

**T.C.
SAKARYA UYGULAMALI BİLİMLER ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**GENETİK ALGORİTMA KULLANILARAK
PROJE YÖNETİMİ KAYNAK SEVİYELENDİRME
PROBLEMLERİNİN ÇÖZÜLMESİ**

YÜKSEK LİSANS TEZİ

Merve BULUT

Enstitü Anabilim Dalı : İNŞAAT MÜHENDİSLİĞİ
Tez Danışmanı : Dr. Öğr. Üyesi Sedat Semih
ÇAĞLAYAN

Aralık 2022

T.C.
SAKARYA UYGULAMALI BİLİMLER ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

GENETİK ALGORİTMA KULLANILARAK
PROJE YÖNETİMİ KAYNAK SEVİYELENDİRME
PROBLEMLERİNİN ÇÖZÜLMESİ

YÜKSEK LİSANS TEZİ

Merve BULUT

Enstitü Anabilim Dalı : İNŞAAT MÜHENDİSLİĞİ

Bu tez 28/12/2022 tarihinde aşağıdaki jüri tarafından
oybirliği/oyçokluğu ile kabul edilmiştir.

JÜRİ	BAŞARI DURUMU
Jüri Başkanı: Dr. Öğr. Üyesi Sedat Semih ÇAĞLAYAN	
Üye: Dr. Öğr. Üyesi Abdülkadir ÖZDEN	
Üye: Dr. Öğr. Üyesi Esra DOBRUCALI	

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Merve Bulut

28/12/2022

TEŐEKKÜR

Yüksek lisans eğitimin süresince kıymetli bilgi ve birikimlerinden faydalandığım, her konuda bilgi ve desteğini sunmaktan geri kalmayan, araştırmanın planlanmasından yazılmasına dek bütün aşamalarında yardımlarını esirgemeyen, teşvik eden, aynı titizlikte beni yönlendiren kıymetli danışman hocam Dr. Sedat Semih ÇAĞLAYAN'a teşekkürlerimi sunarım.

İÇİNDEKİLER

BEYAN.....	ii
TEŞEKKÜR	i
İÇİNDEKİLER	ii
KISALTMALAR	iv
SİMGELER.....	vi
TABLolar LİSTESİ.....	vii
ŞEKİLLER LİSTESİ.....	viii
ÖZET.....	ix
SUMMARY	x

BÖLÜM 1.

GİRİŞ	1
1.1. Tezin Amacı	2
1.2. CPM	2
1.2.1. CPM'in kullanılma sebepleri	3
1.2.2. Kritik yol hesaplama	3
1.3. Kaynak Tahsisi	6
1.4. Kaynak Dengeleme ve Kaynak Tahsisi.....	6

BÖLÜM 2.

LİTERATÜR ÇALIŞMASI	8
2.1. Planlama Kalitesini Arttırmaya Yönelik Çalışmalar.....	8
2.2. Planlama Optimizasyonu İçin Genetik Algoritmaların Kullanımı.....	10

BÖLÜM 3.

PROJE YÖNETİMİNDE KAYNAK Dengeleme	15
3.1. Kaynak Dengelemenin Formülasyonu	16
3.1.1. Aktivitelerin Planlanması.....	17
3.1.2. Kaynak Dengeleme Problemleri	18
3.1.3. Kaynak Dengelemenin Amaç Fonksiyonları	19
3.1.4. Optimizasyon Modeli.....	20
3.2. Normalleştirilmiş Entropiye Dayalı Kaynak Seviyelendirme.....	20

3.2.1. Entropi.....	20
3.2.2. Kaynak Dengelemede Entropi	20
3.2.3. Normalleştirilmiş Kaynak Seviyelendirme Entropisi	21
BÖLÜM 4.	
GENETİK ALGORİTMA	23
4.1. Genetik Algoritmaların Avantajları	23
4.2. Genetik Algoritmaların Dezavantajları	24
4.3. Uyarlanabilir Genetik Operatörlü Genetik Algoritma	24
4.4. Genetik Algoritma Hesaplama Süreci.....	25
BÖLÜM 5.	
METODOLOJİ	29
5.1. Genetik Algoritma İle Optimizasyonun Yapılması.....	32
5.2. Myfitness Fonksiyonu	33
5.3. MyConstraints Fonksiyonu	35
5.4. Main Fonksiyonu.....	37
BÖLÜM 6.	
BULGULAR VE TARTIŞMALAR	38
BÖLÜM 7.	
SONUÇ VE ÖNERİLER.....	41
7.1. Sonuç	41
7.2. Çalışmanın Literatüre Faydası.....	42
7.2.1. Teorik katkısı	42
7.2.2. Pratik katkısı	42
7.2.3. Çalışmanın kısıtları	43
7.3. Öneriler.....	43
KAYNAKLAR	44

KISALTMALAR

BT	: Bilgi Teknolojisi
BY	: Bilgi Yönetimi
CPM	: Critical Path Method (Kritik Yol Metodu)
D&C'ler	: Delays and Conflicts (Gecikmeler ve Çatışmalar)
EC	: Enerji Tüketimi
EDAM	: Effect-Based Delay Analysis Method (Etki Tabanlı Gecikme Analizi)
EF	: Early Finish (Erken Bitiş)
ES	: Early Start (Erken Başlangıç)
GA	: Genetik Algoritma
GARLS	: Genetic Algorithms Resource Leveling System (Genetik Algoritma Tabanlı Kaynak Dengeleme Çizelgeleme Sistemi)
LF	: Late Finish (Geç Bitiş)
LS	: Late Start - LS (Geç Başlangıç)
MOO	: Multiple Objective Optimization (Çok Amaçlı Optimizasyon)
PERT	: Program Evaluation and Review Technique (Program Değerlendirme ve Gözden Geçirme Tekniği)
PM	: Project Managers (Proje Yöneticileri)
QGA	: Quantum Genetic Algorithms (Kuantum Genetik Algoritması)
RCPSP	: Resource Constraint Project Scheduling Problems (Kaynak Kısıtlı Proje Çizelgeleme Problemi)
RII	: Relative Importance Index (Görelî Önem İndeksi)
RLP	: Resource Leveling Problems (Kaynak Dengeleme Problemleri)
ÖS	: Öteleme Süresi
EB	: Erken Başlangıç

ET : Erken Tamamlama
GB : Ge Bařlangı
GT : Ge Tamamlama
TB : Toplam Bolluk
S : Aktivite Sresi



SİMGELER

- $x(i)$: Aktivitenin kaç gün teleneceđi
 $d(i)$: Her bir aktivitenin sresi
 $dr(i)$: i . Aktivitenin gnlk kaynak ihtiyaı



TABLolar LİSTESİ

Tablo 5.1. Aktivite Süreleri, Öncülleri ve Günlük Kaynak Miktarları.....	30
--	----



ŞEKİLLER LİSTESİ

Şekil 3.1 : Ağ Diyagramı ve Kaynak Dengeleme Şeması	17
Şekil 3.2 : Farklı Proje Süreleri İçin Entropi Değerleri	211
Şekil 3.3 : (a)Kaynak Dağılımı 1 ve (b)Kaynak Dağılımı 2	22
Şekil 4.1 : Gen, Kromozom, Popülasyon Gösterimi	25
Şekil 4.2 : Genetik Algoritma Akış Şeması	266
Şekil 4.3 : Çaprazlama	27
Şekil 4.4 : Ebeveynler Arasında Gen Alışverişi.....	27
Şekil 4.5 : Yeni Yavru.....	27
Şekil 4.6 : Mutasyon Öncesi	28
Şekil 4.7 : Mutasyon Sonrası	28
Şekil 5.1 : Dengeleme Öncesi CPM Çözümü	30
Şekil 5.2 : Geleneksel Kutu Diyagramı Gösteri.....	31
Şekil 5.3 : Öteleme Yapılmadan Önceki Gantt Çizelgesi	31
Şekil 5.4 : Önerilen Kutu Diyagramı Gösterimi	32
Şekil 5.5 : Matlab’da Aktivite Sürelerinin Tanımlanması	34
Şekil 5.6 : Matlab’da Aktivitelerin Günlük Kaynak İhtiyacının Tanımlanması.....	34
Şekil 5.7 : Matlab’da Proje Süresinin Elde Edilmesi.....	35
Şekil 5.8 : Matlab’da MyConstraints Fonksiyonu Tanımlanması	35
Şekil 5.9 : Matlab’da For Döngüsü ile Gantt Çizelgesi Elde Edilmesi.....	36
Şekil 5.10 : Matlab’da Toplam Kaynak Miktarının Elde Edilmesi	36
Şekil 5.11 : Matlab’da MyConstraints Fonksiyonunun Elde Edilmesi.....	37
Şekil 5.12 : Matlab’da Main Fonksiyonunun Tanımlanması.....	37
Şekil 6.1 : Matlab’da Optimizasyon Sonucu.....	38
Şekil 6.2 : Optimizasyon sonrası elde edilen x değerleri.....	39
Şekil 6.3 : Kaynak Dengeleme Sonrası CPM Çözümü.....	39
Şekil 6.4 : Optimizasyon sonrası Gantt Çizelgesi.....	39

GENETİK ALGORİTMA KULLANILARAK PROJE YÖNETİMİ KAYNAK DENGELEME PROBLEMLERİNİN ÇÖZÜLMESİ

ÖZET

İnşaat projeleri birden fazla meslek disiplini gerektiren alanlarla birlikte çalışarak proje sürecini tamamlaması gerekmektedir. Projelerin başarıyla sonuçlanması için proje yönetim tekniklerine başvurulması kaçınılmazdır. İnşaat projelerinin karmaşık yapısı gereği planlanan süreden çok daha uzun sürede tamamlanması evrensel bir sorundur. Projenin planlanan süreden geç tamamlanması inşaat projelerinin başarısızlıkla sonuçlanmasına sebep olmaktadır veya proje performansını ciddi anlamda olumsuz yönde etkileyip maddi zararlara yol açmaktadır. Günümüze gelişerek gelmiş proje yönetim teknikleri ve teknolojinin sunduğu büyük bir kolaylık olan bilgisayar programlarına rağmen inşaat projeleri hala proje sürecinde gecikmeler, düşük performans, üretkenlikte azalma gibi çoğu sorunla her defasında karşılaşmaktadır.

Tez çalışmasında inşaat proje yönetiminde kullanılan kaynak dengeleme problemlerinin inşaat proje yönetimlerinde kullanımını neredeyse rastlanmayan genetik algoritma yöntemiyle çözülmesi amaçlanmıştır. Kaynak dengeleme proje planlamayı daha kaliteli hale getirmektedir. Kaynak dengeleme problemlerinin geleneksel yöntemlerle çözümü somut bir sonuç vermediğinden ve çok uzun süreçlerde çözüm sağlanabildiğinden optimizasyona ihtiyaç duyulmaktadır. Bu sorunu ortadan kaldırmak amacıyla bu çalışmada kaynak dengeleme problemlerini sezgisel optimizasyon yöntemlerinden genetik algoritmayla çözmek amaçlanmıştır. Bu tez çalışmasında genetik algoritmayla çözülmesi amaçlanan projelerde kısıtlı kaynaklar kullanılarak optimizasyon gerçekleştirilmiştir. Temel amaç, proje süresini minimum seviyede tutarak kısıtlı kaynaklar altında kaynak dengelemesinin yapılmasıdır.

Tez çalışmasındaki örnek proje geleneksel proje yöntemlerinde en sık kullanılan “Kritik Yol Metodu (Critical Path Method - CPM)” ile çözümlenip proje süresi elde edilmiştir. Geleneksel yöntemlerden farklı ve yeni olarak bu çalışmada CPM’de olmayan bir teknik eklenmiş ve bazı aktivitelerin ötelenerek projenin optimizasyonu sağlanmıştır. Öteleme işleminden sonra yeni proje süresi önce geleneksel yöntemlerle elde edilmiştir. Daha sonra ise aynı proje genetik algoritma yöntemi kullanılarak kısıtlı kaynaklar altında Matlab programında çözümlenmiştir. Elde edilen sonucun doğruluğu karşılaştırılıp onaylanmıştır. Genetik algoritma ile bilgisayar programında optimizasyon yapılması geleneksel yöntemlerden çok daha hızlı, güvenilir ve en optimum sonucu verdiği gözlemlenmiştir.

Anahtar Kelimeler: Kaynak Dengeleme, Genetik Algoritma, Kritik Yol Metodu, İnşaat Proje Yönetimi, Optimizasyon

SOLVING PROJECT MANAGEMENT RESOURCE LEVELING PROBLEMS USING GENETIC ALGORITHM

SUMMARY

Since construction projects need to complete the project process by working with fields that require more than one professional discipline, it is inevitable to apply project management techniques for the successful conclusion of the projects. Due to the complex nature of construction projects, it is a universal problem that they are completed in a much longer time than planned. Completion of the project after the planned time causes the construction projects to fail or seriously affects the performance of the project and causes financial losses. Despite the advanced project management techniques and computer programs, which are a great convenience provided by technology, construction projects still face many problems such as delays in the project process, poor performance, and decrease in productivity.

In this thesis, it is aimed to solve the resource leveling problems used in construction project management with the genetic algorithm method, which is almost not used in the construction project management. Resource leveling makes project planning better. Optimization is needed since the solution of resource leveling problems with traditional methods does not give a concrete result and can be solved in very long processes. In order to eliminate this problem, in this study, it is aimed to solve resource leveling problems with genetic algorithm, one of the heuristic optimization methods. In this thesis, optimization was carried out using limited resources in the projects aimed to be solved by genetic algorithm. The main purpose is to balance the resources under limited resources by keeping the project duration to a minimum.

The sample project in the thesis study was analyzed with the “Critical Path Method (CPM)”, which is the most frequently used in traditional project methods, and the project duration was obtained. Different from the traditional methods and new, a technique that is not in the Critical Path Method has been added in this study and the optimization of the project has been achieved by postponing some activities. After the offset process, the new project duration was first obtained by traditional methods. Then, the same project was solved in Matlab program under limited resources using genetic algorithm method. The accuracy of the obtained result was compared and confirmed. It has been observed that optimization in the computer program with the genetic algorithm gives much faster, more reliable and optimum results than traditional methods.

Keywords: Resource Leveling, Genetic Algorithm, CPM, Construction Project Management, Optimization

BÖLÜM 1. GİRİŞ

İnşaat projelerinin planlanan süreden çok daha uzun sürede tamamlanması, evrensel olarak var olan bir konudur [1]. Süreçteki bu gecikmeler inşaat projelerinin performansını olumsuz etkilemektedir [2]. Kapsamlı bilimsel çalışmalara ve gelişmiş yönetim tekniklerine rağmen, inşaat projeleri hala gecikmeler, çatışmalar, üretkenlik kaybı, düşük performans, koordinasyon ve daha pek çok sorunu çözememektedir. İnşaat projelerinin zayıf performansı küresel bir sorundur [3].

İnşaat projelerindeki performansı arttırmak için verimli bir proje yönetimi yapılmalıdır. Proje yönetiminin amacı, inşaat projelerinde bazı değişiklikler yaparak daha iyi proje performansı elde etmektir ve proje planlama, tasarım ve güvenlik yönetimi gibi birçok faaliyet sürece dahil edilebilir [4].

Proje yönetiminde optimizasyonun sağlanması için geleneksel yöntemler inşaat projeleri gibi büyük ölçekli ve karmaşık projelerde büyük oranda başarısız olduğundan bu tez çalışmasında bu konuya bir çözüm getirilmeye çalışılmıştır. Proje yönetiminde sıkça kullanılan “Kritik Yol Metodu (CPM)” ile çözümlenmiş bir proje genetik algoritma yöntemiyle optimize edilmektedir.

Tez çalışmasında proje yönetimi, kaynak dengeleme, genetik algoritma kavramları yapılan literatür çalışmaları neticesinde detaylı olarak açıklanmaktadır. Örnek projenin önce CPM ile çözümü sağlanmış ve toplam proje süresi elde edilmiştir. CPM ile projenin nasıl çözüldüğü çeşitli şekil ve diyagramlar ile detaylıca açıklanmaktadır. Örnek projenin CPM ile hesaplanan toplam proje süresi minimum seviyede tutulması amaçlanarak kaynak dengeleme yöntemleri ile yeni proje süresi hesaplanmaktadır.

