

Karma modellenli tip-2 montaj hattı dengeleme problemi için bir kısıt programlama modeli

A constraint programming model for mixed model type 2 assembly line balancing problem

Hacı Mehmet ALAĞAŞI^{1*}, Mehmet PINARBAŞI², Mustafa YÜZÜKIRMIZI³, Bilal TOKLU⁴

¹Endüstri Mühendisliği Bölümü, Mühendislik Fakültesi, Kırıkkale Üniversitesi, Kırıkkale, Türkiye.
hmalagas@kku.edu.tr

²Endüstri Mühendisliği Bölümü, Mühendislik Fakültesi, Hitit Üniversitesi, Çorum, Türkiye.
mehmetpinarbasi@hitit.edu.tr

³İşletme Bölümü, İktisadi ve İdari Bilimler Fakültesi, Melikşah Üniversitesi, Kayseri, Türkiye.
myuzukirmizi@hotmail.com

⁴Endüstri Mühendisliği Bölümü, Mühendislik Fakültesi, Gazi Üniversitesi, Ankara, Türkiye.
btoklu@gazi.edu.tr

Geliş Tarihi/Received: 24.08.2015, Kabul Tarihi/Accepted: 27.01.2016

* Yazılan yazar/Corresponding author

doi: 10.5505/pajes.2016.93276

Araştırma Makalesi/Research Article

Öz

Bu çalışmada karma modellenli montaj hattı dengeleme problemleri için yeni bir kısıt programlama modeli sunulmuştur. Önerilen model verilen bir istasyon sayısı ile çevrim zamanını en küçüklemektedir. Önerilen model literatürdeki örnek problemler ile test edilmiştir ve modelin performansı karma modellenli montaj hattı dengeleme problemlerinin matematiksel modeli ile karşılaştırılmalı olarak değerlendirilmiştir. Performans kriterleri olarak ulaşılan en iyi çözüm değeri ve CPU süresi kullanılmıştır. Deneysel sonuçlar önerilen kısıt programlama modelinin problemin çözümünde iyi performans gösteren bir alternatif modelleme tekniği olduğunu göstermiştir.

Anahtar kelimeler: Karma modellenli montaj hattı, Kısıt programlama, Tip 2 problemi.

Abstract

This paper presents a new constraint programming model for mixed-model assembly line balancing problem. The proposed model minimizes the cycle time for a given number of stations. The proposed model is tested with literature problems and its performance is evaluated by comparing to mathematical model. Best obtained solution and elapsed CPU time are used as performance criteria. The experimental results show that the proposed constraint programming model performs well and can be used as an alternative modeling technique to solve the problem.

Keywords: Mixed-model assembly line, Constraint programming, type 2 problem

1 Giriş

Montaj hattı, son ürünü elde etmek için belirlenen görevlerle ürünün alt parçalarının montajının yapıldığı bir üretim prosesidir. Bir montaj hattı sistemi, birbirine konveyör veya benzer bir malzeme taşıma sistemi ile bağlı olan sıralı iş istasyonlarından oluşmaktadır. Görevler istasyonlara aralarındaki öncelik ilişkisi dikkate alınarak atanmaktadır ve her bir görev belirlenen görev süresi içinde yapılmaktadır. Her bir istasyona atanan görevler çevrim süresi içinde tekrarlı olarak yapılır. Üretim yapılacak olan ürün istasyonları sıra ile ziyaret edip tüm montaj işlemleri yapıldıktan sonra bitmiş ürün olarak son istasyonda hattın ayrılır. Görevlerin belirlenen kısıtlar altında bir veya daha fazla amaca göre istasyonlara optimum atamalarının yapılması problemi montaj hattı dengeleme problemi olarak isimlendirilmektedir [1].

Montaj hatlarının üretimde ilk kullanılmaya başlandığı dönemlerde hat üzerinde tek bir ürünün üretilmesinden dolayı hattın dengelenmesi daha kolaydı. Üretim sistemlerinin genelinde olduğu gibi montaj hatları da değişen ve farklılaşan müşteri taleplerinden etkilenmiştir. Çeşitlenen müşteri taleplerinden dolayı tek tip üretim yapılan montaj hatları için talep yetersiz kalmıştır. Bunun sonucunda farklı modellerin aynı hat üzerinde üretilmesi zorunlu hale gelmiştir. Ancak, bu beraberinde her bir model için görev sürelerinin farklı olabilmesi, bazı görevlerin bazı modellerde yapılmasına gerek

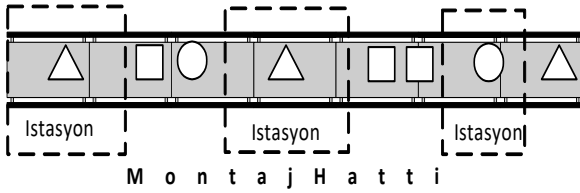
duyulmaması, model taleplerinin zamanla değişkenlik göstermesi gibi durumları ortaya çıkarmıştır. Bu oluşan yeni durumlar dolayısıyla tek modellenli montaj hattına göre karma modellenli montaj hattının dengelenmesi daha da zorlaşmıştır.

Karma modellenli montaj hatları farklı modellerin aynı hat üzerinde üretildiği hatlardır (Şekil 1). Karma modellenli montaj hattı dengeleme problemleri (KMMHDP) şekillerine, amaçlarına, görev sürelerinin değişkenliğine, vb. özelliklerine göre sınıflara ayrılmaktadır. Literatürde özellikle amaçlarına göre sınıflandırma ön plana çıkmaktadır. Montaj hattı dengeleme problemleri amaçlarına göre dört ana kategoride toplanmaktadır. Tip-1 olarak tanımlanan problem tipinde sabit çevrim süresinde istasyon sayısı minimizasyonu, buna karşılık olarak tip-2 problemlerinde ise sabit istasyon sayısında çevrim süresi minimizasyonu amaçlanmıştır. Montaj prosesindeki veya ürün yelpazesindeki değişiklikler montaj hattının yeniden dizayn edilmesini gerektirir. Bu tip yeniden dengeleme problemleri tip-2 problemi olarak ele alınıp çözülebilir. Tip-E problemlerinde ise belirlenen bir etkinlik fonksiyonunun maksimizasyonu veya minimizasyonu amaçlanmaktadır. Tip-F'de ise uygun bir atamanın bulunması hedeflenmektedir.

Bu çalışmada tip-2 karma modellenli montaj hattı dengeleme problemleri için yeni bir kısıt programlama modeli geliştirilmiştir. Kısıt programlama birçok kombinatoryal problemin çözümünde kullanılan alternatif bir modelleme tekniğidir. Ancak yazarlar tarafından yapılan literatür

taramasına göre kısıt programlamanın karma modellenin montaj hattı dengeleme problemi için bir uygulamasına rastlanmamıştır.

Çalışmanın planı şu şekildedir: Birinci bölümde verilen giriş bölümünün ardından, ikinci bölümde KMMHDP için literatür araştırması verilmiştir. KMMHDP hakkında genel bilgi ve matematiksel model bölüm üçte verilmiştir. Kullanılan çözüm yöntemi olan kısıt programlama yönteminden dördüncü bölümde bahsedilmiştir. Beşinci bölümde problem için önerilen kısıt programlama modeli verilmiştir ve örnek bir problem ile açıklanmıştır. Deneysel çalışmalar ve sonuçları ile birlikte problem bileşenlerinin analizi altıncı bölümde verilmiştir. Çalışma yedinci bölümde sonuçlardan ve gelecekte yapılabilecek çalışmalardan bahsedilerek tamamlanmıştır.



