet M., Hansen P., Deux problèmes d'affectation nonlineaires, *RAIRO, Recherche Operationnelle*, 16 (3), 263-276, 1982.
- Love R.F., Wong J.Y., On solving a one-dimensional space allocation problem with integer programming, *INFOR*, 14 (2), 139-143, 1976.
- Heragu S.S., Kusiak A., Efficient models for the facility layout problem, *Eur. J. Oper. Res.*, 53 (1), 1-13, 1991.
- Amaral A.R.S., A new lower bound for the single row facility layout problem, *Discrete Appl. Math.*, 157 (1), 183-190, 2009.
- Amaral A.R.S., Letchford A.N., A polyhedral approach to the single row facility layout problem, *Math. Program.*, 141 (1-2), 453-477, 2013.
- Hungerlander P., Rendl F., A computational study and survey of methods for the single-row facility layout problem, *Computational Optimization and Applications*, 55, 1-20, 2013.
- Kumar K.R., Hadjinicola G.C., Lin T.L., A heuristic for the single-row facility layout problem, *Eur. J. Oper. Res.*, 87 (1), 65-73, 1995.
- Braglia M., Heuristics for single-row layout problems in flexible manufacturing systems, *Production Planning & Control*, 8, 558-567, 1997.
- Djellab H., Gourgand M., A new heuristic procedure for the single-row facility layout problem, *Int. J. Computer Integr. Manuf.*, 14 (3), 270-280, 2001.
- Palubeckis G., Fast simulated annealing for single-row equidistant facility layout, *Appl. Math. Comput.*, 263, 287-301, 2015.
- Kothari R., Ghosh D., Tabu search for the single row facility layout problem using exhaustive 2-opt and insertion neighborhoods, *Eur. J. Oper. Res.*, 224, 93-100, 2013.
- Kothari R., Ghosh D., An efficient genetic algorithm for single row facility layout, *Optimization Letters*, 6 (2), 679-690, 2014.
- Kothari R., Ghosh D., A scatter search algorithm for the single row facility layout problem, *Journal of Heuristics*, 20 (2), 125-142, 2014.
- Solimanpur M., Vrat P., Shankar R., An ant algorithm for the single row layout problem in flexible manufacturing systems, *Computers & Operations Research*, 32 (3), 583-598, 2005.
- Kumar R.M.S., Asokan P., Kumanan S., An artificial immune system-based algorithm to solve linear and loop layout problems in flexible manufacturing systems, *International Journal of Product Development*, 10, 165-179, 2010.
- Kothari R., Ghosh D., Path relinking for single row facility layout, Ahmedabad, Indian Institute of Management, 2012.
- Kunlei L., Chaoyong Z., Liang G., Xinyu S., Single row facility layout problem using an imperialist competitive algorithm, 41. International Conference on Computers & Industrial Engineering, 578-586, 2011.
- Ou-Yang C., Utamima A., Hybrid estimation of distribution algorithm for solving single row facility layout problem, *Comput. Ind. Eng.*, 66, (1), 95-103, 2013.
- Keller B., Buscher U., Single row layout models, *Eur. J. Oper. Res.*, 245 (3), 629-644, 2015.
- Şahin R., Türkbey O., A new hybrid heuristic algorithm for the multi objective facility layout problem, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 25 (1), 119-130, 2010.
- Rosenblatt M.J., The facilities layout problem: a multi-goal approach, *Int. J. Prod. Res.*, 17 (4), 323-332, 1979.
- Na Z., Kelin X., Wei Q., Multi-objective combinatorial optimization of multi-product single-row layout,

- Electrical and Control Engineering (ICECE), 2010 International Conference on. IEEE, 1324-1327, 2010.
30. Ertay T., Ruan D., Tuzkaya U.R., Integrating data envelopment analysis and analytic hierarchy for the facility layout design in manufacturing systems, *Information Sciences*, 176, 237-262, 2006.
31. Chen G.Y.H., Lo J.C., Dynamic facility layout with multi-objectives, *Asia-Pacific Journal of Operational Research*, 31 (4), 2014.
32. Emami S., Nookabadi A.S., Managing a new multi-objective model for the dynamic facility layout problem, *Int. J. Adv. Manuf. Technol.*, 68 (9), 2215-2228, 2013.
33. Lenin N., Kumar M.S., Islam M., Ravindran D., Multi-objective optimization in single-row layout design using a genetic algorithm, *Int. J. Adv. Manuf. Technol.*, 67 (5), 1777-1790, 2013.
34. Malakooti B., *Operations and Production Systems with Multiple Objectives*, John Wiley & Sons, A.B.D., 2014.
35. Deb K., Pratap A., Agarwal S., Meyarivan T., A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Trans. Evol. Comput.*, 6 (2), 182-197, 2002.
36. Srinivas N., Deb K., Multiobjective optimization using nondominated sorting in genetic algorithms, *Evol. Comput.*, 2 (3), 221-248, 1995.
37. Xu Z., Ming X.G., Zheng M., Li M., He L., Song W., Cross-trained workers scheduling for field service using improved NSGA-II, *Int. J. Prod. Res.*, 53 (4), 1255-1272, 2015.
38. Fallah-Mehdipour E., Haddad O.B., Tabari M.M.R., Marino M.A., Extraction of decision alternatives in construction management projects: Application and adaptation of NSGA-II and MOPSO, *Expert Syst. Appl.*, 39, 2794-2803, 2012.
39. Chan F.T.S., Jha A., Tiwari M.K., Bi-objective optimization of three echelon supply chain involving truck selection and loading using NSGA-II with heuristics algorithm, *Appl. Soft Comput.*, 38, 978-987, 2016.
40. Mousavi S.M., Sadeghi J., Niaki S.T.A., A bi-objective inventory optimization model under inflation and discount using tuned Pareto-based algorithms: NSGA-II, NRGA, and MOPSO, *Appl. Soft Comput.*, 43, 57-72, 2016.
41. Okay F.Y., Özdemir S., Improving Coverage in Wireless Sensor Networks Using Multiobjective Evolutionary Algorithms, *Journal of the Faculty of Engineering and Architecture of Gazi University*, 30 (2), 143-153, 2015.
42. Dutta K.N., Sahu S., A multigoal heuristic for facilities design problems: MUGHAL, *Int. J. Prod. Res.*, 20 (2), 147-154, 1982.
43. Urban T.L., A multiple criteria model for the facilities layout problem, *Int. J. Prod. Res.*, 25 (12), 1805-1812, 1987.
44. Fortenberry J.C., Cox J.F., Multiple criteria approach to the facilities layout problem, *Int. J. Prod. Res.*, 23 (4), 773-782, 1985.
45. Suresh G., Sahu S., Multiobjective facility layout using simulated annealing, *Int. J. Prod. Econ.*, 32 (2), 239-254, 1993.