Geleneksel yöntemlerle kaynak dengeleme öncesi proje süresi ve kaynak dengeleme sonrası proje süreleri hesaplanan örnek çalışmanın genetik algoritma ile optimizasyonu Matlab bilgisayar programında sağlanmaktadır. Genetik algoritma ile yapılan

optimizasyon için kullanılan karar deęişkenleri, kısıtlamalar ve fonksiyonlar belirtilmektedir. Optimizasyonun çözümlenmesi için karar deęişkenleri cinsinden kısıtlamalar ve objektif fonksiyonlar yazılmaktadır. Tüm bu işlemler sonucunda genetik algoritma ile elde edilen kaynak dengelemesi yapılmış yeni proje süresinin doğruluęu onaylanmaktadır.

1.1. Tezin Amacı

İnşaat proje yönetiminde kullanılan kaynak dengeleme problemlerinin sezgisel optimizasyondan yöntemlerinden genetik algoritma ile proje süresi optimizasyonu yapmak amaçlanmaktadır. Kaynak dengeleme proje planlamayı daha kaliteli hale getirmektedir. Kaynak dengeleme problemlerinin geleneksel yöntemlerle çözümlü somut bir sonuç vermediğinden ve çok uzun süreçlerde çözüm sağlanabildiğinden optimizasyona ihtiyaç duyulmaktadır. Bu sorunu ortadan kaldırmak amacıyla bu çalışmada kaynak dengeleme problemlerini sezgisel optimizasyon yöntemlerinden genetik algoritmayla çözmek amaçlanmıştır. Bu tez çalışmasında genetik algoritmayla çözümlenmesi amaçlanan projelerde kısıtlı kaynaklar kullanılarak optimizasyon gerçekleştirilmiştir. Temel amaç, proje süresini minimum seviyede tutarak kısıtlı kaynaklar altında kaynak dengelemesinin yapılmasıdır.

1.2. CPM

CPM bir projeyi tamamlamak için gerekli olan kilit görevlerin veya kritik görevlerin haritasını çıkarmayı içeren bir proje yönetimi tekniğidir. Bir proje yol haritası oluşturmak, nihai hedefe ulaşmak için yapılması gerekenleri görselleştirmeye yardımcı olur. Bu teknikten yararlanmak, görev bağımlılıklarını yönetmeye ve gerçekçi zaman çerçeveleri belirlemeye olanak tanır.

CPM, projenin tamamlanması için gerekli olan görevlerin belirlendiği ve zamanlama esnekliklerini belirlendiği bir tekniktir. Proje yönetiminde kritik bir yol, tüm projenin tamamlanması için zamanında tamamlanması gereken en uzun faaliyetler dizisidir. Kritik görevlerdeki herhangi bir gecikme, projenin toplam süresini uzatacaktır.

CPM, proje zaman çizelgesindeki en önemli görevleri keşfetme, görev bağımlılıklarını belirleme ve görev sürelerini hesaplama etrafında döner.

CPM, 1950'lerin sonlarında, verimsiz planlama nedeniyle artan maliyetler sorununu çözmek için bir yöntem olarak geliştirilmiştir. O zamandan beri CPM, projeleri planlamak ve görevleri önceliklendirmek için popüler hale geldi. Karmaşık projeleri bireysel görevlere ayırmaya ve projenin esnekliğini daha iyi anlamaya yardımcı olmaktadır.

1.2.1. CPM'in kullanılma sebepleri

CPM, projelerin nasıl planlanacağı, kaynakların nasıl tahsis edileceği ve görevlerin nasıl planlanacağı konusunda önemli bilgiler sağlar.

Bu yöntemin kullanılması için bazı nedenler:

1. Gelecek planlamasını geliştirir: CPM, beklentileri gerçek sonuçlarla karşılaştırmak için kullanılabilir. Mevcut projelerden kullanılan veriler, gelecekteki proje planları için bilgi sağlayabilir.
2. Daha etkili kaynak yönetimini kolaylaştırır: CPM, proje yöneticilerinin görevleri önceliklendirmesine yardımcı olarak onlara kaynakları nasıl ve nerede dağıtacakları konusunda daha iyi bir fikir verir.
3. Darboğazlardan kaçınmaya yardımcı olur: Projelerdeki darboğazlar değerli zaman kaybına neden olabilir. Bir ağ diyagramı kullanarak proje bağımlılıklarını çizmek, hangi etkinliklerin paralel olarak yürütülebileceği ve yürütülemeyeceği konusunda daha iyi bir fikir vermekte ve buna göre zamanlama yapılmasına olanak sağlamaktadır.

1.2.2. Kritik yol hesaplama

Kritik yolu bulmak, kritik ve kritik olmayan görevlerin süresine bakmayı içerir. Aşağıda, tüm aşamalar anlatılmaktadır.

1. Etkinlikler listelenir

Çıktıları üretmek için gereken tüm proje aktivitelerini veya görevlerini listelemek için bir iş kısırlım yapısı kullanılır. İş kısırlım yapısındaki faaliyetlerin listesi, CPM'nin geri kalanı için temel görevi görür.

2. Bağımlılıklar belirlenir

İş kırılım yapısına göre birbirine bağımlı görevler belirlenir. Bu ayrıca, diğer görevlerle paralel olarak yapılabilecek herhangi bir işin belirlenmesine yardımcı olmaktadır. Bağımlı görevlerin listesi, kritik yolu belirlemek için kullanılacak olan bir etkinlik dizisi olarak adlandırılır.

3. Bir ağ diyagramı oluşturulur

Bir sonraki adım, iş kırılım yapısını, faaliyetlerin kronolojisini gösteren bir akış şeması olan bir ağ diyagramına dönüştürmektir. Her görev için bir kutu oluşturulur ve görev bağımlılıklarını göstermek için oklar kullanılır.

4. Görev süresi tahmin edilir

En uzun kritik görevler dizisi olan kritik yolu hesaplamak için önce her bir etkinliğin süresinin tahmin edilmesi gerekir. Süreyi tahmin etmek için şunlar denir:

1. Deneyim ve bilgiye dayalı tahminler yapma
2. Önceki proje verilerine dayalı tahmin
3. Endüstri standartlarına dayalı tahmin

Alternatif olarak, ileri geçiş ve geri gidiş tekniği kullanılır:

4.1. İleri geçiş: Bu, önceden belirlenmiş bir başlangıç tarihini kullanarak erken başlangıç (Early Start - ES) ve erken bitiş (Early Finish - EF) tarihlerini hesaplamak için kullanılır. ES, hemen öncekilerden en yüksek EF değeridir, oysa EF, ES + süredir. Hesaplama, ilk aktivitenin ES'sinde 0 ile başlar ve program boyunca devam eder. ES ve EF tarihlerinin belirlenmesi, kaynakların projeye erken tahsis edilmesini sağlar.

4.2. Geriye doğru geçiş: Bu, geç başlangıç (Late Start - LS) ve geç bitiş (Late Finish - LF) tarihlerini hesaplamak için kullanılır. LS süre iken, LF bağlı olan aktivitelerin en düşük LS değeridir. Hesaplama, programlanan son etkinlikle başlar ve tüm program boyunca geriye doğru ilerler.

Erken ve geç başlangıç ve bitiş tarihleri, daha sonra her bir görevin bolluğunu veya zamanlama esnekliğini hesaplamak için kullanılır.

5. Kritik yol hesaplanır

Kritik yolun hesaplanması manuel olarak yapılabilir, ancak bunun yerine bir kritik yol algoritması kullanarak zamandan tasarruf edilebilir.

Kritik yolu manuel olarak hesaplama adımları şunlardır:

Adım 1: Her aktivitenin yanına başlangıç ve bitiş zamanı yazılır.

İlk aktivitenin başlangıç zamanı 0'dır ve bitiş zamanı aktivitenin süresidir. Bir sonraki aktivitenin başlangıç zamanı, önceki aktivitenin bitiş zamanıdır ve bitiş zamanı, başlangıç zamanı artı süredir. Bu tüm aktiviteler için yapılır.

Adım 2: Tüm dizinin süresini belirlemek için dizideki son etkinliğin bitiş süresine bakılır.

Adım 3: En uzun süreli faaliyetler dizisi kritik yoldur.

Kritik yolu belirdikten sonra, gerçek proje zamanlamasını buna göre oluşturulabilir.

6. Bolluk hesaplanır

Float veya bolluk, belirli bir görevin esneklik miktarını ifade eder. Sonraki görevleri veya proje bitiş tarihini etkilemeden görevin ne kadar ertelenebileceğini gösterir.

Değişkeni bulmak, projenin ne kadar esnekliğe sahip olduğunu ölçmek için yararlıdır. Bolluk, proje risklerini veya ortaya çıkan beklenmedik sorunları kapatmak için kullanılması gereken bir kaynaktır.

Kritik görevlerin sıfır dalgalanması vardır, bu da tarihlerinin ayarlandığı anlamına gelir. Pozitif değişken sayılara sahip görevler kritik olmayan yola aittir, yani projenin tamamlanma tarihini etkilemeden ertelenebilirler. Süre veya kaynaklar kısıtlıysa, kritik olmayan görevler atlanabilir.

Toplam bolluğun hesaplanması bir algoritma ile veya manuel olarak yapılabilir.

Toplam bolluk: Bu, projenin bitiş tarihini geciktirmeden veya bir program kısıtlamasını ihlal etmeden bir aktivitenin erken başlangıç tarihinden itibaren ertelenebileceği süre miktarıdır.

Toplam Bolluk = LS - ES veya LF - EF

1.3. Kaynak Tahsisi

İnşaat projesindeki faaliyetler için kaynaklar gerçek hayatta sınırlıdır. Bu yüzden, bir inşaat projesinde kaynak israfını ve kıtlığını önlemek için inşaat proje yönetimi için kaynak tahsisi büyük önem taşımaktadır.

İnşaat projesi gibi bir proje, bir faaliyetler ağından oluşur ve ağdaki bir düğüm, bir faaliyete karşılık gelir. Bir projedeki her faaliyetin karşılık gelen bir süresi vardır ve ayrıca bunlarla birlikte kendini yürütmek için emek veya malzeme gibi belirli miktarda kaynağa ihtiyaç duyar. Faaliyet süresi genellikle planlama birimleri olarak adlandırılan zamanın bütünleşik artışlarıyla ölçülür. Bir faaliyetin normal süresi, normal şartlar altında o faaliyeti tamamlamak için gereken süreyi ifade eder. Bazı aktiviteler için bir aktivitenin süresini normal sürenin altına düşürmek mümkündür. Bu, aktivite süresini azaltabilecek çökme yoluyla yapılır. Bu nedenle, kilitlenme süresi, para çekme işlemi gerçekleştirilirken bir faaliyeti tamamlamak için mümkün olan en kısa süreyi ifade eder. Kaynak tahsisi problemi, kaynak mevcudiyetinin bazı maksimum değerlerle sınırlandırıldığı varsayımıyla başlar ve amaç, en kısa proje süresini bulmak için mevcut kaynakları proje faaliyetlerine tahsis etmektir.

Bazen kısıtlı kaynak planlaması olarak da adlandırılan kaynak tahsisi, projenin kaçınılmaz uzantısını minimumda tutarken sınırlı sayıda kaynağın verimli bir şekilde kullanılabilmesi için proje görevlerini yeniden planlamaya çalışır. Pratik uygulamaları nedeniyle, inşaat alanında kaynak tahsisi sorunları yoğun olarak çalışılmaktadır. Bu problemleri çözmek için kullanılan erken teknikler, tam sayılı programlama, dal - sınırlı, doğrusal programlama veya dinamik programlama gibi matematiksel modeller kullanıldı [5]. Bununla birlikte, matematiksel yöntemler, boyut olarak makul olan herhangi bir gerçek yaşam projesi için hesaplama açısından izlenebilir değildir ve bu da onların pratik olmamasına sebep olur [6].

1.4. Kaynak Dengeleme ve Kaynak Tahsisi

Kaynak Tahsisi ve Kaynak Dengeleme, kaynakları planlamak için iki temel kategoridir. Kaynak tahsisi, mevcut kaynaklar üzerinde kesin sınırlamalar olduğunda kullanılır. Kaynak tahsisi teknikleri kullanılarak dengeleme yapılırken, hem kritik olmayan hem de

kritik faaliyetler, proje süresini orijinal kritik yol uzunluğunun ötesine mümkün olduğunca az uzatmak amacıyla kaydırılır. Kaynak tahsisi, sınırlı kaynakların belirli bir süre içinde nasıl tahsis edileceğine ilişkin bir kriter listesine bağlıdır. Mevcut kaynakların, belirli bir faaliyet tarafından talep edilen kaynaklarla karşılaştırılması gerekir. Kullanılabilir kaynaklar yetersiz olduğunda, gerekli kaynakları serbest bırakmak için faaliyetin yeniden planlanması gerekir. İki aktivitenin aynı anda aynı kaynakları gerektirmesi ve her iki aktiviteyi planlandığı gibi başlatmak için yeterli kaynağın olmaması durumunda, önceliği en yüksek olan aktivite önce kıt kaynağı alacaktır. Kaynak dengeleme, yeterli kaynak olduğunda kullanılır, ancak kaynak kullanımındaki dalgalanmaların dengelenmesi gerekir. Kritik yol tarafından hesaplanan proje süresi başlangıçta sabit kalır. Dengeleme işlemi, yalnızca kritik olmayan etkinliklerin kayan noktalarında kaydırılmasıyla gerçekleştirilir.

BÖLÜM 2. LİTERATÜR ÇALIŞMASI

Genetik algoritmalar (GA), optimizasyon yöntemi olarak akademide sıklıkla kullanılan yöntemlerden biridir. Sadece inşaat mühendisliği değil birçok sektör kendi alanlarındaki sorunlara çözüm bulmak için genetik algoritmaya başvurmuştur. Bilişim teknolojileri sektörü, makine mühendisliği problemleri, inşaat mühendisliği problemleri, proje planlama çalışmaları kullanılan başlıca alanlardandır.

Literatür çalışmaları incelendiğinde bazı çalışmaların planlama kalitesini artırmaya yönelik çalışmalar olduğu görülmüştür. Bazı çalışmaların da planlama optimizasyonunda genetik algoritmaların kullanımını inceleyen çalışmalar olduğu görülmüştür.

2.1. Planlama Kalitesini Arttırmaya Yönelik Çalışmalar

Proje planlama, proje yönetiminde kritik ve karmaşık aşamalardandır. Bu yüzden de akademik alana çokça incelenen bir konudur. Projedeki kaynaklarının paylaşımı proje süresince değişkenlik göstermektedir. Bu yüzden proje süresince kaynak kullanımını doğru tahsis etmek çok önemlidir. Bir projenin başarısı büyük oranda kaynakların doğru tahsisine bağlıdır. Bu sebeple de kaynak seviyesinin tespiti, proje yöneticilerinin rastladığı büyük zorluklar arasındadır. Tüm bunları dikkate alan Markou ve diğerleri, “Çoklu- Davranış (Multi-Attribute)” Karar Modelleri kullanıp proje kaynakları çizelgeleme ve dengelenmesinin uygulanmasını incelemişlerdir. Bu incelemeyi bir güneş parkı inşaatının proje planlamasında gösterdiği performansa bakarak yapmışlardır. Sonrasında ise ortaya çıkan sonuçlar, MS-Project programında elde edilen kaynak dengeleme sonuçlarla karşılaştırılmıştır [7].

Unegbu ve diğerlerinin yaptığı bu çalışma, Nijerya'daki inşaat sektörü için inşaat projelerinin proje performans ölçütleri ile proje yönetimi uygulamaları arasındaki ilişkiyi araştırır. Anket araştırma yöntemi, veri toplama için birincil araç olarak anket kullanılarak benimsenmiştir. Müteahhitlere, müşterilere ve danışmanlara iki yüz elli anket dağıtılır ve

bunlardan toplam iki yüz yirmi bir geçerli yanıt alınır. Çalışmanın sonuçlarına göre, müşteri memnuniyeti ile proje başarısı arasında yüksek bir ilişki olduğu tespit edilir [8].