Şekil 1: Karma modellenin montaj hattı.

2 Literatür araştırması

Montaj hattı dengeleme konusunda çok sayıda çalışma bulunmaktadır. Konunun çok geniş olmasından dolayı literatür incelemesi seri, karma modellenin montaj hattı dengeleme problemleri ile sınırlandırılarak özet olarak Şekil 2'de verilmiştir.

Karma modellenin montaj hattı ile ilgili ilk çözüm önerisini Arcus [2] COMSOAL metodunu önerdiği çalışmasında ele almış ve talep yüzdeleri kullanarak belirlediği ağırlıklı çevrim zamanını kullanarak atamaları yapmıştır. Bu çalışma tek bir amacı dikkate almıştır. Bu çalışmada olduğu gibi araştırmacılar istasyon sayısının minimizasyonunu amaçlamışlar ve geliştirdikleri sezgiseller ile problemi çözmüşlerdir [3]-[6]. Bu çalışmaların ortak özelliği ise amaç olarak sadece istasyon sayısının minimizasyonunu almalarıdır. Ayrıca, Thomopoulos [4] çalışmasında farklı modellerin öncelik ilişkilerinin birleştirilmesini ilk defa önermiştir. Bu çalışmalara ek olarak, Gökçen ve Erel [7] tek amaç değerini dikkate almışlar ve ilk defa KMMHDP için 0-1 tam sayılı programlama modeli önermişlerdir. Gökçen ve Erel [8] bir diğer çalışmalarında benzer olarak istasyon sayısının minimizasyonunu ele almışlar, ancak model bazında farklı çevrim süresi değerlerinin olduğunu kabul etmişlerdir.

Karma modellenin hatlar için, araştırmacılar amaç fonksiyonunda istasyon sayısının veya çevrim süresinin minimizasyonuna ek olarak; istasyon bazında modeller arasındaki süre farklılıkları, her bir model için fark eden hattın boş zamanının, düzgünlük indeksinin veya istasyon kurulumu, işçilik gibi maliyetlerin minimizasyonu değerlerini de dikkate alarak atamalar yapmışlardır [9]-[18]. Sawik [19], farklı bakış açısı olarak problemi bir akış tipi çizelgeleme olarak ele alıp taleplerin maksimum tamamlanma zamanını minimize etmiştir. Ayrıca, Cevikcan ve diğerleri [20] ise takım odaklı bir montaj hattı kurarak takımların boş zamanlarını minimize eden bir çalışma yapmışlardır.

Görev sürelerinin stokastik varsayıldığı montaj hattı tiplerinde ise amaç fonksiyonları her ne kadar istasyon sayısının veya

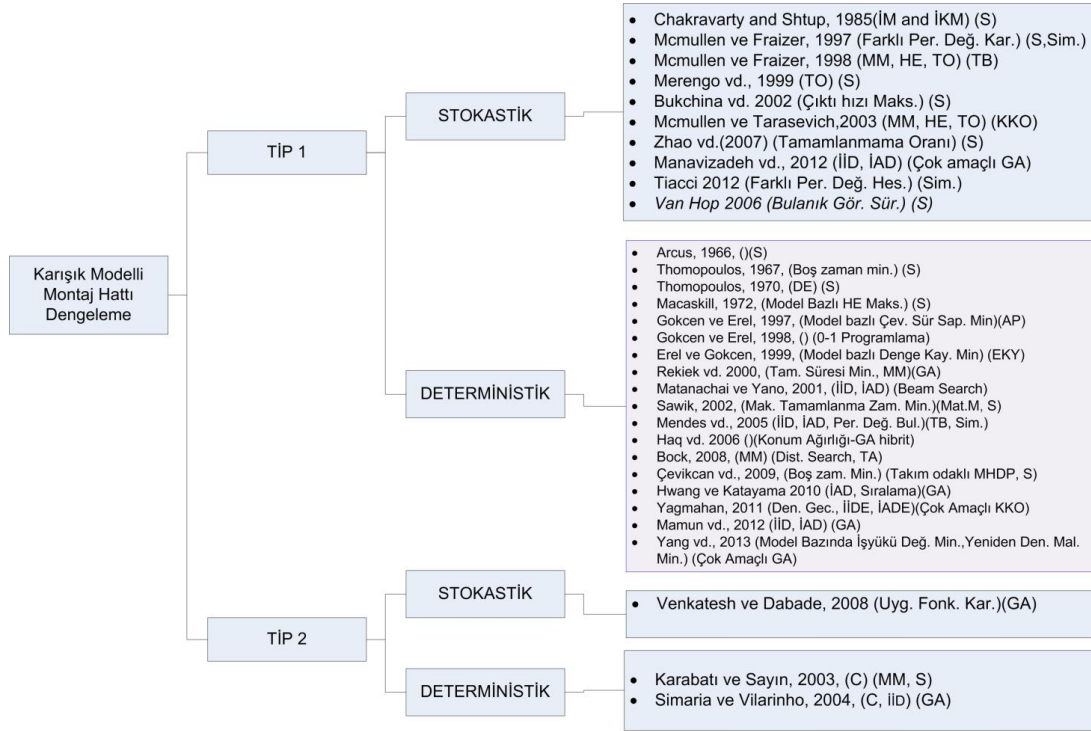
çevrim süresinin minimizasyonu gibi ana amaçları içerse de bunun yanında deterministik hatlarda olduğu gibi farklı değerlerin de minimizasyonunu içermektedir. Ancak bu tip hatlarda sürelerin değişkenliğinden kaynaklı olarak deterministik görev süreli hatlara göre kullanılan amaç değerleri de farklılaşmaktadır. Bunlar; üretim, işçilik, envanter ve kurulum maliyetleri gibi maliyet fonksiyonları [21]-[23], hat etkinliğinin maksimizasyonu [22],[23], belirlenen süre içinde tamamlanan işlerin maksimizasyonu veya tamamlanamayan işlerin minimizasyonu [22]-[25], işlem gören ürün sayısı, çıktı hızı, sistem akış zamanı ve kullanım oranı gibi sistemin performans değerleri [26]-[28]. Bunlara ek olarak, Manavizadeh ve diğerleri [29] çalışmalarında model bazında istasyon içi ve istasyonlar arası dengeyi dikkate alan çok amaçlı bir genetik algoritma ile problemi çözmüşlerdir. Ayrıca, Venkatesh ve Dabade [30] çalışmalarında tip-2 modelini ele almışlar ve genetik algoritmada farklı uygunluk fonksiyonları kullanarak en iyi uygunluk fonksiyonunu bulmayı amaçlamışlardır.

Stokastik ve deterministik görev sürelerinden farklı olarak, Hop [31] görev süreleri bulanık kabul edip tip-1 KMMHDP'yi geliştirdiği sezgisel yöntemle çözmüştür.