Gelişmekte olan dünyada, sistematik inşaat sektörü bilgi yönetiminin eksikliği ortada olan bir sorundur. Bu göz önüne alındığında performans artışını iyileştirmek için organizasyonel ve operasyonel süreç iyileştirme değerlerinin daha fazla anlaşılması gerekmektedir. Buna karşılık, Hui Yap diğerlerinin yaptığı bu araştırma, inşaat tabanlı organizasyonlar tarafından kullanılan Bilgi Yönetimi (BY) uygulamalarının ve BY sağlayan araç ve tekniklerin yararlılığını değerlendirir. Malezya'da yapılan bir saha araştırmasına dayanan değişkenlerin algılanan önemi, göreceli önem indeksi (Relative importance index - RII) tekniği kullanılarak önceliklendirilir. Cevaplayıcı gruplar arasındaki uyumun derecesini değerlendirmek için Spearman's rank korelasyon testleri kullanılır. Ardından, bir faktör analizi, proje teslimi için BY faydalarının altında yatan beş ana boyutu ortaya çıkarılır. Konvansiyonel teknikler, inşaat uygulamaları için BT (Bilgi Teknolojisi) tabanlı araçlardan daha etkili olarak derecelendirilir. Faydalar ile araçlar ve teknikler arasındaki ilişki de analiz edilir. Bu belge, inşaat proje teslimini iyileştirmek için KM değerine anlamlı bir katkıda bulunur [9].

Warren Liao ve diğerlerinin yazdığı bu makale, bir inşaat veya mühendislik projesinin ömrü boyunca karşılaşılan problemleri ele almak için meta-sezgisel yöntemler kullanan önceki çalışmalarını ayrıntılı bir şekilde incelemiştir. İnceleme, kısıtlı veya kısıtlanmamış kaynakların verimli kullanımı yoluyla maliyet ve zamanı optimize etme amaçlarıyla, mühendislik, maliyet tahmini, planlama, çizelgeleme ve proje operasyonlarının izlenmesi ve kontrolünü içeren kritik problemler etrafında düzenlenmiştir. Meta - sezgisel yöntemler vurgulanmıştır çünkü bunlar özellikle genel optimumun bilinmediği veya hesaplama açısından elde edilmesinin çok masraflı olduğu durumlarda yaklaşık çözümler yeterince iyi olduğunda uygundur. İncelemeye dayanarak, bu araştırma alanı ve pratik kullanım potansiyelleri hakkındaki görüşlerini sunmaktadırlar. Daha fazla ilerleme yapılması gereken proje yönetimi ile ilgili konuların belirlenmesi için de bir girişimde bulunmaktadır [10].

İnşaat projelerindeki karmaşıklık ve belirsizlikten ötürü, proje yönetimi iyileştirmesini gerçekleştirmek için birçok farklı hedefin aynı anda dikkate alınması gerekir. Progresif teknolojilerin hızla gelişmesiyle birlikte, proje yönetimi problemlerinin çok amaçlı

optimizasyon (Multiple Objective Optimization - MOO) perspektifinden çözülmesi büyük ilgi görüp, bir trend haline gelmiştir. İnşaat endüstrisindeki sorunları çözmek için MOO'nun benimsenmesini daha iyi teşvik etmek amacıyla, Guo ve Zhang'ın yazdığı bu makale mevcut durumu anlamak için hem bilimsel hem de nitel analiz altında sistematik bir inceleme sunar ve proje iyileştirme için MOO'nun gelecekteki araştırma eğilimlerini tartışır. Sistematik incelemeye dayanarak, daha iyi proje iyileştirme için MOO'nun geniş çapta benimsenmesini teşvik etmenin zorlukları, yani dinamik özelliklerle uyumsuzluk, girdi - çıktı ilişkisinin belirsizliği ve ileri proje yönetiminde düşük etkileşim belirlenir. Bu zorlukları ele almak için üç potansiyel yön, uyarlanabilir optimizasyon, yorumlanabilir haritalama ve etkileşimli optimizasyon tartışılmaktadır [11].

Gecikmeler ve çatışmalar (Delays and Conflicts - D&C'ler) inşaat projelerinin performansını olumsuz yönde etkilemektedir. Bu sorun birçok çalışmada ele alınmıştır, fakat aralarındaki ilişki hala belirsizlik içerisinde. Bu nedenle, Tariq ve Gardezi'nin yaptığı bu çalışma bu ilişkiyi ortaya çıkarmayı amaçlamaktadır. D&C'lerin önceki sıralamasını belirlemek ve küresel bir sıralama geliştirmek ve karşılıklı ilişkilerin diğer gizli yönlerini keşfetmek için literatür gözden geçirilmiştir. Veriler, küresel bir perspektifte D&C'lerin ilk beş nedeninin ve aralarındaki ortak noktaların belirlenmesiyle sonuçlanan Jenks optimizasyonu yoluyla sınıflandırılmıştır. D&C'lerin tanımlanan ilk beş küresel nedeni arasında, yaygın nedeni tespit edilmiştir [2].

2.2. Planlama Optimizasyonu İçin Genetik Algoritmaların Kullanımı

Kaynak tahsisi ve dengeleme, inşaat projelerinin karmaşık doğasından dolayı inşaat proje yönetiminde en büyük zorluklardan biridir. CPM ve Program Değerlendirme ve Gözden Geçirme Tekniği (Program Evaluation and Review Technique - PERT), inşaat projelerinin programlanması için en yaygın kullanılan yöntemlerdir. Ancak, kaynak kullanım profilindeki istenmeyen dalgalanmaları en aza indiremez. Bu sorunu çözmeyi hedefleyen Hussain ve diğerlerinin yaptığı bu çalışmada, GA'nın bir uygulaması, MATLAB yazılımı kullanılarak kaynak tahsisi ve dengelemenin optimizasyonu için bir model geliştirilir. Problem, birçok nesil kullanılarak yürütülür. Kaynak tahsisi ve tesviye öncesi toplam proje 32 günken, kaynak sınırlaması kısıtlamalarını karşılamak için, kaynak tahsisi ve tesviye sonrası projenin süresi 54 gündür [12].

Program gecikmesinin toplam proje süresi üzerindeki etkisi tartışılmaya devam etmektedir. Mevcut gecikme analizi yöntemlerinin hiçbiri, bir varsayım unsuru, öznel değerlendirme ve teorik projeksiyon içermesi nedeniyle kusursuz değildir. Windows tabanlı gecikme analizi yöntemleri, inşaat programı gecikmelerini belirleme ve ölçmede mükemmeldir. Mevcut pencere tabanlı gecikme analizi yöntemlerindeki olası sorunları tanımlayan önceki bir çalışmaya dayanarak, Yang ve Kao'nun yaptığı bu çalışma, etki tabanlı gecikme analizi yöntemi (Effect Based Delay Analysis Method - EDAM yöntemi) olarak adlandırılan yenilikçi bir pencere tabanlı gecikme analizi yöntemi önermektedir. EDAM yöntemi, çıkarılan pencereleri kullanarak gecikme analizi yapar ve gecikmelerin kritik yol üzerindeki etkilerini dikkate alıp gecikme etkilerini belirler. EDAM yöntemi, varsayımsal durumlara uygulanmasına ve diğer yöntemlerle karşılaştırmalarına göre, gecikme analizinde etkindir ve eş zamanlı gecikmeleri çözmede ve kısaltılmış çizelgeyi belirlemede etkilidir. Önerilen EDAM yöntemi, inşaat projeleri için program gecikme analizi için iyi bir alternatiftir [13].

CPM, proje yönetimi için ağ tabanlı bir yaklaşımdır. Bu yöntem, tüm projenin tamamlanma süresinin kısaltılabilmesi için kısaltılması gereken kritik yolu bulmamızı sağlayan en uzun yolu tanımlar. Ancak, CPM'deki belirsizliği dikkate almak kolay değildir. Takakura ve diğerlerinin yaptığı bu çalışmada, görev sürelerinin tarihsel işlem verilerinden elde edilen ayırık histogramlar olarak ifade edildiği, kritik noktalarda görev sürelerini iyileştirerek tüm görevlerin belirli bir tamamlanma süresi içinde tamamlanma olasılığını maksimize eden stokastik CPM problemleri için bir optimizasyon problemi ele alınmıştır.

Nüfus artışı ve teknolojik ilerleme, enerji talebinde artışa yol açmıştır. Fosil yakıtlar şu anda, ana enerji kaynakları olmakla birlikte sınırlıdır ve yanmaları çevre kirliliğine neden olmaktadır. Bu nedenle alternatif enerji kaynakları düşünülmelidir. Bu amaçla biyogaz, organik atıklardan üretilen uygun bir yenilenebilir enerji seçeneği olabilir. Buna örnek olarak hayvan gübresi ve kırsal atıklar verilebilir. Bununla birlikte, büyük ölçekli bir biyogaz tesisi inşaat projesini yönetmek, değişen sürelerde birçok koordineli faaliyet gerektirir ve çok sayıda bağımlılık içerir. Bu nedenle, Zareei'nin yaptığı bu çalışma, CPM kullanılarak biyogaz tesisi inşaat projesinin analizi için planlama ve çizelgeleme uygulamasına odaklanmaktadır. Sonuçlar, İran'da 50 m³'lük sabit kubbeli bir biyogaz tesisi inşa etmenin minimum tamamlanma süresinin, proje adımlarını

ertelememesi halinde 38 hafta olacağını ortaya koymuştur. Ayrıca, faaliyetler arasındaki ilişkileri göstermek ve projenin ilerlemesini izlemek için bir proje ağı önerilmiştir [14].

Geleneksel analitik ve sezgisel yaklaşımlar, inşaat kaynak dengeleme problemlerini çözerken yeterince verimli değildir. Leu ve diğerlerinin yaptığı bu çalışmada geleneksel inşaat kaynak dengeleme problemlerinin üstesinden gelmek için genetik algoritma kullanılmıştır. Önerilen algoritmanın, kaynak dengeleme hedefine tabii olan faaliyetlerin başlangıç ve bitiş tarihlerinin yanı sıra çoklu inşaat kaynaklarının optimal veya optimale yakın kombinasyonunu etkin bir şekilde sağlayabildiği tespit edilmiştir [15].

Kaynak kısıtlı proje çizelgeleme problemi (Resource Constraint Project Scheduling Problems - RCPS), inşaat ve birçok endüstriyel disiplinde yaygın olarak kullanılan proje çizelgeleme problemlerinden biridir. Problemin zorluğu, uygulanabilir uzayda çözümler bulmak için bazı uygun arama mekanizmaları tasarlamaktır. Hua ve diğerlerinin yaptığı bu çalışmada zaman penceresi ayrıştırmasına dayalı geliştirilmiş bir genetik algoritma önerilmiştir. Popülasyon çeşitliliğini artırmak için üç türetme yöntemi uygulanmaktadır. Örnekleme sayısı tahsis stratejisi ve yıkıcı alt sınırların kullanılması, arama verimliliğini artırır. Hesaplama deneyleri, önerilen yaklaşımın yalnızca ayrıştırma mekanizmasını kullanmaktan daha etkili olduğunu ve iki gerçek yaşam vakasını çözmeye rekabetçi olduğunu göstermektedir. Bu araştırma, arama alt uzaylarının sürekli olarak değiştirilmesinin, gelecekte diğer evrimsel algoritmaları kullanarak RCPS'yi incelemek için yararlı olabilecek potansiyel avantajlara sahip olduğunu göstermektedir. Son olarak her bir birey için örnekleme sürelerini esnek bir şekilde belirlemek için makine öğrenimi yöntemleri kullanılarak daha iyi bazı sonuçlar elde edilebileceği kararına varılmıştır [16].

Kaynak yönetimi, bir projenin zamanında hesaplanan maliyetle tamamlanmasını sağlar. Ancak buna rağmen kaynaklar kısıtlıdır ve proje faaliyetlerinde kullanımları zaman çizelgesinde çatışmalara yol açar. Kaynak dengeleme problemleri, kaynak tüketiminin mümkün olduğu kadar verimli hale nasıl getirileceğini düşünür. Tienda ve diğerlerinin yazdığı bu makale, "Kaynak Dengeleme Problemi" için uyarlanabilir bir genetik algoritma sunmuştur [17].

PERT/CPM ağ teknikleri, gerekli tüm kaynakların mevcut olacağı varsayımına dayanmaktadır. Kaynakların kısıtlılığı, proje gecikmelerinin olağan bir nedenidir. RCPS çözümünü için diğer yöntemler uygulanmıştır. Bu yöntemlerin amacı, hem öncelik hem de

kaynak kısıtlamalarını göz önünde bulundurarak proje süresini en aza indirmektir. Proje yönetimi yazılım paketleri, kaynak seviyelendirmeyi kullanarak kaynak çakışmalarını çözer. Kastor ve Sirakoulis'un yazdığı bu makale, vaka çalışmaları olarak iki gerçek inşaat projesini seviyelendirirken sonuçları karşılaştırmıştır [18].

İnşaat projesindeki faaliyetler için kaynaklar gerçek hayatta sınırlıdır. Bu nedenle, bir inşaat projesinde kaynak israfını ve kıtlığını önlemek için inşaat proje yönetimi için kaynak tahsisi büyük önem taşımaktadır. Liu ve diğerlerinin yazdığı bu makale, kaynak tahsisi için bir genetik algoritma modeli sunmaktadır. Geleneksel çaprazlama yöntemleriyle karşılaştırıldığında, önerilen model, yasa dışı kromozomların üretilmesini önlemek için yeni bir çaprazlama operatörü geliştirmektedir. Model, kaynak tahsisi probleminde etkin bir şekilde optimum çözümü sağlayabilir. Önerilen yaklaşımın performansını göstermek için açıklayıcı bir örnek sunulmuştur [19].

İnşaat sektöründe, proje yöneticilerinin (Project Managers - PM) kaynak tahsisi düzenleme problemini farklı bakış açılarıyla ele alması büyük önem taşımaktadır. Bu durumda, inşaatta kaynakların yönetimi bir zorluk haline gelir. Genel olarak konuşursak, inşaatta zaman, maliyet ve enerji tüketimi dahil olmak üzere birbiriyle çelişen optimize edilmesi gereken birçok hedef vardır. He ve diğerlerinin yazdığı bu makale, bu hedefler arasında en iyi değiş tokuş ilişkisini elde etmek için kuantum genetik algoritmasına (Quantum Genetic Algorithms - QGA) dayalı çok amaçlı bir optimizasyon çerçevesi önermiştir. Her bir inşaat faaliyetinde tahsis edilen inşaat kaynakları, yürütme süresini, maliyetini ve enerji tüketimini belirleyecektir ve nihayet inşaat faaliyetleri arasındaki korelasyonlar nedeniyle projenin karmaşık bir zaman - maliyet - enerji tüketimi değiş tokuş çerçevesi oluşturulmuştur. Zaman, maliyet ve enerji tüketimi arasındaki en iyi kombinasyonu ve bu durum altında en uygun kaynak düzenleme şemasını bulmak için QGA yapılmıştır. Bu kaynak tahsisi modunun rasyonelliğini kontrol etmek için inşaat süreci BIM'de simüle edilmiştir. Önerilen modelin uygulanmasını göstermek için Çin'deki bir endüstriyel fabrika ofis binası örnek olarak sunulmuştur. Sonuçlar, sunulan yöntemin etkili bir şekilde maliyeti %7, zamanı %17 ve enerji tüketimini %21 oranında azaltabileceği gösterilmiştir [20].

Sınırlı kaynak tahsisi sorunu, birçok inşaat projesinde, yüklenicinin kullanabileceği kaynakların miktarı üzerinde farklı sınırlamalar olduğunda ortaya çıkar. Çizelgeleme

hedefi, kaynak gereksinimleri mevcut kaynak miktarını aşmayana kadar faaliyetleri kaydırarak kaynak çatışmasını çözerken proje süresini minimumda tutmaktır. Amaç, mevcut kaynakları kullanarak proje süresini en aza indirmek ve mevcut ekipman ve işgücünün kullanımını arttırmaktır. Sınırlı kaynak yönetimi için çok sayıda bilgisayar paketi geliştirilmiştir. Bu paketlerden bazıları, kritik yol hesaplamalarından elde edilen ölçümlere dayalı olarak proje faaliyetlerine öncelikler atar. Khattab ve Soyland'ın yaptığı bu çalışmada, ticari bilgisayar paketleri tarafından alınan yaklaşım ile bazı buluşsal tekniklerde geliştirilen bazı öncelik sıralama prosedürleri arasındaki temel farkı göstermiştir [21].