Amaçlarına göre incelediğimiz çalışmaların çözüm yöntemlerini değerlendirdiğimizde ise genel olarak sezgisel yöntemlerin kullanıldığı görülmektedir. Şekil 2'deki özet gösterim incelendiğinde hem deterministik hem de stokastik görev süreli çalışmaların neredeyse tamamında sezgisel yöntemler kullanılmıştır. Sezgisel yöntemlerin kullanımının en önemli sebebi problemin zorluğundan dolayı kesin sonuç veren algoritmaların büyük problemlerin çözümünde yetersiz kalmasıdır. Kesin sonuç veren yöntem olarak Gökçen ve Erel [8] çalışması söylenebilir. Bu çalışmada tip-1 modeli için 0-1 tam sayılı programlama modelini önermişler ve çözmüşlerdir.

Sezgisel çalışmalar içinde meta sezgisel kullanılan çalışmalara bakıldığında ise genetik algoritma kullanılan çalışmaların daha fazla olduğu görülmektedir [6],[10],[12],[15],[17],[18],[29],[30]. Bunun yanında tavlama benzetimi [22],[13], karınca kolonisi optimizasyonu [23],[16], tabu arama [14] ve demet arama (Beam search) [9] algoritmaları kullanılmıştır. Ayrıca simülasyon kullanan çalışmalar olsa da, araştırmacılar simülasyonu bir yöntemle buldukları uygun atamaların performansını ölçmek için kullanmışlardır [13],[26],[28]. Simülasyon modelinin kurulmasının uzun sürmesi ve çalışma süresinin fazla olmasından ve genel olarak hat dengeleme problemlerinde de çok fazla alternatif atama üretilecek olmasından dolayı çok kullanışlı olmamıştır. Bunun dışında Gökçen ve Erel [7] çalışmalarında amaç programlama ve Erel ve Gökçen [32] çalışmalarında ise en kısa yol algoritmalarını kullanmışlardır. Bu çalışmalar dışında kalan ve Şekil 2'de verilen çalışmalarda ise araştırmacılar kendi geliştirdikleri sezgisel yöntemleri kullanarak problemleri çözmüşlerdir.

Kısıt programlama literatürde birçok kombinatoriyal problemin çözümünde kullanılan bir tekniktir [33]. Kısıt programlamanın uygulamalarını çizelgeleme probleminde Terekhov ve diğ. [34], tedarik zinciri tasarımında Li ve Womer [35], araç sıralama problemlerinde Siala ve diğ. [36] olarak vermek mümkündür. Kısıt programlamanın montaj hattı dengeleme problemi için uygulaması literatürde oldukça azdır. Bockmayr ve Pisaruk [37] montaj hattı dengeleme problemini kısıt programlama ile ilk ele alan çalışmadır. Çalışmada dal kesme ve kısıt programlamaya dayalı bir hibrit çözüm yöntemi sunulmuştur.



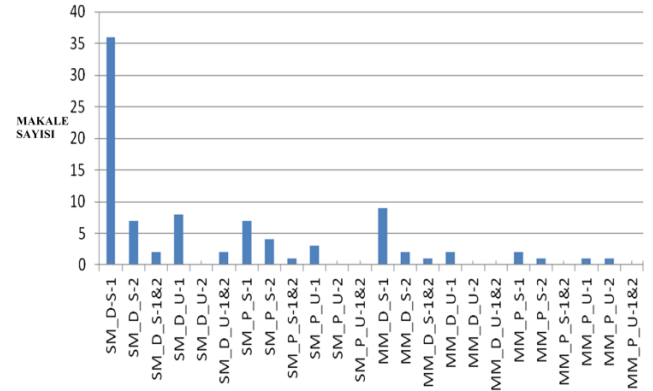
C: Çevrim süresi, İM: İşçilik Maliyeti, İKM: Station Setup Cost, S: Sezgisel, Sim: Simulasyon,MM: Maliyet Minimizasyonu, HE: Hat Etkinliği, TO: Tamamlanma Oranı, TB: Tavlama Benzetimi, KKO: Karınca Kolonisi Optimizasyonu, İİD: İstasyon İçi Denge, İAD: İstasyonlar Arası Denge, GA: Genetik Algoritma, DE: Düzgünlük Endeksi, AP: Amaç Programlama, EKY: En Kısa Yol, Mat.M: Matematiksel Model, TA: Tabu Arama, İİDE: İstasyon İçi Düzgünlük Endeksi, İADE: İstasyonlar Arası Düzgünlük Endeksi

Şekil 2: Düz karma modelli montaj hattı dengeleme problemleri literatür özeti.

Pastor ve diğ. [38] tip-1 ve tip-2 montaj hattı dengeleme problemlerinin matematiksel modellerine alternatif ilk kısıt programlama modellerini önermişlerdir. Topaloğlu ve diğ. [33] çalışmalarında, alternatif öncelik ilişkilerine sahip montaj hattı için kural tabanlı bir model önermişlerdir. Problemi tam sayılı programlama ve kısıt programlama kullanarak çözmüşlerdir. Sonuç olarak kısıt programlamanın çözüme ulaşmada daha etkin olduğunu göstermişlerdir. Alağaç ve diğ. [39] çalışmalarında stokastik işlem süreli tip-2 MHDP için bir model önermişlerdir. Modelde görevlerin istasyonlara atanmasında kısıt programlama kullanmışlardır. Bulunan atama kombinasyonları için kapalı kuyruk ağı modelini kullanarak çıktı hızı değerlerini bulmuşlardır. En büyük çıktı değerine sahip olan atamayı optimum atama olarak belirlemişlerdir. Öztürk ve diğ. [40] karma modelli montaj hattını ele aldıkları çalışmalarında montaj hattı problemini bir çizelgeleme problemi gibi ele alıp, bütün ürünlerin üretiminin tamamlanması için gerekli bütün görevlerin bitiş zamanını minimize etmeyi amaçlamışlardır. Problemden görev ataması, görev sıralaması ve ürün sıralamasını yapan bir metodoloji önermişlerdir. Çözüm için MIP ve CP birlikte kullanmışlardır.

Son dönemde Sivasankaran ve Shahabudeen [41] yaptıkları literatür taramasında montaj hattı problem tiplerine göre yapılan çalışmaları bir grafik ile özetlemişlerdir. Şekil 3'te de görüleceği üzere tip-2 (grafikte x ekseninde verilen ifadelerde sonunda 2 ile belirtilen problem tipleri) problemleri ile ilgili yapılan çalışmalar azdır. Buna ek olarak karma modelli montaj hattı çalışmaları (grafikte MM ifadesi karma modelli çalışmaları gösterir) da az sayıdadır. Bu çalışmada tip-2 karma modelli montaj hattı dengeleme problemi için bir kısıt programlama modeli geliştirilmiştir. Çalışmada bulunan sonuçlar

matematiksel modellerle karşılaştırılmış ve sonuçlara ilişkin tartışmalar verilmiştir.



Şekil 3: Tip-1 ve Tip-2 montaj hattı problem tiplerine göre yapılan çalışmaların sayısı [41].