BÖLÜM 3. PROJE YÖNETİMİNDE KAYNAK DENGELEME

Proje yönetimi açısından, kaynak dengeleme yapma ihtiyacının birkaç nedeni vardır. Birinci neden, inşaat kaynaklarının fiziksel sınırlarını karşılama ihtiyacıdır. İkincisi, kaynak taleplerindeki günlük dalgalanmalardan kaçınma ihtiyacı ve üçüncüsü, inşaat kaynakları için eşit bir uygulama akışı sağlama ihtiyacıdır [22]. Kaynak dengeleme problemleri, pratik uygulamaları nedeniyle inşaat ve imalat endüstrilerinde yoğun olarak çalışılmaktadır.

Kaynak dengeleme problemlerini çözmeye yönelik ilk girişimler iki kategoride sınıflandırılabilir: analitik modeller ve buluşsal modeller [23]. Kaynak dengeleme problemlerini çözmek için birkaç analitik veya sezgisel model üretilir. Analitik modellerin birincil dezavantajı, büyük ve karmaşık bir sorunu etkin bir şekilde çözülememesidir. Sezgisel modeller probleme bağlıdır, bu nedenle temel kuralları tüm inşaat durumlarına eşit olarak uygulanamamaktadır. Ayrıca, sezgisel modellerden elde edilen çözümler genellikle ideal değildir [24].

Genetik algoritmaları kullanan yeni bir yaklaşım, bu dezavantajların üstesinden gelmektedir. GA tabanlı kaynak dengeleme çözümleri sistemi (Genetic Algorithms Resource Leveling System - GARLS), belirli bir proje süresi ile proje kaynak kullanım varyasyonunu en aza indirme hedefine ulaşmak için birden fazla mevcut inşaat kaynağını seviyelendirir.

Kaynak dengelemenin amacı, orijinal proje süresi sabit kalırken eşit bir kaynak uygulaması akışı bulmak amacıyla inşaat kaynaklarını düzeltmektir. Kaynak dengeleme modellerinin popüler teknikleri iki alanda kategorize edilebilir: analitik yöntemler ve sezgisel yöntemler. Kaynak dengeleme problemlerini çözmeye yönelik ilk girişimler, tam sayılı doğrusal programlama ve kapsamlı numaralandırma dahil olmak üzere matematiksel modeller kullandı. Easa, kaynak dengeleme problemini çözmek için tam sayılı programlama tekniklerini kullandı [23].

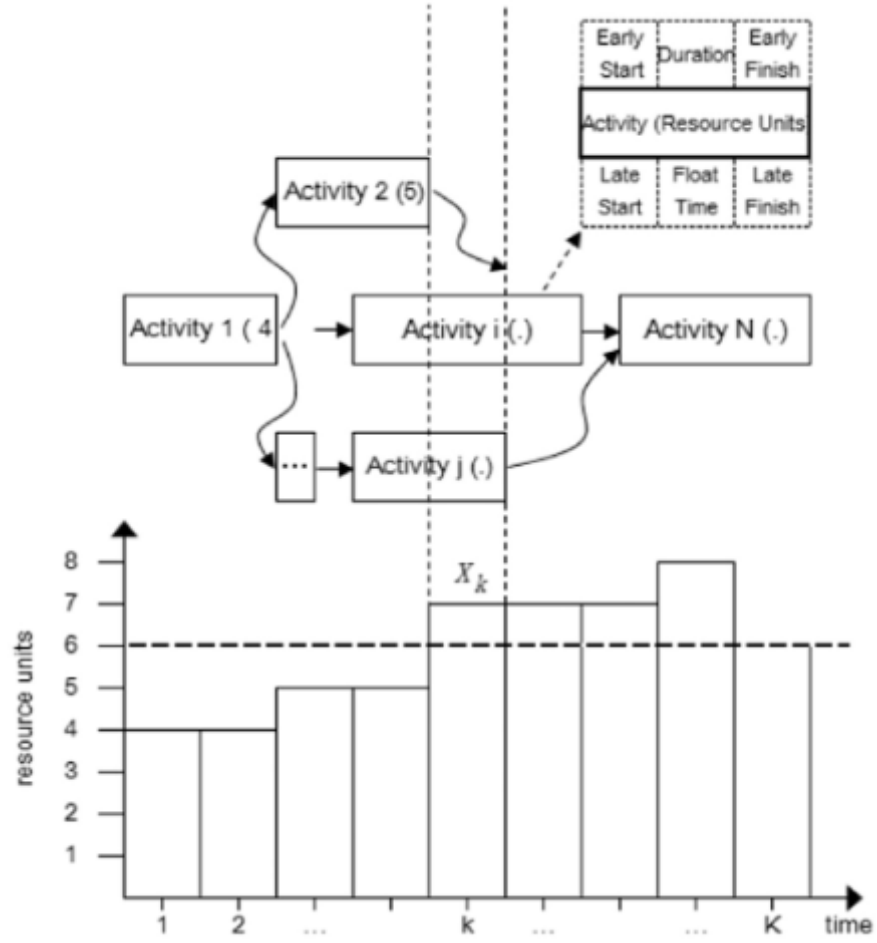
Bununla birlikte, kaynak dengeleme probleminin, özellikle büyük ölçekli problemler için bir “kombinatoriyal patlama” olgusu vardır. Patlama probleminden kaçınmak için problemlerin çözümünde çoğunlukla sezgisel kurallar kullanılmıştır. Kaynak dengeleme sezgisel prosedürleri için büyük çabalar, uygulanabilir çözümler üreten kuralların geliştirilmesine harcanmıştır. Bugüne kadar, proje çizelgeleme problemlerini çözmek için birçok sezgisel çizelgeleme kuralı önerilmiştir. Ağ yapılarının ve kaynaklarının çeşitliliği nedeniyle, tek bir sezgisel yöntem, tüm kaynak dengeleme problemleri için her zaman en iyi çözümü üretemez.

GARLS algoritmalarını kurmak için bazı varsayımlar yapılmıştır: 1. ağ diyagramındaki aktiviteler bölünemez; 2. aktivite süresi ve ağ mantığının sabit olduğu varsayılır; ve 3. kaynaklara olan talepler bir faaliyetin süresi boyunca sabit tutulur.

Kaynak tahsisi, düğümlerin faaliyetleri temsil ettiği ve yayların öncelik ilişkilerini temsil ettiği proje yönetiminde kullanıldığı gibi bir ağ diyagramı kullanılarak analiz edilebilir. Faaliyetlerin gerçekleştirilmesi dönemler ve kaynak birimleri alır ve ardından ufkun kaynak seviyelendirme şeması, ağ diyagramı ile birlikte Şekil 3.1'deki gibi görüntülenebilir.

3.1. Kaynak Dengelemenin Formülasyonu

Kaynak tahsisi, düğümlerin faaliyetleri temsil ettiği ve yayların öncelik ilişkilerini temsil ettiği proje yönetiminde kullanıldığı gibi bir ağ diyagramı kullanılarak analiz edilebilir. Faaliyetlerin gerçekleştirilmesi dönemler ve kaynak birimleri alır ve ardından ufkun kaynak dengeleme şeması, ağ diyagramı ile birlikte Şekil 3.1'deki gibi görüntülenebilir.



Şekil 3.1 : Ağ Diyagramı ve Kaynak Dengeleme Şeması [25]

3.1.1. Aktivitelerin planlanması

Öncelikle proje çıktılarına göre faaliyetler (işler veya görevler) tanımlanır [25].

Aktivite seti, A ,

$$A = \{a_1, a_2, \dots, a_N\} \quad (3.1)$$

Bir ağdaki toplam aktivite sayısını belirtmek için N kullanılır. Faaliyet bir kez başladıktan sonra kesintiye uğramadan tamamlanmalıdır. Her aktivitenin başlangıç zamanları seti, T ,

$$T = \{t_1, t_2, \dots, t_N\} \quad (3.2)$$

Her aktivitenin süre seti, D ,

$$D = \{d_1, d_2, \dots, d_N\} \quad (3.3)$$

Faaliyetler arasındaki bağlar şu şekilde ifade edilebilir:

$$t_i + d_i \leq t_j, i, j = 1, 2, \dots, N \quad (3.4)$$

Bu, j faaliyetinin ancak i faaliyeti bittiğinde başlayabileceği anlamına gelir. CPM, planlanan faaliyetlerin mantıksal bitiş noktalarına veya projenin sonuna kadar olan en uzun yolunu hesaplar. Erken başlama zamanları seti T^{es} ve geç başlama zamanları seti T^{ls} ,

$$T^{es} = \{t_1^{es}, t_2^{es}, \dots, t_N^{es}\} \quad (3.5)$$

$$T^{ls} = \{t_1^{ls}, t_2^{ls}, \dots, t_N^{ls}\} \quad (3.6)$$

Her aktivite sınırlıdır:

$$t_i^{es} \leq t_i \leq t_i^{ls}, i, j = 1, 2, \dots, N \quad (3.7)$$

Bu varsayımlara dayanarak, faaliyetler zaman, maliyet ve kaynak kullanımı dahil olmak üzere farklı proje hedeflerine göre birçok farklı şekilde planlanabilir.

3.1.2. Kaynak dengeleme problemleri

Her bir faaliyetin yenilenebilir kaynak birimleri seti (örneğin; insan gücü, makineler ve boşluklar), R ,

$$R = \{r_1, r_2, \dots, r_N\} \quad (3.8)$$

Her günün kaynak tüketimi, x_k ,

$$X_k = \sum_{i \in I_k} r_i, k = 1, 2, \dots, K \quad (3.9)$$

Veya,

$$X_k = \sum_{i=1}^N c_i r_i \begin{cases} c_i = 1, \text{ if } t_i \leq k \leq t_i + d_i - 1 \\ c_i = 0, \text{ if } k \leq t_i \text{ veya } k \geq t_i + d_i \end{cases} \quad (3.10)$$

Şekil 3.1.'de dikey paralel noktalı çizgilerle gösterildiği gibi, burada K proje süresi zamanıdır ve I_k aktivite k . günde çalıştığında aktivite setidir.

Yatay kesikli çizgi, ortalama günlük kaynak kullanımını temsil eder. Açıkça, kaynak kullanımındaki büyük farklılıklarla ilişkili zorluklardan kaçınmak için proje çizelgelemesinde kaynak dengeleme gereklidir. Gerçekleştirilecek projedeki mantık ve kısıtlamalara göre T 'yi planlamak için çok sayıda farklı olasılık vardır. Bu çizelgelerin her biri, kaynak tüketiminin verimliliği açısından önemli farklılıklara sahiptir.

Denk. (3.9) veya Denk. (3.10) orijinal olarak tek tip kaynak dengeleme için geliştirilmiş ve daha sonra bu aktiviteleri tamamlamak için çeşitli kaynak türleri olduğu anlamına gelen çoklu tip kaynak dengeleme genişletilmiştir.

3.1.3. Kaynak Dengelemenin Amaç Fonksiyonları

Kaynak kullanımlarındaki varyasyonların nasıl ölçüldüğüne bağlı olarak, farklı amaç fonksiyonları ele alınabilmektedir. t zamanı ayrık ise, ayrık Kaynak Dengeleme Problemleri (Resource Leveling Problems - RLP), 'nin amaç fonksiyonu şu şekilde yazılabilir:

(3.11)

$$f(x) = \sum_{m=1}^M \sum_{k=1}^K f_m(x_{mk})$$

Burada amaç fonksiyonu $f_m(x_{mk})$, k. gün boyunca kaynak değişimini ölçer ve X_{mk} , m kaynak türü ($m = 1, 2, \dots, M$) için her günün kaynak tüketimidir. Aşağıdakiler de dahil olmak üzere amaç fonksiyonları için çeşitli ifadeler vardır.

$$f_m(x_{mk}) = c_m x_{mk}^2 \quad (3.12)$$

Burada c_m , kaynak türü m'deki dalgalanmaların ceza maliyetini temsil eden, kaynak türü m'nin ağırlık katsayısıdır. Denk. (3.12.), kaynak birimlerinin karesi olarak yazılan denklemin amaç fonksiyonunu temsil eder. Bir sonraki ifade

$$f_m(x_{mk}) = c_m (x_{mk} - x_m)^2 \quad (3.13)$$

Ve bu, öngörülen kaynak kullanım eşiğinden sapma olarak yazılan denklemin amaç fonksiyonunu temsil eder. Burada x_m genellikle ortalama kaynak kullanımına eşit olarak seçilir. Ayrıca $(x_{mk} - x_m)^+$, istenen kaynak kullanımından pozitif sapmayı kapatmak için kullanılır [23], [26]. Bir sonraki ifade

$$f_m(x_{mk}) = c_m (x_{mk+1} - x_{mk})^2 \quad (3.14)$$

Ve bu, kaynak kullanımının dönemden döneme değişimi olan komşu kaynak birimlerinin dalgalanması olarak yazılan RLP'nin amaç fonksiyonunu temsil eder. Son ifade

$$f_m(x_{mk}) = -c_m \frac{x_{mk}}{\sum_{k=1}^K x_{mk}} \ln \frac{x_{mk}}{\sum_{k=1}^K x_{mk}} \quad (3.15)$$

kaynak tabanlı entropi [olarak yazılan denklemin amaç fonksiyonunu temsil eder ve biz buna basitçe kaynak seviyelendirme entropisi diyoruz. Entropi formülasyonu, günlük kaynak değerlerinin atanan toplam kaynak değerlerine oranıdır.

Halihazırda, mevcut tüm amaç fonksiyonları, ilgili optimizasyon teknikleri ile ayrı ayrı tartışılmaktadır, ancak birbirleri arasında performans karşılaştırmaları bulunmamaktadır.

3.1.4. Optimizasyon Modeli

RLP, kaynak kullanımının varyasyonunu en aza indirmeyi amaçlayan, proje son tarihi ve önceliği ile sınırlandırılan her bir faaliyetin başlangıç zamanını belirten proje takviminin geliştirilmesini içerir. RLP'nin optimizasyon modeli, tüm proje yaşam döngüsü boyunca kaynak kullanımındaki değişiklikten kaynaklanan maliyetlerin toplamını en aza indirir. Yukarıdaki tartışmalara dayanarak, RLP optimizasyon formülasyonları şu şekilde yazılabilir:

$$\min \sum_{m=1}^M \sum_{k=1}^K f_m(x_{mk}) \quad (3.16)$$

$$t_i^e \leq t_i \leq t_i^1, i, j = 1, 2, \dots, N \quad (3.17)$$

$$t_i + d_i \leq t_j, i, j = 1, 2, \dots, N \quad (3.18)$$

Burada x_{mk} RLP ile hesaplanır. Burada $f_m(x_{mk})$, gerçek durumda denklem (3.12), (3.13), (3.14) veya (3.15) ile değiştirilmelidir. denklem (3.17) ve (3.18) faaliyet önceliği ve proje süresi ile sınırlandırılmış minimizasyonun aranması anlamına gelirken, (3.14) diğerlerinden farklıdır ve maksimizasyonun dikkate alınması gerekir.