3 Karma modelli montaj hattı dengeleme problemi

Bu çalışmada ele alınan problem, tip-2 karma modelli montaj hattı dengeleme problemidir (KMMHDP-2). Benzer M tane modelin aynı hat üzerinde montajı yapılmaktadır. Modeller kendine özgü öncelik diyagramına sahiptirler ancak bu diyagramlar birleştirilip N tane görevin olduğu tek bir öncelik diyagramı oluşturulabilmektedir. Her bir görevin süresi bellidir ancak her bir model için farklılık gösterebilmektedir. Eğer bir model için görevin süresi sıfır ise bu model için bu görev yapılmaktadır. Problemin montaj hattı literatüründeki yeri

Şekil 4'te verilmiştir. Probleme ilişkin varsayım ve kabuller aşağıdaki gibidir:

Görev zamanları bilinmektedir ve sabittir.

İstasyon sayısı sabittir ve bilinmektedir.

Bir görev birden fazla istasyona atanamaz.

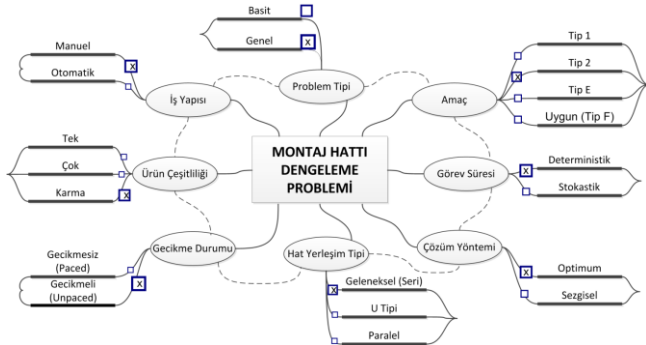
Bütün görevler bir istasyona atanmak zorundadır.

Görevler bölünemezler.

Öncelik diyagramları belirlidir ve bilinmektedirler.

Bir görev ancak kendisinden önceki görevler tamamlandıktan sonra başlayabilir.

Hat üzerinde birden fazla model karma olarak üretilmektedir.



Şekil 4: Problem tanımı.

Bu varsayım ve kabullere göre problemin notasyonları ve ilk defa Gokcen ve Erel [8] tarafından önerilen 0-1 tam sayılı matematiksel modeli aşağıdaki gibidir.

Notasyonlar:

i, j	Görev indeksi,
k	İstasyon indeksi,
m	Model indeksi,
N	Birleştirilmiş öncelik diyagramındaki görevlerin sayısı,
M	Hat üzerinde montaj edilen modellerin sayısı,
S	İş istasyonu sayısıdır,
C	Hattın çevrim süresidir,
t_{im}	i görevinin m modeli için işlem süresidir,
$(i, j) \in Pr$	Öncelik ilişkisi kısıdı, i görevi mutlaka j görevinden önce yapılmalıdır,
x_{ik}	=1 eğer i görevi k istasyonuna atanmış ise, =0 diğer durumlarda.

Amaç:

$$\text{Min} \sum_{m=1}^M C_m \quad (1)$$

Kısıtlar:

$$\sum_{k=1}^S X_{ik} = 1 \quad i = 1, \dots, N \quad (2)$$

$$\sum_{k=1}^S k * x_{ik} - \sum_{k=1}^S k * x_{jk} \leq 0 \quad (i, j) \in Pr \quad (3)$$

$$\sum_{i=1}^N t_{im} * x_{ik} \leq C_m \quad k = 1, \dots, S; m = 1, \dots, M \quad (4)$$

$$x_{ik} \in \{0,1\} \quad i = 1, \dots, N; k = 1, \dots, S \quad (5)$$

Matematiksel modelde, amaç fonksiyonu (1) her bir modelin çevrim sürelerinin toplamının minimizasyonudur. Denklem 2 her görevin mutlaka bir istasyona atanmasını garanti etmektedir. Öncelik ilişkileri denklem 3 eşitsizliği ile sağlanmıştır. Denklem 4 ise istasyon süresinin çevrim süresini aşmasını engellemektedir. Ayrıca bunun tersi olarak çevrim süresini en büyük istasyon süresine eşitlemektedir. Karar değişkeninin tipi ise denklem 5 ile tanımlanmıştır.

4 Kısıt programlama

Kısıt programlama, bilgisayar programlamanın esnekliğini ve matematiksel programlamanın çözüme ulaşma becerisini bir araya getiren bir programlama yaklaşımıdır. Kısıt programlama ile matematiksel programlamada tanımlanması zor olan sayısal veya sayısal olmayan eşitlik/eşitsizlikler daha kolay bir şekilde programlanabilmektedir. Buna karşılık bilgisayar programlamada bulunmayan matematiksel programlamanın en önemli özelliklerinden biri olan belirli bir amaca göre problemin çözülmesi ile optimum çözümlere ulaşılmaktadır [42].

Bir kısıt programlama yapısı şu şekilde tanımlanmaktadır. X_i bir karar değişkeni olmak üzere, bu değişkenin alabileceği değerler kümesi S_i ile tanımlanmaktadır. Karar değişkenlerinin alabileceği değerleri sınırlayan bir C_k kısıtlar kümesinden oluşan yapının bütünü kısıt programlamayı ifade etmektedir. Bir çözümün olabilmesi, bütün karar değişkenlerinin bütün kısıtları sağlayan ve tanımlı oldukları alanlardan bir değeri almaları ile mümkündür.

Kısıt programlama yöntem olarak tam sayılı matematiksel programlama ile benzer özelliklere sahiptir. Puget ve Lustig [43] tarafından bu iki yöntem modelleme kabiliyeti, düğüm prosesi ve arama algoritması olmak üzere üç açıdan birbiri ile karşılaştırılmıştır.

Modelleme esnekliği yönünde değerlendirildiğinde her iki yöntem de karar değişkenlerine, kısıtlara ve amaç fonksiyonuna sahiptir ancak kısıt programlama dil olarak programlama dillerinin benzeri bir yapıyı kullanmasından dolayı modellemede özellikle dilsel ve şart ifadeleri daha kolay modellenmektedir. Düğüm işleme prosesi yönünden bakıldığında, tam sayılı matematiksel modelleme bir problemin yumuşatılmış sürekli doğrusal programlama çözümü ile elde edilen sonuçtan dallanılarak problemin sonucu bulunur. Kısıt programlama ise kısıt türetme algoritmaları kullanılarak düğümlere ulaşılır ve sonuç bulunur. Arama algoritması olarak kısıt programlama dal-sınır algoritmasına benzer bir arama algoritması kullanılmaktadır. Her bir düğüm karar değişkenini

ve her bir dal karar değişkeninin aldığı değeri göstermektedir. Her iki yöntem için de araştırmacılar arama algoritmaları geliştirmişlerdir. İki yöntemin birbirlerine göre avantajları olsa da herhangi birinin diğerine üstün olduğu söylenememektedir [43].

4.1 Kısıt sağlama ve kısıt optimizasyon problemleri

Kısıt programlama amaç fonksiyonunun olup olmasına göre iki tipte sınıflandırılır. Birinci tip olan Kısıt Sağlama Problemleri amaç fonksiyonu olmadan modellenir ve uygun bir çözüm veya bütün çözümler bulunana kadar çalıştırılır. Amaç fonksiyonu eklenerek çözülen kısıt programlama problemleri ise Kısıt Optimizasyon Problemleri olarak adlandırılır.