3.2. Normalleştirilmiş Entropiye Dayalı Kaynak Seviyelendirme

3.2.1. Entropi

İstatistiksel mekanikte entropi, bir sistemin düzensizliğinin bir ölçüsüdür (entropi ne kadar yüksekse, düzensizlik de o kadar yüksek olur). Bu tanım, entropiyi, sistemin olası mikroskobik konfigürasyonlarının (mikro haller) sayısının doğal logaritması ile orantılı olarak tanımlar. Spesifik olarak, entropi, işgal edilme olasılığı önemli olan durumların sayısının logaritmik bir ölçüsüdür:

$$S = -k_B \sum_i p_i \log p_i \quad (3.19)$$

3.2.2. Kaynak dengelemede entropi

Entropi kavramı başlangıçta termodinamik bir terim olmasına rağmen, bilgi teorisi, psikodinamik, ekonomi ve inşaat yönetimi dahil olmak üzere diğer çalışma alanlarına uyarlanmıştır. Tek bir kaynak türü için toplam proje entropisi H olur.

$$H = - \sum_{k=1}^K \left[x_k / \sum_{k=1}^K x_k \ln \left(x_k / \sum_{k=1}^K x_k \right) \right] \quad (3.20)$$

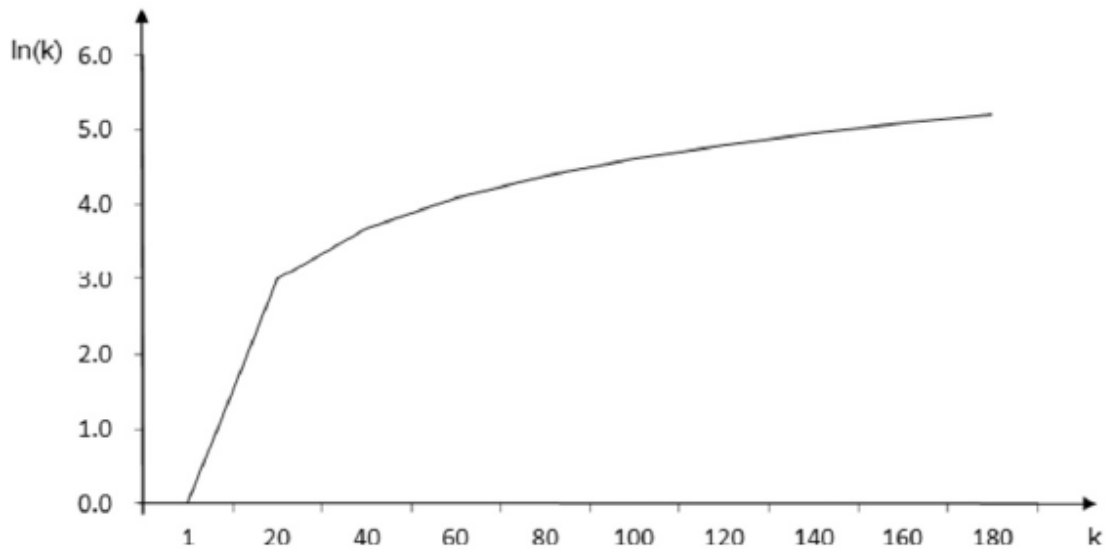
x_k , istatistiksel mekanikteki entropi tanımıyla aynı olan, k. gün boyunca projeye atanan kaynak birimlerinin sayısı olduğunda, bu nedenle daha yüksek entropi değerleri, daha iyi kaynak seviyelendirmeyi gösterir. Bu, ortalama kaynak tahsisinin projenin maksimum entropisine karşılık geldiği anlamına gelir:

$$H_{max} = - \sum_{k=1}^K \left(\frac{\sum \frac{x}{K}}{\sum x} \ln \frac{\sum \frac{x}{K}}{\sum x} \right) = \ln K \quad (3.21)$$

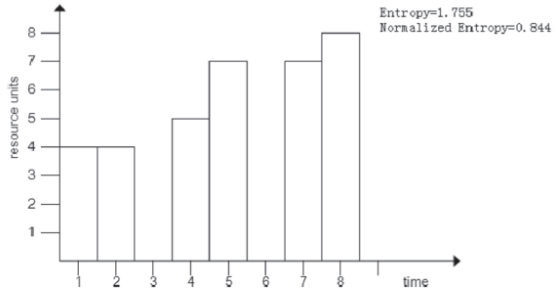
Bu nedenle, maksimum kaynak seviyelendirme entropisi, toplam kaynak birimlerinden bağımsızdır, ancak toplam proje süresi ile ilgilidir.

3.2.3. Normalleştirilmiş Kaynak Seviyelendirme Entropisi

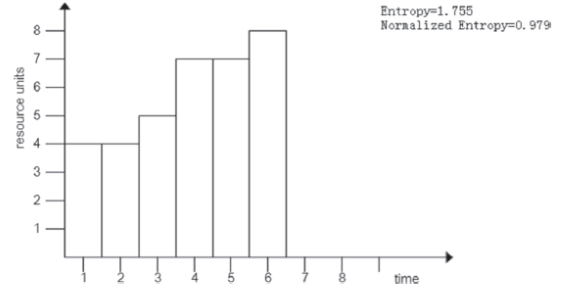
Christodoulou ve diğerlerinin yaptığı çalışmada [27] entropinin iki temel ilkesine sahip olduğuna dikkat çekmiştir: Şekil 3.2.'de gösterildiği gibi, toplamsallık ve maksimumluk. Kaynak dağılımları Şekil 3.3.'de gösterilmektedir.



Şekil 3.2 : Farklı Proje Süreleri İçin Entropi Değerleri [27]



(a)



(b)

Şekil 3.3 : (a) Kaynak Dağılımı 1 ve (b) Kaynak Dağılımı



BÖLÜM 4. GENETİK ALGORİTMA

Genetik algoritma, ilk olarak 1975 yılında Holland tarafından önerilen stokastik bir arama algoritmasıdır. Algoritma belirli bir problemi hedeflemez, ancak ortak bir çözüm çerçevesi sağlar. Bu nedenle, birçok optimizasyon problemine yaygın olarak yerleştirilmiştir. RCPSP'yi çözmek için genetik algoritmanın kullanımı veya diğer meta sezgisel algoritmalarla kombinasyonları önemli bir araştırma yönü olmaya devam etmektedir. Genetik algoritmaların temel bileşenleri, kodlama şeması, kod çözme şeması, başlatma, çaprazlama, mutasyon, seçim ve durdurma koşullarını içerir. Çaprazlama, yeni yavrular oluşturmak için iki ebeveyn bireyin kısmi yapılarının değiştirilmesi ve yeniden birleştirilmesi anlamına gelir. Çaprazlama işlemleri, genetik algoritmaların küresel arama kapasitesini önemli ölçüde geliştirebilir ve iyi yavrular üretmek için önemli araçlardır. Mutasyon işlemi, yerel optimal çözümlere düşmekten kaçınmak için tasarlanmıştır.

Doğada türler, kalıtım, çeşitlilik, hayatta kalma mücadelesi ve çevreye uyumlarına göre doğa tarafından seçilir veya yok edilir. Genetik algoritma, bu sürecin matematiksel bir temsilidir. Genetik algoritmanın temel fikri, problemin çözümünü temsil eden bir başlangıç popülasyonundan başlamaktır. Popülasyondaki bireylerin uygunluklarına göre yeni nesil popülasyonlar oluşturmak için seleksiyon, çaprazlama ve mutasyon işlemleri yapılır. Yeni popülasyonun bireyleri, yalnızca önceki neslin genlerini miras almakla kalmaz, aynı zamanda genel performansta önceki nesli geride bırakır.

4.1. Genetik Algoritmaların Avantajları

Diğer geleneksel algoritmalarla karşılaştırıldığında genetic algoritma çeşitli avantajlara sahiptir.

1. Kendi kendini organize etme
2. Kendi kendine uyarlama

3. Paralellik

Fonksiyon optimizasyonu, kombinatorial optimizasyon, üretim çizelgeleme, otomatik kontrol ve diğer alanlarda yaygın olarak kullanılmaktadır.

4.2. Genetik Algoritmaların Dezavantajları

Ancak genetik algoritmanın bazı dezavantajları da vardır [28]. Temel genetik algoritma, parametrelere güçlü bir şekilde bağımlı olan önceden ayarlanmış kontrol parametrelerini kullanır. Parametrelerin boyutu genellikle deneyim ve tekrarlanan deneylerle belirlenir. Parametreler yanlış seçildiğinde, algoritmanın performansı etkilenecektir. Genetik algoritmanın optimal çözümü elde edemeyeceği, popülasyon büyüklüğünün uygun seçimi ile ilgilidir. Popülasyonun çeşitliliğini artıran ve global optimal çözümü elde etme olasılığını artıran daha büyük ölçekli başlangıç popülasyonu seçilir. Bununla birlikte, popülasyon çok büyükse, algoritmanın hesaplama süresi artar ve hesaplama verimliliği azalır. Popülasyon büyüklüğü çok küçükse, popülasyonun çeşitliliği azalacaktır ve algoritma çok hızlı yakınsayacaktır, böylece global optimal çözüm bulunamaz.

Bu nedenle, popülasyon büyüklüğü algoritmanın yakınsaması ile büyük bir ilişkiye sahiptir. Temel genetik algoritma popülasyonun boyutunu sabitler ve erken olgunun ana nedeni, popülasyonun küresel optimal çözüm elde edilmeden önce çeşitliliğini kaybetmesi ve bu da algoritmanın yerel optimal çözüme düşmesidir.

Genetik algoritmanın yakınsaması sadece popülasyon büyüklüğü ile ilgili değildir, aynı zamanda genetik operatörlerden de etkilenir. Çaprazlama olasılığının boyutu ve mutasyon olasılığı, genetik algoritmanın yeni bireyler oluşturma hızını ve algoritmanın yakınsamasını doğrudan etkiler. Temel genetik algoritma, değişmez çaprazlama olasılığını ve mutasyon olasılığını benimser, böylece bu kontrol parametreleri, algoritmanın optimal çözümü arama yeteneğini etkileyen evrim sürecine göre kendi kendini ayarlayamaz [29],

4.3. Uyarlanabilir Genetik Operatörlü Genetik Algoritma

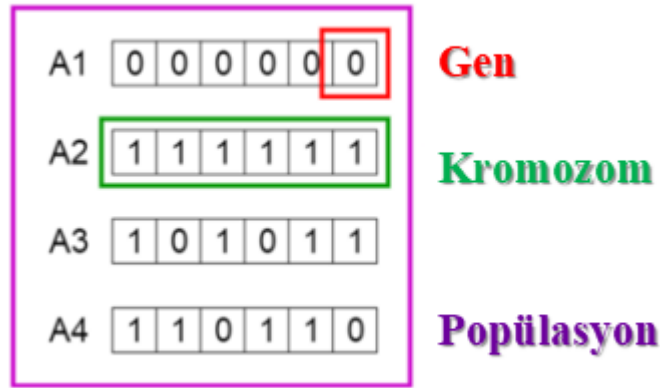
Adaptif genetik algoritmayı geliştirmenin temel yolları aşağıdaki gibi özetlenebilir.

1. Genetik algoritmanın bileşenleri veya kullanılan teknikler geliştirilir.

2. Hibrit bir genetik algoritma kullanılır.
3. Evrim süreci sırasında kontrol parametrelerini ayarlamak için dinamik uyarlamalı teknoloji kullanılır.
4. Standart olmayan genetik operatörler kullanılır.
5. Paralel genetik algoritma kullanılır.

4.4. Genetik Algoritma Hesaplama Süreci

Genetik algoritmanın hesaplama süreci genellikle beş adımdan oluşur: uygun kodlama modunun seçilmesi, popülasyonun başlatılması, bireysel uygunluğun hesaplanması, yeni nesil popülasyonun oluşturulması ve sonlandırma koşullarının belirlenmesi [30].



Şekil 4.1 : Gen, Kromozom, Popülasyon Gösterimi

Adım 1: Kodlama modu seçimi

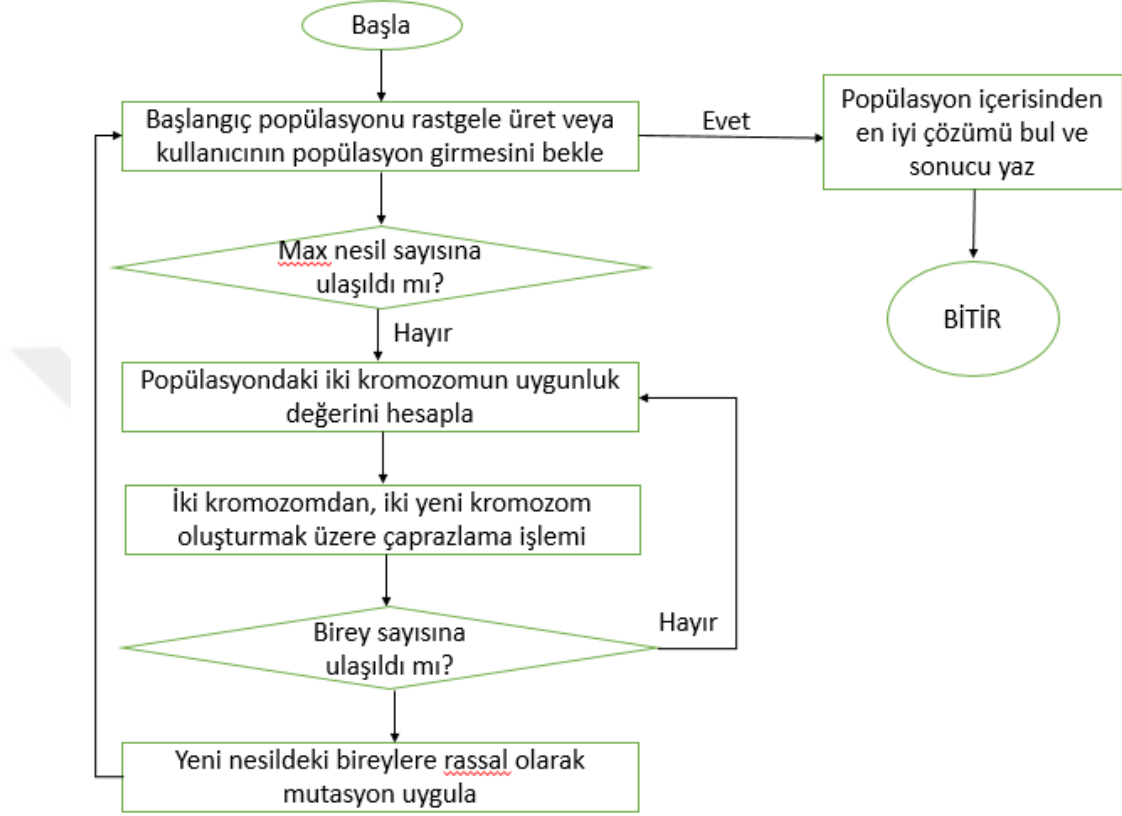
Kromozom, bileşenlerin montaj sırasını temsil eder ve kromozomdaki genler, bileşenlerin benzersiz kimlik numarasını temsil eder. Bu nedenle, ortak doğal sayı kodlaması benimsenmiştir.

Adım 2: Popülasyonu başlatma

Popülasyon başlatma, birkaç kromozom üretmek, popülasyon numarasını, yineleme sürelerini, mutasyon olasılığını ve genetik algoritmanın çaprazlama olasılığını vermektir. Popülasyon boyutu ve iterasyon sayısı genellikle kromozom uzunluğu ile orantılıdır. Bu

kromozomlardaki genler arasında katı kısıtlamalar vardır. Bu kısıtlamalar, üretilen kromozomların uygulanabilir olmasını sağlayabilir.

Genetik algoritmaya ait örnek bir akış diyagramı aşağıda verilmiştir (Şekil 4.2.).



Şekil 4.2 : Genetik Algoritma Akış Şeması [31]

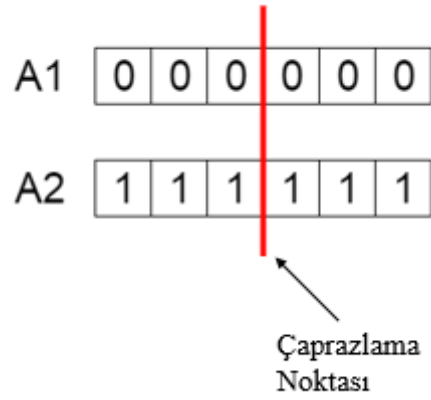
Adım 3: Gelecek nesil bireyleri oluşturma

Yeni nesil bireyler, çaprazlama ve mutasyon işlemleriyle üretilir.