Bir kısıt sağlama problemi (X, D, C) şeklinde üçlü gösterimle ifade edilir. Burada;

X karar değişkenleri dizisini ifade eder: $X = \{x_1, \dots, x_n\}$

D karar değişkenlerinin alabileceği değerler kümesini tanımlar. Alan (domain) olarak ifade edilen bu bildirim matematiksel olarak şöyle ifade edilir: $v_i \in D(x_i) \quad i = 1, \dots, n$.

$C = \{C_1, \dots, C_n\}$ kısıtları ifade eder. C_j X karar değişkenleri arasındaki ilişkiyi ifade etmekte ve $var(C_j) = \{x_1, \dots, x_n\}$ ise değişkenlerin alt dizisini ifade etmektedir.

$S_i; C_j \subseteq D(x_1) \times \dots \times D(x_k), S_i \subseteq X$

Kısıt sağlama probleminin bir çözüme sahip olması için değişkenlerin aldığı değerlerin bütün kısıtları sağlıyor olması gerekir. Bir atama karar değişkenlerinin aldığı değerler seti şeklindedir $A = \{(x_1, v_1), (x_2, v_2), \dots, (x_p, v_p)\}$. Eğer atama bütün kısıtları sağlıyorsa uygun bir atama, eğer sağlamıyorsa uygun olmayan bir atama olarak isimlendirilir.

4.2 Arama ağacı

Arama ağacında, karar değişkeni bir düğüm olarak ve değişkenlerin alabileceği olası değerler ise dallar olarak gösterilir. Arama bir boş atama kümesi ile başlar ve değer ataması yapılmayan hiçbir değişken kalmayınca kadar devam eder. Eğer aramada bir karar değişkeni için kısıtları da göz önüne alarak uygun bir değer bulunamazsa geri dönüş mekanizması çalışır ve diğer dallar denir. Bu geri dönüş algoritması tekrar edilerek çözüm ağacı iteratif olarak üretilir [44].

Bir arama ağacında dalları tarama arama stratejileri ile yapılmaktadır. Öncelikle derinlik arama stratejisi, arama bir dallanma ile başlamakta ve her seferinde önce komşu düğüme, eğer komşular bitmiş ise sonrasında da bir önceki düğümün komşusuna geçişlerle bütün arama uzayını tarar. Bir diğer strateji olan çok noktalı arama stratejisinde uygun arama uzayının farklı bölgeleri taranarak çözümler elde edilir. Yeniden başlamalı arama stratejisinde ise belirlenen bir hata değerine veya zaman sınırına ulaşıldığında arama yeniden başlatılır. Otomatik arama stratejisinde ise bu üç stratejiden uygun olana karar verip aramaya seçilen stratejiyi kullanılarak devam eder.

4.3 Tanım kümesi azaltma

Tanım Kümesi Azaltma her bir değişken için uygun değer aralığının kısıtlara göre uygun olmayan değerlerin çözüm alanından çıkarılmasıdır. Örneğin, X_1 değer aralığı [2,5] ise ve X_2 değer aralığı [1,6] ise ve kısıdımız $X_1 \geq X_2$ ise; burada X_1 'in alabileceği değerler aralığı [2,5] ve X_2 'nin alabileceği değerler aralığı [1,5] olarak güncellenecektir. X_2 'nin çözüm aralığından 6 çıkarılacaktır. Bu şekilde her bir değişkenin kısıtlara göre

değer aralığının güncellenmesi tanım kümesi azaltma olarak adlandırılmıştır.

Bir tam sayılı matematiksel programlamada kısıtları sağlamak için iterasyonlar yapıldığından kısıt sayısının artması problemin optimum çözümünü bulmayı zorlaştırmaktadır. Her bir kısıtı sağlamak için dallanmalar yapılması gerekmektedir. Ancak kısıt programlamada ise kısıtların sayısının artması uygun çözüm aralığını daraltacağı için optimum çözüme ulaşmayı kolaylaştırmaktadır.

4.4 Kısıt türetme

Tanım kümesi azaltma bölümündeki örneğimizi kullanarak açıklarsak; arama ağacında dallanma için ilk olarak X_1 seçilmiş ve $X_1 = 3$ değerini almış olsun. X_1 kısıtı güncellenir ve yeni kısıdımız $3 \geq X_2$ olur. Bu durumda X_2 karar değişkeni 4 ve 5 değerlerini alamayacaktır. Bu iki değer çözüm aralığından çıkarılacaktır. Benzer şekilde X_2 önce seçilmiş ve 4 değerini almış olsa, X_1 'in çözüm aralığından 2 ve 3 çıkarılacaktır. Bu şekilde yapılan işlemlerin bütününe kısıt türetme denir.

Kısıt programlamada arama algoritmalarında yay tutarlılık, alan tutarlılık veya k-tutarlılık gibi yöntemler kullanılarak kısıt türetme yapılmaktadır. Bu aşama daha çok bilgisayar mühendisliğini ilgilendirdiği için çalışma konusu dışında kalmaktadır.

5 Karma modellenli montaj hattı dengeleme problemi-2 kısıt programlama modeli

Matematiksel modeli temel olarak KMMHDP-2 için kısıt programlama modeli oluşturulmuştur. Kısıt programlama mantığında açıklayacak olursak SN değişkeni tamsayı değerleri alabilen bir karar değişkenidir. Alabileceği değerler kümesi tüm tamsayı değerlerini kapsamaktadır. Karar değişkenlerinin alabileceği değerler kümesi modelde verilen kısıt 7-12 ile sınırlandırılmıştır. Bütün bu kısıtları sağlayan karar değişkenleri kümesi bir çözümü oluşturmaktadır. KMMHDP-2 kısıt programlama modeli şu şekildedir:

Amaç:

$$\text{Min} \sum_{m=1}^M C_m \quad (2)$$

Kısıtlar:

$$SN_i \leq SN_j \quad (i, j) \in Pr \quad (3)$$

$$SN_i \leq L_i \quad i = 1, \dots, N \quad (4)$$

$$SN_i \geq E_i \quad i = 1, \dots, N \quad (5)$$

$$\exists [SN_i] = k \quad k = 1, \dots, S; \quad i = 1, \dots, N \quad (6)$$

$$C_m \geq \sum_{i=1}^{|T|} \tau_{im} * Y_{ik} \quad k = 1, \dots, S; \quad m = 1, \dots, M \quad (7)$$

$$SN_i \in [E_i, L_i] \text{ ve tamsayı } i = 1, \dots, N \quad (8)$$

$$Y_{ik} = \begin{cases} 1, & SN_i = k \\ 0, & SN_i \neq k \end{cases} \quad (9)$$

Amaç (6), modellerin çevrim sürelerinin toplamının minimizasyonunu ifade etmektedir. Kısıt (7) görevler arasındaki öncelik ilişkilerini sağlamaktadır. Eğer i görevi j görevinden önce yapılması gerekiyor ise i görevinin atıldığı

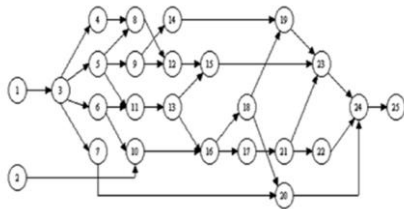
istasyonun j görevinin atandığı istasyondan önce veya aynı olmasını garanti etmektedir. Kısıt (8) görevin atanabileceği son istasyon numarasının hesaplanan L değerinden küçük olmasını, kısıt (9) ise hesaplanan E değerinden büyük olmasını sağlamaktadır. Kısıt (10) istasyonun açılması için gerek ve yeter şart olan bir istasyona en az bir görevin atanmasını garanti etmektedir. Her bir model için çevrim süresinin en az o modelin en yüksek istasyon süresine eşit olması kısıt (11) ile sağlanmaktadır. Kısıt (12) ve (13) değişkenlerin alacağı değer aralıklarını ve tipini tanımlamaktadır.