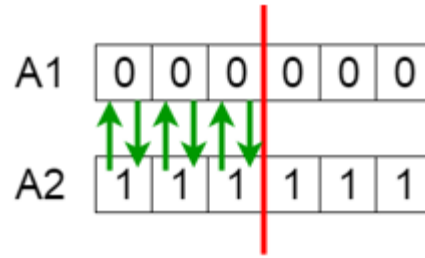
1.Aşama: Çaprazlama

Çaprazlama, iki kromozom üzerinde aynı konumdaki gen parçalarının değiş tokuşudur. Optimizasyon için çaprazlama işleminin (Şekil 4.3.) aşağıdaki koşulları karşılaması gerekir:

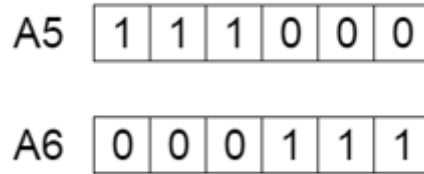
- Ana kromozomun mükemmel genini büyük ölçüde miras alır;
- Yavru kromozomlardaki tüm gen değerleri birbiriyle tekrarlanamaz;
- Yavru kromozomlar katı kısıtlamaları karşılamalıdır.



Şekil 4.3 : Çaprazlama



Şekil 4.4 : Ebeveynler Arasında Gen Alışverişi



Şekil 4.5 : Yeni Yavru

2.Aşama: Mutasyon işlemi

Kromozomdaki tüm gen değerleri birbiriyle tekrarlanamadığı için kromozomdaki iki genin rastgele değiş tokuş edilmesiyle mutasyon işlemi gerçekleştirilebilir. Bu yöntemde değişim mutasyonu da denir. Mutasyon işlemi, mutasyona uğramak için iki noktalı karşılıklılığı benimser.

A5

1	1	1	0	0	0
---	---	---	---	---	---

Şekil 4.6 : Mutasyon Öncesi

A5

1	1	0	1	1	0
---	---	---	---	---	---

Şekil 4.7 : Mutasyon Sonrası

Adım 4: Kromozomların kodunun çözülmesi

Bileşenlerin kodunun çözülmesiyle elde edilir.

Adım 5: Uygunluk değerinin hesaplanması

“Fitness” değeri hesaplanır.

Adım 6: Kromozom seçimi

Kromozom seçimi, önceki nesilden daha iyi bireyleri seçip bir sonraki nesle aktarmaktır. Seçim teknolojisi, genetik programlamanın verimliliği üzerinde büyük bir etkiye sahiptir. Yalnızca hızlı bir şekilde yakınsayan değil, aynı zamanda popülasyon çeşitliliğini de koruyan bir seçim yöntemine ihtiyaç vardır. Bu sefer rulet çarkı seçimi benimsenmiştir. Rulet çarkı seçimi, genel olarak bir GA popülasyonundan seçkin bireyleri seçmek için kullanılır. Bir üyenin ebeveyn olarak seçilme şansı, döl üretme şansı, uygunluğu (objektif fonksiyon değeri) ile doğrusal orantılıdır. Temel fikri, bir bireyin seçilme olasılığının, uygunluk fonksiyonunun değeri ile doğru orantılı olmasıdır.

Adım 7: Yineleme sayısının belirlenmesi

Bir önceki adımın sonuçlarına göre, koşullara göre yinelemenin tamamlanıp tamamlanmadığına karar verilir. Tamamlanırsa, sonuç çıktılanır, aksi takdirde yeni nesil popülasyonu yeniden oluşturmak için 3. adıma dönülür.

BÖLÜM 5. METODOLOJİ

Mevcut proje geleneksel yöntemlerden kritik yol metodu ile çözülerek Gant Çizelgesi elde edilir. Çizelgede günlük kaynak ihtiyaçları incelendiğinde günden güne gerekli olan kaynak miktarının çok büyük farklılıklar gösterdiği tespit edilir. Özellikle bazı günlerde gerekli kaynak miktarının 24'ü aşması, kısıtlı kaynaklar altında yönetilen projelerde sıkça rastlanan bir problemdir. Tüm kaynakların belli bir değeri geçmemesi gereken bir proje için kaynak dengelemenin mevcut yöntemlerle yapılması çok uzun zaman alabilmektedir. Aynı zamanda da kaynak dengelemenin kesin ve optimum sonuç veren bir yöntemi literatürde bulunmamaktadır. Tüm bunlar dikkate alındığında, bu çalışmada kaynak dengelemenin Genetik Algoritma yöntemiyle Matlab programında çözülerek en kısa sürede en optimum sonuca ulaşılması amaçlanmaktadır.

Geleneksel yöntemlerden Kutu Diyagramı kritik yol metodu ile projenin Gant Çizelgesi elde edilmiştir. Toplamda 13 aktiviteden oluşan projenin toplam proje süresi 38 gün olarak elde edilmektedir.

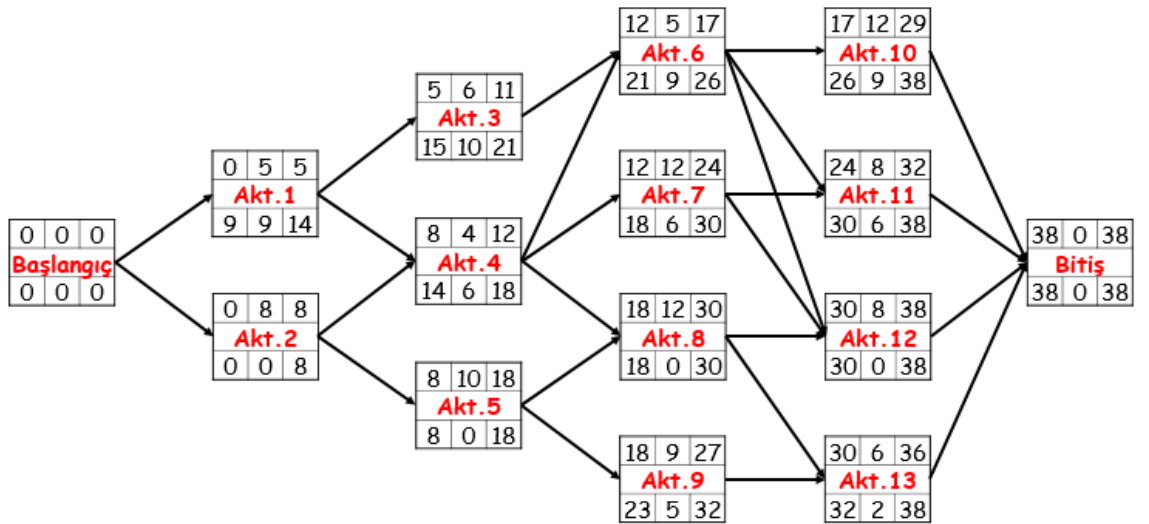
Gant Çizelgesi incelendiğinde bazı günlerde mevcut kaynak miktarından fazla kaynağa ihtiyaç duyulduğu tespit edilmiştir. Kaynak dengelenmesinin yapılması için bazı aktivitelerin ötelenmesi gerekmektedir.

Geleneksel yöntemlerle optimizasyon yapıldığında hem işlem çok uzun sürmektedir, hem de elde edilen sonucun doğruluğu kontrol edecek bir yöntem literatürde bulunmamaktadır. Bu yüzden bu tez çalışmasında kaynak dengelenme problemlerinin genetik algoritma kullanılarak çözülmesi ile çok daha kısa sürede en optimum sonuca ulaşılması hedeflenmektedir.

Tablo 5.1. Aktivite Süreleri, Öncülleri ve Günlük Kaynak Miktarları

Aktivite	Süre (Gün)	Öncüller	Günlük Kaynaklar
Aktivite 1	5	-	16
Aktivite 2	8	-	8
Aktivite 3	6	Aktivite 1	15
Aktivite 4	4	Aktivite 1, 2	14
Aktivite 5	10	Aktivite 2	9
Aktivite 6	5	Aktivite 3, 4	14
Aktivite 7	12	Aktivite 4	13
Aktivite 8	12	Aktivite 4, 5	8
Aktivite 9	9	Aktivite 5	3
Aktivite 10	12	Aktivite 6	12
Aktivite 11	8	Aktivite 6, 7	9
Aktivite 12	8	Aktivite 6, 7, 8	3
Aktivite 13	6	Aktivite 8, 9	10

Optimizasyon öncesi örnek proje kutu diyagramı yöntemiyle çözümlenmiştir (Şekil 5.1.). Örnek olarak kullanılan proje 13 aktiviteden oluşmaktadır. Aktivitelerin öncülleri ve aktivite süreleri dikkate alınarak CPM ile örnek projenin Gant Çizelgesi elde edilmektedir (Şekil 5.3.).



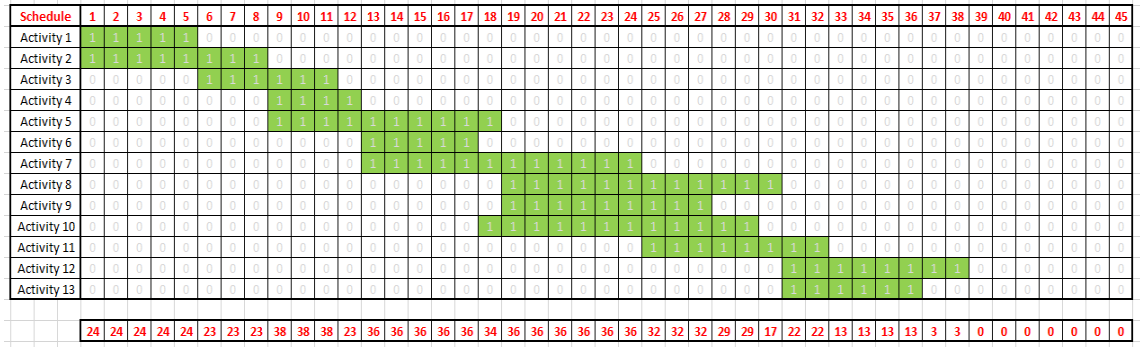
Şekil 5.1 : Dengeleme Öncesi CPM Çözümü

Geleneksel Gösterim



Şekil 5.2 : Geleneksel Kutu Diyagramı Gösteri

Mevcut durumda, hiçbir aktivite ötelenmediğinde, bazı günlerde kaynak miktarı 24'ün üzerine çıkmaktadır.



Şekil 5.3 : Öteleme Yapılmadan Önceki Gantt Çizelgesi

Kaynaklar kısıtlı olduğundan günlük kaynak miktarının 24'ün üzerine çıkması istenmeyen bir durumdur. Bu sebeple kaynak dengeleme yapılması gerekmektedir. Dengeli bir Gant Çizelgesi elde etmek için bu çalışmada geleneksel kutu diyagramı yöntemine (Şekil 5.2.) farklı bir boyut kazandırılarak öteleme süresinin yazıldığı ek bir kutucuk eklenmektedir (Şekil 5.4.).

Önerilen Gösterim

DP		
ES	D	EF
Aktivite		
LS	TF	LF

DP: Öteleme Süresi
ES: Erken Başlangıç
EF: Erken Bitiş
LS: Geç Başlangıç
LF: Geç Bitiş
TF: Toplam Bolluk
D: Aktivite Süresi

Şekil 5.4 : Önerilen Kutu Diyagramı Gösterimi

Toplamda 4 adet aktivite ötelenmiştir ve bunlar aktivite 4, aktivite 6, aktivite 11 ve aktivite 12 olmuştur. Sırasıyla 3 gün, 12 gün, 4 gün ve 4 gün ötelenmiştir. Bu ötelenme işlemleri sonucunda proje süresi 38'den 44'e çıkmıştır.

5.1. Genetik Algoritma İle Optimizasyonun Yapılması

Tez çalışmasında genetik algoritma kullanılarak proje süresi optimize edilmiştir. Her genetik algoritma problemde 2 farklı fonksiyon yazılmalıdır. Bu çalışmada "Myfitness" ve "Myconstraints" fonksiyonları yazılmıştır. Optimizasyonun sağlanması için karar değişkenleri cinsinden kısıtlamaları ve objektif fonksiyonlar yazılmıştır. Bunun sebebi tez çalışmasındaki asıl amacın proje süresini minimize etmek olmasıdır.

Kaynak dengeleme yapılması hedeflenen projenin genetik algoritma ile çözümü Matlab programı kullanılarak yapılmıştır. Matlab programı üzerinde öncelikle optimize edilmesi hedeflenen objektif fonksiyon olan "myfitness" fonksiyonu tanımlanmıştır. Sonrasında kısıtlamaları temsil eden "myconstraints" fonksiyonu tanımlanmıştır. Asıl algoritmayı yazmak için tanımlanan bu iki fonksiyon "main" fonksiyonunda kullanılmıştır. Ana fonksiyonda objektif fonksiyon ve kısıtlamaların yanı sıra genetik algoritmanın temelini oluşturan genetik parametlerin de girilmesi gerekmektedir. Bunlar; popülasyon sayısı, maksimum jenerasyon sayısı, çaprazlama oranı ve mutasyon oranıdır. Her bir genetik parametrenin sayısal değeri tez çalışmasının uygulama kısmında detaylandırılmıştır.

Objektif fonksiyon, amacı temsil etmektedir.

Karar deęişkenleri, hangi aktivitenin kaç gün öteleneceęidir.

Amaç: Proje süresini minimumda tutmak

Kısıtlamalar: Kaynaklar (Her gün kullanılan kaynak sayısı maksimum 24 olmalı)

$x(i)$: Aktivitenin kaç gün öteleneceęi

$d(i)$: Her bir aktivitenin süresi

$dr(i)$: i . Aktivitenin günlük kaynak ihtiyacı

$Schedule_1(i)$: i . gün içerisinde 1. aktivitenin gerçekleşip gerçekleşmemesi durumu

$tr(i)$: i . gündeki toplam kaynak sayısı

$ES(i)$: i . aktivitenin erken başlangıç süresi

$EF(i)$: i . aktivitenin erken bitiş süresi

Myfitness: Optimize edilen fonksiyon

Myconstraints: Kısıtlamalar

5.2. Myfitness Fonksiyonu

Function $y = myFitness(x)$

Aktivitelerin süresi girilir.

$d(1) = 5$;

...

$d(13) = 6$;

Kutu diyagramlarında kritik yol yönteminin mantığı kullanılarak proje süresi elde edilir.

Erken tamamlama süresi: $EF(i) = ES(i) + d(i)$

$ES(1) = 0 + x(1)$;

$EF(1) = ES(1) + d(1)$;

...