Burada kullanılan E ve L değerleri Gökçen ve Erel [7] tarafından önerilen denklem 14 ve 15'teki formüller ile hesaplanmıştır. Formüllerde kullanılan çevrim süresi değeri için teorik çevrim süresi değeri kullanılmıştır.

$$E_i = \max_{m=1, \dots, M} \left[\frac{t_{im} + \sum_{j \in PR_i} t_{jm}}{C_m} \right]^+ \quad (10)$$

$$L_i = \min_{m=1, \dots, M} \left[K + 1 - \left[\frac{t_{im} + \sum_{j \in S_i} t_{jm}}{C_m} \right]^+ \right] \quad (11)$$

Modeli bir örnek problem üzerinde açıklayacak olursak: Problemin verileri Simario ve Vilarinho [12]'nin çalışmasından alınmıştır. Çalışmalarında, bizim çalışmamızdan farklı olarak, bir istasyonda birden fazla işçinin de bulunabileceğini yani paralel istasyon olabileceğini kabul ederek sabit işçi sayısına göre dengeleme yapmışlardır. Bu montaj hattında 25 görev bulunmakta ve sabit 8 istasyon sayısına göre dengeleme yapılmıştır. Hat üzerinde 2 model üretildiği varsayılmıştır. Problemin birleştirilmiş öncelik diyagramı Şekil 5'te verilmiştir. Modellerin görev süreleri ise Tablo 1'de verilmiştir. Tabloda yer alan görev sürelerinden "0" olanlar o model için o görevin yapılmadığını belirtmektedir. Tablo 2'de ise görevler için hesaplanan E ve L değerleri verilmiştir.



Şekil 5: Örnek problem birleştirilmiş öncelik diyagramı.

Tablo 1: Modellerin görev süreleri.

Görev No	Görev		Görev No	Görev		Görev No	Görev		Görev No	Görev	
	tA	tB		tA	tB		tA	tB		tA	tB
1	0	20	8	0	20	15	55	55	22	47	47
2	77	77	9	66	66	16	19	20	23	96	82
3	73	73	10	25	25	17	37	0	24	41	37
4	150	150	11	55	55	18	94	94	25	125	0
5	88	88	12	71	71	19	13	13			
6	62	0	13	59	59	20	0	90			
7	36	0	14	13	0	21	20	20			

Tablo 2: Görevlerin E ve L değerleri.

Görev No	E	L	Görev No	E	L	Görev No	E	L	Görev No	E	L
1	1	1	8	3	6	15	5	7	22	4	7
2	1	5	9	2	6	16	3	6	23	7	7
3	1	1	10	2	5	17	3	7	24	8	7
4	2	5	11	2	5	18	4	6	25	8	8
5	2	3	12	4	6	19	5	7			
6	1	4	13	3	5	20	5	7			
7	1	7	14	2	7	21	4	7			

Problemin kısıt programlama modeli ILOG CPLEX Optimization Studio 12.6 CP Solver ile modellenip çözüldüğünde elde edilen atama değerleri Tablo 3'te görülmektedir. Ayrıca her bir istasyon için modellerin istasyon süreleri Tablo 4'teki gibidir.

Tablo 3: Çözüm sonucu görevlerin atandığı istasyon numaraları.

Görevler	İstasyon No	Görevler	İstasyon No	Görevler	İstasyon No
1	1	10	4	19	6
2	1	11	3	20	7
3	1	12	5	21	6
4	4	13	3	22	6
5	2	14	2	23	7
6	3	15	5	24	8
7	7	16	5	25	8
8	5	17	5		
9	2	18	6		

Tablo 4: Model bazında istasyon süreleri.

İstasyon No	Model 1	Model 2
1	150	170
2	167	154
3	176	114
4	175	175
5	182	166
6	174	174
7	132	172
8	166	37

Bulunan sonuçlara göre amaç değeri minimum 357 elde edilmiştir. Model 1 için çevrim süresi 182 ve model 2 için çevrim süresi 175 olarak bulunmuştur.

6 Deneysel çalışmalar ve problem bileşenlerinin analizi

Deneysel çalışmalar için kullandığımız veri setleri Simaria [45] Doktora Tezinden ve www.assembly-line-balancing.de [46] sitesinden alınmıştır. Problemler 25, 28, 30, 45, 70, 75 ve 297 görevden oluşmaktadır. Hat üzerinde 2, 3 veya 4 ürün üretildiği varsayılmıştır. 25, 28, 30 görevli problem sabit 8, 45 görevli problem sabit 12, 70 ve 75 görevli problemler sabit 20 ve 297 görevli problem sabit 45 istasyon sayısına göre dengelenmiştir (Tablo 5).

Problemler tam sayılı programlama modeli ve önerdiğimiz kısıt programlama modeli kullanılarak çözdürülmüştür. Her iki tip

model de ILOG CPLEX Optimization Studio 12.6 kullanılarak modellenmiştir. Her bir problem 1 saat zaman sınırı ile çözdürülmüştür. Çözüm sonuçları karşılaştırılmalı olarak Tablo 6'da verilmiştir. Çözüm sonuçlarına göre tam sayılı programlama modeli zaman sınırı içinde 6 problemde optimum sonuca ulaşmıştır. Kısıt programlama modeli ile 4 problem için optimum sonuca ulaşmıştır. 14 problemin 8'inde kısıt programlama modeli matematiksel modele göre eşit veya daha iyi sonuçlara ulaşmıştır.

Tablo 5: Deneysel setleri.

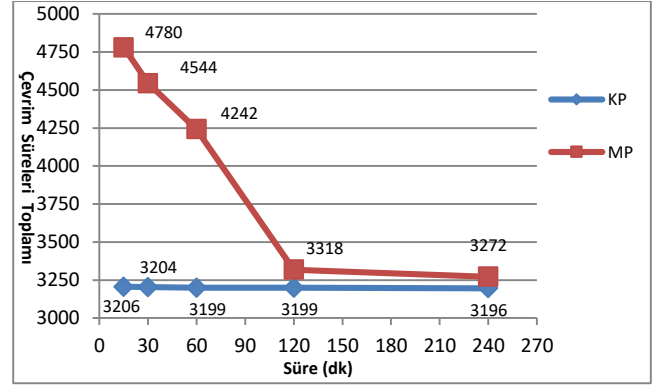
	Model Sayısı	Görev Sayısı	İstasyon Sayısı
Simaria_Vilarinhio	2	25	8
Simaria_Vilarinhio	3	25	8
Heskiaoff	2	28	8
Heskiaoff	3	28	8
Sawyer	2	30	8
Sawyer	3	30	8
Kilbridge	2	45	12
Kilbridge	3	45	12
Tongue	2	70	20
Tongue	3	70	20
WeeMag	2	75	20
WeeMag	4	75	20
Scholl	2	297	45
Scholl	4	297	45

Tablo 6: Deneysel çalışmaların sonuçları.