$ES(13) = \max([EF(8), EF(9)]) + x(13);$

$F(13) = ES(13) + d(13);$

$Project_Duration = \max([EF(10), EF(11), EF(12), EF(13)]);$

$y = Project_Duration;$

end

```
1 function y = myFitness(x)
2
3 - d(1) = 5;
4 - d(2) = 8;
5 - d(3) = 6;
6 - d(4) = 4;
7 - d(5) = 10;
8 - d(6) = 5;
9 - d(7) = 12;
10 - d(8) = 12;
11 - d(9) = 9;
12 - d(10) = 12;
13 - d(11) = 8;
14 - d(12) = 8;
15 - d(13) = 6;
```

Şekil 5.5 : Matlab’da Aktivite Sürelerinin Tanımlanması

```
17 - dr(1) = 16;
18 - dr(2) = 8;
19 - dr(3) = 15;
20 - dr(4) = 15;
21 - dr(5) = 9;
22 - dr(6) = 14;
23 - dr(7) = 13;
24 - dr(8) = 8;
25 - dr(9) = 3;
26 - dr(10) = 12;
27 - dr(11) = 9;
28 - dr(12) = 3;
29 - dr(13) = 10;
```

Şekil 5.6 : Matlab’da Aktivitelerin Günlük Kaynak İhtiyacının Tanımlanması

```

31 - ES(1) = 0 + x(1);
32 - EF(1) = ES(1) + d(1);
33
34 - ES(2) = 0 + x(2);
35 - EF(2) = ES(2) + d(2);
36
37 - ES(3) = EF(1) + x(3);
38 - EF(3) = ES(3) + d(3);
39
40 - ES(4) = max([EF(1), EF(2)]) + x(4);
41 - EF(4) = ES(4) + d(4);
42
43 - ES(5) = EF(2) + x(5);
44 - EF(5) = ES(5) + d(5);
45
46 - ES(6) = max([EF(3), EF(4)]) + x(6);
47 - EF(6) = ES(6) + d(6);
48
49 - ES(7) = EF(4) + x(7);
50 - EF(7) = ES(7) + d(7);
51
52 - ES(8) = max([EF(4), EF(5)]) + x(8);
53 - EF(8) = ES(8) + d(8);
54
55 - ES(9) = EF(5) + x(9);
56 - EF(9) = ES(9) + d(9);
57
58 - ES(10) = EF(6) + x(10);
59 - EF(10) = ES(10) + d(10);
60
61 - ES(11) = max([EF(6), EF(7)]) + x(11);
62 - EF(11) = ES(11) + d(11);
63
64 - ES(12) = max([EF(6), EF(7), EF(8)]) + x(12);
65 - EF(12) = ES(12) + d(12);
66
67 - ES(13) = max([EF(8), EF(9)]) + x(13);
68 - EF(13) = ES(13) + d(13);
69
70 - Project_Duration = max([EF(10), EF(11), EF(12), EF(13)]);
71

```

Şekil 5.7 : Matlab'da Proje Süresinin Elde Edilmesi

5.3. MyConstraints Fonksiyonu

MyConstraints fonksiyonu kısıtlamaları temsil etmektedir. Bu çalışmada kısıtlamalar, kaynaklardır. Günlük kaynak kullanımının 24'ü geçmemesi kısıtı mevcuttur. MyConstraints fonksiyonu yazılırken eğer kısıtlamalar bir değerin altında ve üstünde ise "c", eşitlik var ise "c_eq" kullanılmaktadır. Örnek çalışmada günlük kaynak miktarının 24'ten küçük olması gerektiğinden, burada "c" fonksiyonu kullanılmaktadır.

```

function [c, c_eq] = MyConstraints(x)

```

Şekil 5.8 : Matlab'da MyConstraints Fonksiyonu Tanımlanması

MyConstraints fonksiyonu yazılırken (Şekil 5.8.) yine Myfitness fonksiyonundaki gibi sırasıyla aktivite süresi (Şekil 5.5.), aktivitelerin günlük kaynak ihtiyacı (Şekil 5.6.), proje süresi (Şekil 5.7.) bilgileri ile kod yazılmaya başlanmıştır.

Projenin Gant Chart'ını elde etmek için for döngüsü kullanılmıştır (Şekil 5.9.). Optimizasyon sonrası elde edilecek proje süresinin kaç gün olacağı bilinmediğinden, güvenli tarafta kalmak adına 1'den 100'e kadar işlem tekrarlanmıştır.

“Schedule..1(i) = 1” ile 1.aktivitenin o gün var olup olmadığı bulunmuştur. Eğer aktivite o gün var ise alacağı değer 1, eğer aktivite o gün yok ise alacağı değer 0 olmuştur (Şekil 5.9.). Aynı formül 1. aktiviteden 13. aktiviteye kadar uygulanmıştır. Böylece projenin Gant Chart’ı elde edilmiştir.

```
for i = 1:100
    if (i>=ES(1)+1) && (i<=EF(1))
        schedule_1(i) = 1;
    else
        schedule_1(i) = 0;
    end
end
```

Şekil 5.9 : Matlab’da For Döngüsü ile Gantt Çizelgesi Elde Edilmesi

Gantt çizelgesi elde edildikten sonra toplam kaynak miktarının hesaplanması için yine for döngüsü kullanılmıştır (Şekil 5.10.). Toplam kaynak miktarını elde etmek için, aktivitenin o gün gerçekleşip gerçekleşmeyeceğini gösteren değer olan “Schedule” ile günlük kaynak miktarı çarpılmıştır. Eğer aktivite o gün gerçekleşmiyorsa, günlük kaynak miktarı “sıfır(0)” değeri ile çarpıldığından toplam kaynağa etkisi olmamaktadır. Eğer aktivite o gün gerçekleşiyorsa, günlük kaynak miktarı “bir(1)” değeri ile çarpıldığından toplam kaynağa etkisi olmaktadır. Tüm aktiviteler için aynı işlem yapıldıktan sonra hepsi toplanarak toplam kaynak miktarı elde edilmiştir.

```
for i = 1:100
    tr(i) = schedule_1(i)*dr(1) + schedule_2(i)*dr(2) + schedule_3(i)*dr(3) + schedule_4(i)*dr(4) +
    schedule_5(i)*dr(5) + schedule_6(i)*dr(6) + schedule_7(i)*dr(7) + schedule_8(i)*dr(8) + schedule_9(i)*dr(9) +
    schedule_10(i)*dr(10) + schedule_11(i)*dr(11) + schedule_12(i)*dr(12) + + schedule_13(i)*dr(13);
end
```

Şekil 5.10 : Matlab’da Toplam Kaynak Miktarının Elde Edilmesi

MyConstarints fonksiyonunun elde edilmesini sağlayan formül Şekil 5.8.’deki gibi yazılmıştır. MyConstraints fonksiyonu yazılırken eğer kısıtlamalar bir değer altında ve üstünde ise “c”, eşitlik var ise “c_eq” kullanılmaktadır. Örnek çalışmada günlük kaynak miktarının 24’ten küçük olması gerektiğinden, burada “c” fonksiyonu kullanılmaktadır.

Optimizasyon sonrası elde edilecek proje süresinin kaç gün olacağı bilinmediğinden, güvenli tarafta kalmak adına 1'den 100'e kadar işlem tekrarlanmıştır (Şekil 5.11.).

```
c = [tr(1)-24; tr(2)-24; tr(3)-24; tr(4)-24; tr(5)-24; tr(6)-24; tr(7)-24; tr(8)-24; tr(9)-24; tr(10)-24;  
tr(11)-24; tr(12)-24; tr(13)-24; tr(14)-24; tr(15)-24; tr(16)-24; tr(17)-24; tr(18)-24; tr(19)-24; tr(20)-24;  
tr(21)-24; tr(22)-24; tr(23)-24; tr(24)-24; tr(25)-24; tr(26)-24; tr(27)-24; tr(28)-24; tr(29)-24; tr(30)-24;  
tr(31)-24; tr(32)-24; tr(33)-24; tr(34)-24; tr(35)-24; tr(36)-24; tr(37)-24; tr(38)-24; tr(39)-24; tr(40)-24;  
tr(41)-24; tr(42)-24; tr(43)-24; tr(44)-24; tr(45)-24; tr(46)-24; tr(47)-24; tr(48)-24; tr(49)-24; tr(50)-24;  
tr(51)-24; tr(52)-24; tr(53)-24; tr(54)-24; tr(55)-24; tr(56)-24; tr(57)-24; tr(58)-24; tr(59)-24; tr(60)-24;  
tr(61)-24; tr(62)-24; tr(63)-24; tr(64)-24; tr(65)-24; tr(66)-24; tr(67)-24; tr(68)-24; tr(69)-24; tr(70)-24;  
tr(71)-24; tr(72)-24; tr(73)-24; tr(74)-24; tr(75)-24; tr(76)-24; tr(77)-24; tr(78)-24; tr(79)-24; tr(80)-24;  
tr(81)-24; tr(82)-24; tr(83)-24; tr(84)-24; tr(85)-24; tr(86)-24; tr(87)-24; tr(88)-24; tr(89)-24; tr(90)-24;  
tr(91)-24; tr(92)-24; tr(93)-24; tr(94)-24; tr(95)-24; tr(96)-24; tr(97)-24; tr(98)-24; tr(99)-24; tr(100)-24];  
  
c_eq = [];
```

Şekil 5.11 : Matlab'da MyConstraints Fonksiyonunun Elde Edilmesi

5.4. Main Fonksiyonu

Main fonksiyonu yazılmadan önce myFitness ve myConstraints fonksiyonları tanımlanır. Main fonksiyonunda tanımlanan bu fonksiyonlar kullanılır. Bunların haricinde en önemli nokta popülasyon sayısı, maksimum jenerasyon gibi verilerin belirlenmesidir. Bu sayısal veriler değişken sayısına göre deneme - yanılma yöntemi ve tecrübeye dayalı olarak seçilir. Seçilen değerler Şekil 5.12.'de gösterilmektedir.

```
Lb = [0 0 0 0 0 0 0 0 0 0 0 0 0];  
Ub = [30 30 30 30 30 30 30 30 30 30 30 30 30];  
  
ConsFcn = @MyConstraints;  
FitFcn = @myFitness;  
nvars = 13;  
  
opts = optimoptions(@ga, ...  
    'PopulationSize', 2000, ...  
    'MaxGenerations', 120, ...  
    'EliteCount', 240, ...  
    'FunctionTolerance', 1e-8, ...  
    'PlotFcn', @gaplotbestf);  
  
[x, fval] = ga(FitFcn, nvars, [], [], [], [], Lb, Ub, ConsFcn, [1 2 3 4 5 6 7 8 9 10 11 12 13], opts);
```

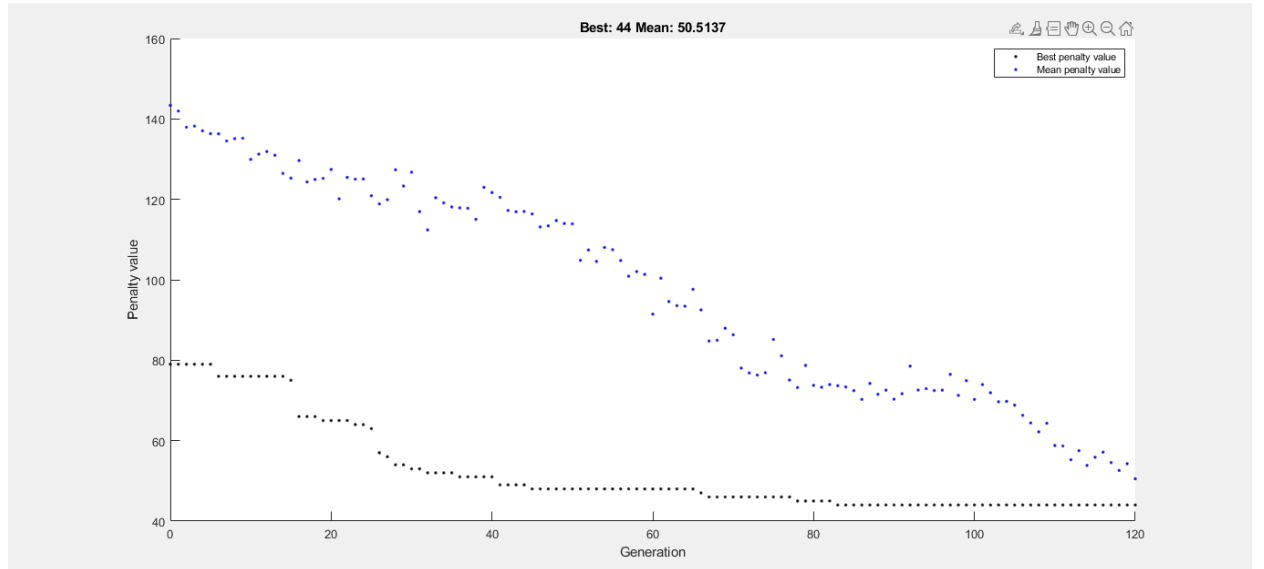
Şekil 5.12 : Matlab'da Main Fonksiyonunun Tanımlanması

BÖLÜM 6. BULGULAR VE TARTIŞMALAR

Bu çalışmada inşaat projelerindeki kaynak düzensizliğinin yol açtığı süre gecikmeleri maddi hasarların önlenmesi için kaynak dengeleme problemlerinin sezgisel optimizasyon yöntemlerinde GA ile çözümlenmesi Matlab programında yapılmaktadır.

Genetik algoritma ile bilgisayar programında optimizasyon yapılması geleneksel yöntemlerden çok daha hızlı, güvenilir ve en optimum sonucu verdiği gözlemlenmektedir. Kaynak dengeleme öncesinde proje süresi 38 gün iken optimizasyon sonrasında proje süresi 44 güne çıkmaktadır (Şekil 6.1.).

Kaynaklar kısıtlı olduğundan günlük kaynak miktarının 24'ün üzerine çıkması istenmeyen bir durumdur. Bu sebeple kaynak dengeleme yapılması gerekmektedir. Dengeli bir Gant Çizelgesi elde etmek için GA ile Matlab programında kaynak dengeleme optimizasyonu yapıldığında aktivitelerin öteleme süreleri elde edilmiştir (Şekil 6.2.). Elde edilen öteleme sürelerine göre yeni CPM grafiği Şekil 6.3.'deki gibidir.



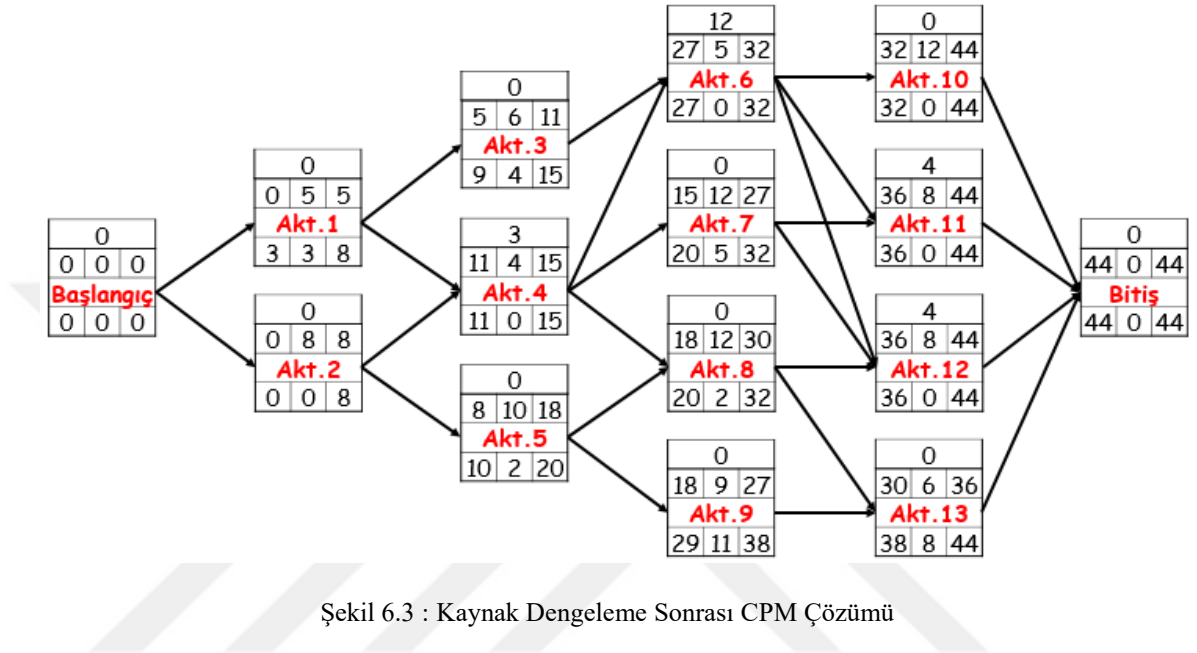
Şekil 6.1 : Matlab'da Optimizasyon Sonucu

```

Command Window
x =
0 0 0 3 0 12 0 0 0 0 4 4 0
fx

```

Şekil 6.2 : Optimizasyon sonrası elde edilen x değerleri



Toplamda 4 adet aktivite ötelenmiştir ve bunlar aktivite 4, aktivite 6, aktivite 11 ve aktivite 12 olmuştur. Sırasıyla 3 gün, 12 gün, 4 gün ve 4 gün ötelenmiştir. Bu ötelenme işlemleri sonucunda proje süresi 38’den 44’e çıkmıştır (Şekil 6.3.).

Optimizasyon öncesi Gantt çizelgesi incelenen projenin günlük kaynak ihtiyaçlarında 24’ü geçen aktiviteler olduğu görülmektedir. Optimizasyon sonrasında günlük kaynak miktarının 24’ü geçmemesi sağlanmakta ve dengeli bir Gantt çizelgesi elde edilmektedir (Şekil 6.4.).