	Matematiksel Model		Kısıt Programlama	
	En İyi Değer	CPU Süresi	En İyi Değer	CPU Süresi
Simaria_Vilarinhio 2	357	0.3	357	0.2
Simaria_Vilarinhio 3	497	0.5	497	1
Heskiaoff 2	488	610	490	42
Heskiaoff 3	664	2	665	180
Sawyer 2	367	0.7	367	7
Sawyer 3	658	2.5	658	17
Kilbridge 2	375	460	376	1040
Kilbridge 3	597	240	602	123
Tongue 2	403	1460	400	2980
Tongue 3	615	100	612	524
WeeMag 2	154	235	159	28
WeeMag 4	311	290	322	2240
Scholl 2	4242	3550	3199	2918
Scholl 4	8021	2150	6495	2659

Kısıt programlama ve matematiksel programlamanın çözümü bulmadaki etkinliklerinin belirlenmesi amacıyla 2 modeli Scholl problemi uzun süreli çalıştırılmış ve sonuçlar karşılaştırılmıştır. Şekil 6'da verilen grafikte de görüldüğü

üzere kısıt programlama modeli daha kısa sürede daha iyi sonuçlara ulaşmaktadır.



Şekil 6: Scholl 2 çözüm sonuçlarının karşılaştırılması.

Bir KMMHDP-2 için değişebilen girdi parametreleri istasyon sayısı, model sayısı ve görev sayısıdır. Bu parametrelerin problemin çözümüne etkileri ele alınmıştır. 25 görevli Simaria ve Vilarinhio ve 30 görevli Sawyer problemleri analizler için kullanılmıştır. Buna göre:

İstasyon sayısı değişimi: 25 görevli ve 2 modelli örnek problemimiz üzerinde istasyon sayısı değişikliği ele alınmıştır. İstasyon sayısı 5 ve 8 olduğu durumlara göre karşılaştırmalar yapılmıştır. Hesapladığımız E ve L değerlerini her iki durum içinde dikkate aldığımızda ve öncelik ilişkileri olmadığını varsaydığımızda 5 istasyon için 4×10^8 kombinasyon ve 8 istasyon olduğunda ise bu sayı 5×10^{11} olmaktadır. İstasyon sayısı 3 artmasına karşılık kombinasyon sayısı yaklaşık 1125 kat artmıştır. Bu artış kısıt programlama modeli için karar değişkenlerinin alabileceği değer aralıklarını artırmaktadır ve optimum çözümü bulmayı zorlaştırmaktadır.

Model sayısı değişimi: Model sayısındaki değişikliğin analizi için 25 görevli örnek problem ele kullanılmıştır. Bu problem için 2 ve 3 modelli hat olduğu varsayılmış ve sabit 5 istasyon sayısına göre analizler yapılmıştır. Model sayısının ve modellerin değişimi her bir model için görev sürelerini de farklılaştırmaktadır. Buna bağlı olarak E ve L değerleri değişmektedir. Bu değişikliğin sonucunda kombinasyon sayısı düşebileceği gibi daha fazla artabilmektedir. Bu yüzden model sayısının değişmesi net olarak problemin çözümünü zorlaştırır sonucuna varılamaz.

Görev sayısı değişimi: Montaj hatlarında görev sayısının artması çözümü zorlaştırdığı gibi çözüm süresini de artırmaktadır. Bir montaj hattı probleminde görev sayısının 1 artışı o görevin E ve L değerlerine bağlı olarak kombinasyon sayısının katlanarak artmasına sebep olacaktır. Bu da problemin arama uzayını büyüüttüğünden çözüm süresini artırmaktadır. Görev sayısı yani karar değişkeni sayısı arttığında kısıt programlama modeli ile optimum çözümü bulmak daha uzun zaman alacaktır.

7 Sonuç ve gelecek çalışmalar

Montaj hatları birçok sanayi kuruluşunda uzun süredir kullanılmaktadır. Bu yaygın kullanımından dolayı araştırmacıların fazla sayıda çalışma yaptıkları bir alan olmuştur. Ancak, tip-2 olarak tarif edilen çevrim süresinin minimizasyonunun amaçlandığı problem tarzında az sayıda çalışma yapıldığı literatür araştırmalarında görülmektedir. Özellikle çalışmamızda ele aldığımız karma modelli montaj hattı dengeleme problemleri için bu sayının çok az olduğu Şekil 3'te görülmektedir.

Çalışmada literatürde az sayıda çalışma bulunan KMMHDP-2 problemi için kısıt programlama modeli önerilmiştir. Problemin çözümünde her bir modelin çevrim süresi değerlerinin toplamının minimizasyonu amaçlanmıştır. Kısıt programlama modelinin KMMHDP-2 probleminin çözümünde kullanılabilecek alternatif bir modelleme tekniği olduğu gösterilmiştir. Modelin sonuçları matematiksel modelle elde edilen sonuçlarla karşılaştırıldığında her iki modelin birbirine bariz bir üstünlük sağladığı söylenemez.

İleriki çalışmalarda, önerdiğimiz kısıt programlama modeline entegre edilecek arama algoritmaları ve arama kuralları kullanılarak arama uzayının daha hızlı taranması sağlanıp daha iyi çözümlere daha hızlı ulaşılabilecektir. Ayrıca kısıt programlamanın arama uzayını taraması için sezgisel algoritmalar da kullanılabilir. Önerilen kısıt programlama modeli diğer montaj hattı dengeleme problemlerine de (stokastik, tip-1, tip-E, paralel gibi) uygulanabilir.