Schedule	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45									
Activity 1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							
Activity 2	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
Activity 3	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
Activity 4	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
Activity 5	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
Activity 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
Activity 7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
Activity 8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
Activity 9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
Activity 10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
Activity 11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Activity 12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Activity 13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Resource	24	24	24	24	24	23	23	23	24	24	24	24	24	24	24	22	22	22	22	24	24	24	24	24	24	24	24	24	22	22	22	24	24	22	22	22	22	24	24	24	24	24	24	24	24	24	24	24	0	0	0			

Şekil 6.4 : Optimizasyon sonrası Gantt Çizelgesi

Genetik algoritmanın bu sorunu çözebilmesi için sorudaki deęişken sayısına ve problemin zorluk derecesine göre tecrübeye baęlı popülasyon sayısı, maksimum jenerasyon sayısı ayarlanmaktadır. Bu deęerler yetersiz sayıda girildiğinde GA doğru cevabı bulamamaktadır. Girilen bu deęerler deneme-yanılma ve tecrübeye baęlı olarak tespit edilmektedir.



BÖLÜM 7. SONUÇ VE ÖNERİLER

7.1. Sonuç

İnşaat projelerinde etkin bir planlama yapılması büyük bir önem arz etmektedir. İnşaat projelerinin karmaşık yapısı gereği planlanan süreden çok daha uzun sürede tamamlanması kaçınılmaz olmaktadır. Projenin planlanan süreden geç tamamlanması proje performansını ciddi anlamda olumsuz yönde etkileyip maddi zararlara yol açmaktadır. Bu maddi zararların önüne geçmek için proje yönetiminde kaynak dengeleme çözümlenmeleri sıkça kullanılmaktadır.

Kaynak dengeleme proje planlamayı daha kaliteli hale getirmektedir. Kaynak dengeleme problemlerinin geleneksel yöntemlerle çözümü somut bir sonuç vermediğinden ve çok uzun süreçlerde çözüm sağlanabildiğinden optimizasyona ihtiyaç duyulmaktadır. Bu sorunu ortadan kaldırmak amacıyla bu çalışmada kaynak dengeleme problemlerini sezgisel optimizasyon yöntemlerinden genetik algoritmayla çözülmesi hedeflenmiştir. Bu tez çalışmasında genetik algoritmayla çözülmesi amaçlanan projelerde kısıtlı kaynaklar kullanılarak optimizasyon gerçekleştirilmiştir. Temel amaç, proje süresini minimum seviyede tutarak kısıtlı kaynaklar altında kaynak dengelemesinin yapılmasıdır.

Tez çalışmasındaki örnek proje geleneksel proje yöntemlerinde en sık kullanılan CPM ile çözümlenip proje süresi elde edilmiştir. Geleneksel yöntemlerden farklı ve yeni olarak bu çalışmada CPM’de olmayan bir teknik eklenmiş ve bazı aktivitelerin ötelenerek projenin optimizasyonu sağlanmıştır. Öteleme işleminden sonra yeni proje süresi önce geleneksel yöntemlerle elde edilmiştir. Daha sonra ise aynı proje genetik algoritma yöntemi kullanılarak kısıtlı kaynaklar altında Matlab programında çözümlenmiştir. Elde edilen sonucun doğruluğu karşılaştırılıp onaylanmıştır. Genetik algoritma ile bilgisayar programında optimizasyon yapılması geleneksel yöntemlerden çok daha hızlı, güvenilir

ve en optimum sonucu verdiđi gözlemlenmiştir. Kaynak dengeleme öncesinde proje süresi 38 gün iken optimizasyon sonrasında proje süresi 44 güne çıkmıştır. Optimizasyon öncesi Gantt çizelgesi incelenen projenin günlük kaynak ihtiyaçlarında 24'ü geçen aktiviteler olduđu görülmektedir. Optimizasyon sonrasında günlük kaynak miktarının 24'ü geçmemesi sağlanmış ve dengeli bir Gantt çizelgesi elde edilmektedir.

Geleneksel yöntemlerle Excel üzerinde kaynak dengeleme çözümlenmesi deneme-yanılma ile yapıldığında hangi aktivitenin kaç gün öteleneceđi karar verilmesi günler süren uzun bir süreç almaktadır. Matlab üzerinde genetik algoritması yazılarak yapılan çözümlenme ise yaklaşık bir dakikada tamamlanmaktadır.

7.2. Çalışmanın Literatüre Faydası

7.2.1. Teorik katkısı

Bu çalışmada kaynak dengeleme problemlerinin sezgisel optimizasyon yöntemlerinden genetik algoritmalar ile nasıl çözümlenebileceđi detaylı olarak anlatılmaktadır. Dolayısıyla bu tez çalışmasının çeşitli optimizasyon problemlerini genetik algoritma ile çözümlenmesine dair ilgisini arttıracakđı düşünölmektedir.

Literatürde inşaat projelerinde kaynak dengeleme sorunlarının çözümlenmesi ile ilgili yapılan çalışmalar yok denecek kadar azdır. Bu çalışma bu alanda literatüre bir öncülük getirip sonrasında bu alanda yapılacak çalışmalara büyük bir katkı sağlamayı hedeflemektedir. Farklı programlar ve projeler için uygulanabilir bir kod ortaya koyan bu çalışma sadece inşaat mühendisliđi problemlerinde deđil, tüm proje yönetimi problemlerindeki optimizasyon uygulamalarına uygulanabilmeyi hedeflemektedir.

7.2.2. Pratik katkısı

İnşaat projelerinin karmaşıklığı sebebiyle planlama ve proje yönetimi büyük bir önem taşımaktadır. Planlama programlarının yüksek maliyeti nedeniyle büyük ölçekli olmayan inşaat projelerinde proje yönetimi ihmal edilmektedir. Bu çalışmadaki kodlama ile düşük maliyetlerle kaynak dengeleme sorunlarına çözüm getirilmesi amaçlanmaktadır.

Bu çalışmada optimizasyon problemlerini GA ile çözenin kodlama mantığı da açıklandığından proje yöneticileri karşılaştıkları farklı problemleri çözmek için de bu yöntemden faydalanabilecektir.

7.2.3. Çalışmanın kısıtları

Bu çalışmada spesifik tek bir örnek üzerinden yola çıkılarak optimizasyon yapılmaktadır. Dolayısıyla tüm genel geçer problemleri çözebilecek bir çözümlenme olduğu söylenememektedir. Fakat yapılan çözümlenmenin açıklayıcı olması birçok farklı geniş kapsamlı örneklere de uygulanabilmesinin önünü açmaktadır.

Bu tez çalışmasında çözümlenmesi yapılan proje tek bir kaynağın kısıtlamaları altında çözümlenmektedir. Dolayısıyla çoklu kaynak kısıtlamaları altında yapılacak kaynak dengeleme problemlerine çözüm sağlayacak bir çalışma olduğu söylenememektedir. Fakat ihtiyaç duyulduğu takdirde, çalışmada izlenen yol ve yöntemler takip edilecek farklı kaynaklara veya çoklu kaynak problemlerine uygulanarak optimizasyon sağlanması mümkündür.

7.3. Öneriler

Bu tez çalışmasında kaynak dengeleme problemleri sezgisel optimizasyon yöntemlerin genetik algoritma ile çözümlenmektedir. Literatürde diğer algoritmaların kaynak dengeleme problemlerine uygulanması da kısıtlıdır. Farklı algoritmaların uygulanarak bu problemlerin çözümlenmesi sağlanabilir. Farklı algoritmaların uygulandığı çözümlenmeler sonucu elde edilen veriler karşılaştırılarak en etkin algoritma metodu elde edilebilir.

Çalışmada GA optimizasyonu Matlab programı kullanılarak elde edilmektedir. İleriki çalışmalarda kaynak dengeleme problemlerinin çözümlenmesi Phyton gibi farklı programlara da uyarlanarak hangi programın daha verimli çözümlenmeler sunduğu karşılaştırılabilir.

KAYNAKLAR

- [1] Mahamid, I., Bruland, A., & Dmairi, N. (2012). Causes of Delay in Road Construction Projects. *Journal of Management in Engineering*, 28(3), 300-310. [https://doi.org/10.1061/\(ASCE\)ME.1943-5479.0000096](https://doi.org/10.1061/(ASCE)ME.1943-5479.0000096)
- [2] Tariq, J., & Shujaa Safdar Gardezi, S. (2023). Study the delays and conflicts for construction projects and their mutual relationship: A review. *Ain Shams Engineering Journal*, 14(1), 101815. <https://doi.org/10.1016/j.asej.2022.101815>
- [3] Bertelsen, S., & Sacks, R. (2007). Towards a new understanding of the construction industry and the nature of its production. *15th Conference of the International Group for Lean Construction*, 46-56.
- [4] Larsson, J., & Larsson, L. (2020). Integration, Application and Importance of Collaboration in Sustainable Project Management. *Sustainability*, 12(2), Art. 2. <https://doi.org/10.3390/su12020585>
- [5] Algorithms for the Multi-Resource Generalized Assignment Problem | *Management Science*. (t.y.). Geliş tarihi 09 Aralık 2022, gönderen <https://pubsonline.informs.org/doi/abs/10.1287/mnsc.37.6.695>
- [6] Least impact algorithm for resource allocation. (t.y.). Geliş tarihi 09 Aralık 2022, gönderen <https://cdnsiencepub.com/doi/abs/10.1139/193-023>
- [7] Markou, Ch., Koulinas, G. K., & Vavatsikos, A. P. (2017). Project resources scheduling and leveling using Multi-Attribute Decision Models: Models implementation and case study. *Expert Systems with Applications*, 77, 160-169. <https://doi.org/10.1016/j.eswa.2017.01.035>
- [8] Unegbu, H. C. O., Yawas, D. S., & Dan-asabe, B. (2022). An investigation of the relationship between project performance measures and project management practices of construction projects for the construction industry in Nigeria. *Journal of King Saud University - Engineering Sciences*, 34(4), 240-249. <https://doi.org/10.1016/j.jksues.2020.10.001>
- [9] Yap, J. B. H., Lim, B. L., & Skitmore, M. (2022). Capitalising knowledge management (KM) for improving project delivery in construction. *Ain Shams Engineering Journal*, 13(6), 101790. <https://doi.org/10.1016/j.asej.2022.101790>

- [10] Liao, T. W., Egbelu, P. J., Sarker, B. R., & Leu, S. S. (2011). Metaheuristics for project and construction management – A state-of-the-art review. *Automation in Construction*, 20(5), 491-505. <https://doi.org/10.1016/j.autcon.2010.12.006>
- [11] Guo, K., & Zhang, L. (2022). Multi-objective optimization for improved project management: Current status and future directions. *Automation in Construction*, 139, 104256. <https://doi.org/10.1016/j.autcon.2022.104256>
- [12] Url1<https://www.academia.edu/13809641/Optimisation_of_Construction_Resource_Allocation_and_Levelling_Using_Genetic_Algorithm>, date retrieved 15.10.2022.
- [13] Yang, J.-B., & Kao, C.-K. (2012). Critical path effect based delay analysis method for construction projects. *International Journal of Project Management*, 30(3), 385-397. <https://doi.org/10.1016/j.ijproman.2011.06.003>
- [14] Zareei, S. (2018). Project scheduling for constructing biogas plant using critical path method. *Renewable and Sustainable Energy Reviews*, 81, 756-759. <https://doi.org/10.1016/j.rser.2017.08.025>
- [15] Leu, S.-S., Yang, C.-H., & Huang, J.-C. (2000). Resource leveling in construction by genetic algorithm-based optimization and its decision support system application. *Automation in Construction*, 10(1), 27-41. [https://doi.org/10.1016/S0926-5805\(99\)00011-4](https://doi.org/10.1016/S0926-5805(99)00011-4)
- [16] Hua, Z., Liu, Z., Yang, L., & Yang, L. (2022). Improved genetic algorithm based on time windows decomposition for solving resource-constrained project scheduling problem. *Automation in Construction*, 142, 104503. <https://doi.org/10.1016/j.autcon.2022.104503>
- [17] Ponz-Tienda, J. L., Yepes, V., Pellicer, E., & Moreno-Flores, J. (2013). The Resource Leveling Problem with multiple resources using an adaptive genetic algorithm. *Automation in Construction*, 29, 161-172. <https://doi.org/10.1016/j.autcon.2012.10.003>
- [18] Kastor, A., & Sirakoulis, K. (2009). The effectiveness of resource levelling tools for Resource Constraint Project Scheduling Problem. *International Journal of Project Management*, 27(5), 493-500. <https://doi.org/10.1016/j.ijproman.2008.08.006>
- [19] Liu, Y., Zhao, S.-L., Du, X.-K., & Li, S.-Q. (2005). Optimization of resource allocation in construction using genetic algorithms. *2005 International Conference on Machine Learning and Cybernetics*, 6, 3428-3432 Vol. 6. <https://doi.org/10.1109/ICMLC.2005.1527534>
- [20] He, W., Li, W., & Wang, W. (2021). Developing a Resource Allocation Approach for Resource-Constrained Construction Operation under Multi-Objective Operation. *Sustainability*, 13(13), Art. 13. <https://doi.org/10.3390/su13137318>

- [21] Khattab, M. M., & Søyland, K. (1996). Limited-resource allocation in construction projects. *Computers & Industrial Engineering*, 31(1), 229-232. [https://doi.org/10.1016/0360-8352\(96\)00118-0](https://doi.org/10.1016/0360-8352(96)00118-0)
- [22] Precedence and Arrow Networking Techniques for Construction | Wiley. (t.y.). Wiley.Com. Geliş tarihi 09 Aralık 2022, gönderen <https://www.wiley.com/en-us/Precedence+and+Arrow+Networking+Techniques+for+Construction-p-9780471041238>
- [23] Easa, S. M. (1989). Resource leveling in construction by optimization. *Journal of Construction Engineering and Management*, 115(2), 302-316. Scopus. [https://doi.org/10.1061/\(ASCE\)0733-9364\(1989\)115:2\(302\)](https://doi.org/10.1061/(ASCE)0733-9364(1989)115:2(302))
- [24] Joseph J. Moder, Cecil R. Phillips, Edward W. Davis-Project Management With CPM, Pert and Precedence Diagramming-Van Nostrand Reinhold (1983) | PDF | *Scheduling (Production Processes) | Mathematical Optimization*. (t.y.). Scribd. <https://www.scribd.com/document/366201111/Joseph-J-Moder-Cecil-R-Phillips-Edward-W-Davis-Project-Management-With-Cpm-Pert-and-Precedence-Diagramming-Van-Nostrand-Reinhold-1983>
- [25] Qiao, J., & Li, Y. (2018). Resource leveling using normalized entropy and relative entropy. *Automation in Construction*, 87, 263-272. <https://doi.org/10.1016/j.autcon.2017.12.022>
- [26] Bandelloni, M., Tucci, M., & Rinaldi, R. (1994). *Optimal resource leveling using non-serial dyanamic programming*. *European Journal of Operational Research*, 78(2), 162-177. [https://doi.org/10.1016/0377-2217\(94\)90380-8](https://doi.org/10.1016/0377-2217(94)90380-8)
- [27] Christodoulou, S. E., Ellinas, G., & Michaelidou-Kamenou, A. (2010). Minimum moment method for resource leveling using entropy maximization. *Journal of Construction Engineering and Management*, 136(5), 518-527. Scopus. [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0000149](https://doi.org/10.1061/(ASCE)CO.1943-7862.0000149)
- [28] Chen, L., Guo, S., Lu, J., & Gerschewski, S. (2021). Outward FDI and efficiency in within-firm resource allocation – Evidence from firm-level data of China. *Journal of Asian Economics*, 74, 101298. <https://doi.org/10.1016/j.asieco.2021.101298>
- [29] Xiao, L. (2021). Parameter Tuning of PID Controller for Beer Filling Machine Liquid Level Control Based on Improved Genetic Algorithm. *Computational Intelligence and Neuro*
- [30] Mallawaarachchi, V. (2020, Mart 1). Introduction to Genetic Algorithms—*Including Example Code*. Medium. <https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3>
- [31] Url-2<<https://docplayer.biz.tr/50035095-Musteri-odakli-bakim-onarim-faaliyetleri-performansinin-dinamik-tamirci-rotalama-problemi-ile-modellemesi-ve-optimizasyonu.html>>, date retrieved 19.11.2022.