8 Kaynaklar

- [1] Özcan U, Toklu B. "Balancing of mixed-model two-sided assembly lines". *Computers & Industrial Engineering*, 57(1), 217-227, 2009.
- [2] Arcus AL. "A computer method of sequencing operations for assembly lines". *International Journal of Production Research*, 4(4), 259-277, 1965.
- [3] Thomopoulos NT. "Line balancing-sequencing for mixed-model assembly". *Management Science*, 14(2), 59-75, 1967.
- [4] Thomopoulos NT. "Mixed model line balancing with smoothed station assignments". *Management Science*, 16(9), 593-603, 1970.
- [5] Macaskill JLC. "Production-line balances for mixed-model lines". *Management Science*, 19(4), 423-434, 1972.
- [6] Haq AN, Jayaprakash J, Rengarajan K. "A hybrid genetic algorithm approach to mixed-model assembly line balancing". *The International Journal of Advanced Manufacturing Technology*, 28(3), 337-341, 2006.
- [7] Gokcen H, Erel E. "A goal programming approach to mixed-model assembly line balancing problem". *International Journal of Production Economics*, 48, 177-185, 1997.
- [8] Gokcen H, Erel E. "Binary integer formulation for mixed-model assembly line balancing problem". *Computers & Industrial Engineering*, 34(2), 451-461, 1998.
- [9] Matanachai S, Yano CA. "Balancing mixed-model assembly lines to reduce work overload". *IIE Transactions*, 33(1), 29-42, 2001.
- [10] Rekiek B, De Lit P, Delchambre A. "Designing mixed-product assembly lines". *IEEE Transactions on Robotics and Automation*, 16(3), 268-280, 2000.
- [11] Karabatı S, Sayın S. "Assembly line balancing in a mixed-model sequencing environment with synchronous transfers". *European Journal of Operational Research*, 149(2), 417-429, 2003.
- [12] Simaria AS, Vilarinho PM. "A genetic algorithm based approach to the mixed-model assembly line balancing problem of type II". *Computers & Industrial Engineering*, 47(4), 391-407, 2004.
- [13] Mendes AR, Ramos AL, Simaria AS, Vilarinho PM. "Combining heuristic procedures and simulation models for balancing a PC camera assembly line". *Computers & Industrial Engineering*, 49(3), 413-431, 2005.
- [14] Bock S. "Using distributed search methods for balancing mixed-model assembly lines in the automotive industry". *OR Spectrum*, 30, 551-578, 2008.
- [15] Hwang R, Katayama H. "Integrated procedure of balancing and sequencing for mixed-model assembly lines: a multi-objective evolutionary approach". *International Journal of Production Research*, 48(21), 6417-6441, 2010.
- [16] Yagmahan B. "Mixed-model assembly line balancing using a multi-objective ant colony optimization approach". *Expert Systems with Applications*, 38(10), 12453-12461, 2011.
- [17] Mamun AA, Khaled AA, Ali SM, Chowdhury MM. "A heuristic approach for balancing mixed-model assembly line of type I using genetic algorithm". *International Journal of Production Research*, 50(18), 5106-5116, 2012.
- [18] Yang C, Gao J, Sun L. "A multi-objective genetic algorithm for mixed-model assembly line rebalancing". *Computers & Industrial Engineering*, 65(1), 109-116, 2013.
- [19] Sawik T. "Monolithic vs. hierarchical balancing and scheduling of a flexible assembly line". *European Journal of Operational Research*, 143(1), 115-124, 2002.
- [20] Cevikcan E, Durmusoglu M B, Unal ME. "A team-oriented design methodology for mixed model assembly systems". *Computers & Industrial Engineering*, 56(2), 576-599, 2009.
- [21] Chakravarty AK, Shtub A. "Balancing mixed model lines with in-process inventories". *Management Science*, 31(9), 1161-1174, 1985.
- [22] McMullen PR, Frazier GV. "Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel workstations". *International Journal of Production Research*, 36(10), 2717-2741, 1998.
- [23] McMullen PR, Tarasewich P. "Using ant techniques to solve the assembly line balancing problem". *IIE Transactions*, 35, 605-617, 2003.
- [24] Merengo C, Nava F, Pozzetti A. "Balancing and sequencing manual mixed-model assembly lines." *International Journal of Production Research*, 37(12), 2835-2860, 1999.
- [25] Zhao X, Liu J, Ohno K, Kot S. "Modeling and analysis of a mixed-model assembly line with stochastic operation times". *Naval Research Logistics*, 54(6), 2007.
- [26] McMullen PR, Frazier GV. "A heuristic for solving mixed-model line balancing problems with stochastic task durations and parallel stations". *International Journal of Production Economics*, 51(3), 177-190, 1997.
- [27] Bukchin J, Dar-El EM, Rubinovitz J. "Mixed-model assembly line design in a make-to-order environment". *Computers & Industrial Engineering*, 41(4), 405-421, 2002.
- [28] Tiacci L. "Event and object oriented simulation to fast evaluate operational objectives of mixed model assembly lines problems". *Simulation Modelling Practice and Theory*, 24, 35-48, 2012.
- [29] Manavizadeh N, Rabbani M, Moshtaghi D, Jolai F. "Mixed-model assembly line balancing in the make-to-order and stochastic environment using multi-objective evolutionary algorithms". *Expert Systems with Applications*, 39(15), 12026-12031, 2012.

- [30] Venkatesh JVL, Dabade BM. "Evaluation of performance measures for representing operational objectives of a mixed model assembly line balancing problem". *International Journal of Production Research*, 46(22), 6367-6388, 2008.
- [31] Hop NV. "A heuristic solution for fuzzy mixed-model line balancing problem". *European Journal of Operational Research*, 168(3), 798-810, 2006.
- [32] Erel E, Gokcen H. "Shortest-route formulation of mixed-model assembly line balancing problem". *European Journal of Operational Research*, 116(1), 194-204, 1999.
- [33] Topaloglu S, Salum L, Supciller AA, "Rule-based modeling and constraint programming based solution of the assembly line balancing problem". *Expert Systems with Applications*, 39(3), 3484-3493, 2012.
- [34] Terekhov D, Doğru MK, Özen U, Beck JC. "Solving two-machine assembly scheduling problems with inventory constraints". *Computers & Industrial Engineering*, 63(1), 120-134, 2012.
- [35] Li H, Womer K. "Optimizing the supply chain configuration for make-to-order manufacturing." *European Journal of Operational Research*, 221(1), 118-128, 2012.
- [36] Siala M, Hebrard E, Huguet MJ. "A study of constraint programming heuristics for the car-sequencing problem". *Engineering Applications of Artificial Intelligence*, 38, 34-44, 2015.
- [37] Bockmayr A, Pizaruk N. "Solving an assembly line balancing problems by combining IP and CP". *Paper presented at the 6th Annual Workshop of ERCIM Working Group on Constraints*, Prague, Czech Republic, 18-20 June 2001.
- [38] Pastor R, Ferrer L, García A. "Evaluating optimization models to solve SALBP". *Lecture Notes in Computer Science*, 4705, 791-803, 2007.
- [39] Alağaç HM, Yüzükırmızı M, Türker AK. "Stokastik montaj hatlarının kısıt programlama ve kapalı kuyruk ağları ile denetlenmesi". *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 28(2), 231-240, 2013.
- [40] Öztürk C, Tunalı S, Hnich B, Örnek MA. "Balancing and scheduling of flexible mixed model assembly lines". *Constraints*, 18(3), 434-469, 2013.
- [41] Sivasankaran P, Shahabudeen P. "Literature review of assembly line balancing problems". *The International Journal Advanced Manufacturing Technology*, 73(9), 1665-1694, 2014.
- [42] Apt KR. *Principles of Constraint Programming*. New York, USA, Cambridge University Press, 2003.
- [43] Puget JF, Lustig I. "Constraint programming and maths programming". *The Knowledge Engineering Review*, 16(1), 5-23, 2001.
- [44] Solnon, C. *Ant Colony Optimization and Constraint Programming*, New York, USA, Wiley-ISTE, 2010.
- [45] Simaria, AS. Assembly line balancing: new perspectives and procedures. PhD Thesis, Universidade de Aveiro, Aveiro, Portugal, 2006.
- [46] Assembly Line Balancing Research. "Assembly line balancing data sets for MMALBP". <http://alb.mansci.de/index.php?content=classview&content2=classview&content3=classviewdfree&content4=classview&classID=58&type=dl> (25.05.2015).