



**T.C.
GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ**

**YÜKSEK
LİSANS
TEZİ**

**ROTA BELİRLEME İÇİN
HİBRİT ZEKİ MODEL GELİŞTİRİLMESİ**

MEHMET ÖNDER ÖNDEROĞLU

BİLİŞİM SİSTEMLERİ ANA BİLİM DALI

MART 2022



ROTA BELİRLEME İÇİN HİBRİT ZEKİ MODEL GELİŞTİRİLMESİ

Mehmet Önder ÖNDEROĞLU

**YÜKSEK LİSANS TEZİ
BİLİŞİM SİSTEMLERİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ**

MART 2022

Mehmet Önder Önderoğlu tarafından hazırlanan “ROTA BELİRLEME İÇİN HİBRİT ZEKİ MODEL GELİŞTİRİLMESİ” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Bilişim Sistemleri Ana Bilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Prof. Dr. İsmail ŞAHİN
Endüstriyel Tasarım Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Başkan: Prof. Dr. Mahir DURSUN
Elektrik - Elektronik Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Üye: Dr. Öğr. Üyesi Semih ÖZDEN
Elektronik ve Haberleşme Mühendisliği, Milli Savunma Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Tez Savunma Tarihi: 11/03/2022

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....

Prof. Dr. Aslıhan TÜFEKÇİ
Bilişim Enstitüsü Müdürü

ETİK BEYAN

Gazi Üniversitesi Bilişim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Mehmet Önder ÖNDEROĞLU

...../...../.....

ROTA BELİRLEME İÇİN HİBRİT ZEKİ MODEL GELİŞTİRİLMESİ

(Yüksek Lisans Tezi)

Mehmet Önder ÖNDEROĞLU

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

Şubat 2022

ÖZET

Aynı gün içerisinde müşterilerine hizmet götürmeye çalışan firmaların en büyük problemlerinden biri günlük rota planlamasının yapılmasıdır. Rota planlamalarının sadece en optimal şekilde yapılması şirketlerin ihtiyaçlarını tam olarak karşılamamaktadır. Şirketler müşterilerine en kısa mesafe rota planı oluştururken kendi belirledikleri önceliklendirmelere göre hizmetin devamlılığını sürekli hale getirmeye çalışmaktadır. Bu önceliklendirme ve en optimum çözümler için insan gücüne ihtiyaç duyulmaktadır. Bu tez çalışmasında şirketlerin bazı önceliklerini ön plana alınarak, en az maliyetle, müşteri memnuniyetinin artırılması amaçlayan zeki bir hibrit sistem tasarlanmıştır. Yapay zekâ destekli olarak tasarlanan Uzman Sistemin gerçekte hayatta işin uzmanı insanlar ile karşılaştırılması sağlanarak sistemin verimliliği ölçümlenmiştir. Oluşturulan hibrit sistem sayesinde guguk kuşu algoritması kullanılarak en kısa mesafe ölçümü yapılmış, Uzman sistem sayesinde şirketlerin önceliklendirme kriterleri iş sıralamalarına etki etmesi sağlanmıştır. Bu sayede iş gücü giderlerinin azalabileceği, optimizasyon konusunda başarılı ve müşteri memnuniyeti sağlayabilecek sonuçlar elde edilmiştir.

Bilim Kodu : 90542

Anahtar Kelimeler : Rota belirleme, uzman sistem, müşteri memnuniyeti, iş verimliliği artırma, iş planlama, yapay zekâ, guguk kuşu algoritması.

Sayfa Adedi : 69

Danışman : Prof. Dr. İsmail ŞAHİN

DEVELOPMENT OF A HYBRID INTELLIGENCE MODEL FOR ROUTING
(M. Sc. Thesis)

Mehmet Önder ÖNDEROĞLU

GAZİ UNIVERSITY
INFORMATICS INSTITUTE
Şubat 2022

ABSTRACT

One of the biggest problems of companies that trying to serve their customers on the same day is the day-to-day route planning. Only optimal route planning does not fully meet the needs of the companies. When creating the shortest distance route plan for customers, companies are trying to keep the service on going based on their own prioritization. Human power is needed for this prioritization and optimal solutions. In this thesis study, a smart hybrid system is designed to put some of the priorities of companies at the forefront and to increase customer satisfaction at minimal cost. The efficiency of the system is measured by comparing the expert system with people who are experts in their business in real life, which is designed with artificial intelligence withinwithi the experts in real life. With the hybrid system created, the shortest distance is measured using the cuckoo algorithm, and thanks to the Expert system, the criteria for prioritization are influenced by the job rankings. In this way, this results in reduced workforce costs, which can be successful in optimization and provide customer satisfaction.

Science Code : 90542

Key Words : Route identification, expert system, cuckoo search algorithm, workplanning, customer satisfaction, productivity of work augmentation, artificial intelligence

Page Number : 69

Supervisor : Prof. Dr. İsmail ŞAHİN

TEŞEKKÜR

Bu tez çalışmamın gerçekleşmesinde bana ilham kaynağı olan, bilgi ve tecrübelerini her ihtiyacımda paylaşan değerli danışman hocam Prof. Dr. İsmail ŞAHİN'e, beni hiçbir şart koşulda yalnız bırakmayıp destek olan kıymetli eşim Dilan ÖNDEROĞLU'na, neşesiyle bir an olsun peşimi bırakmayan kendisi için tüm zorluklara katlanabileceğim biricik kızım Defne'ye, en zorlandığım anlarda bana vazgeçmemeyi öğreten ve bu çalışmalar için en az benim kadar heyecanlanan değerli anneme, iş yerinde tez çalışmalarım boyunca stresimin azalmasını sağlayan değerli dostum Murathan POYRAZ'a teşekkürlerimi sunarım.



İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	x
ŞEKİLLERİN LİSTESİ.....	xi
RESİMLERİN LİSTESİ	xii
KISALTMALAR.....	xiii
1. GİRİŞ.....	1
2. LİTERATÜR TARAMASI	3
3. ROTA PROBLEMİ	7
3.1. Araç Rotalama Problemi	7
3.2. Araç Rotalama Problemi Temel Bileşenleri.....	7
3.3. Araç Rotalama Problemi Matematiksel Modeli.....	9
3.4. Rotalama Türleri.....	10
3.4.1. Rota süresi sınırsız rotalama problemleri	10
3.4.2. Rota süresi sınırlı rotalama problemleri	11
3.4.3. Zaman pencereci rotalama problemleri	11
3.4.4. Eşzamanlı rotalama ve çizelgeleme problemleri.....	12
3.4.5. Kapasiteli araç rotalama	13
4. ARAÇ ROTALAMA YÖNTEMLERİ.....	15
4.1. En Kısa Yol Yöntemi	16
4.2. Süpürme (Sweep) Yöntemi	16

	Sayfa
4.3. Gezgin Satıcı Problemi.....	18
4.4. Tasarruf Algoritması	19
4.5. Doğrusal Programlama Yardımıyla Araç Rotalama	20
4.6. Tavlama Benzetimi Yöntemi	22
4.7. Karınca Kolonisi Yöntemi	23
4.8. Genetik Algoritma Yöntemi.....	23
4.9. Guguk Kuşu Arama Yöntemi.....	24
5. GUGUK KUŞU ALGORİTMASI VE YAPAY ZEKA	25
5.1. Yapay Zekâ Tanımı ve Tarihçesi	25
5.2. Yapay Zekânın İlgilendiği ve Kullanıldığı Alanlar.....	28
5.3. Guguk Kuşu Arama Algoritması	31
5.3.1. Guguk kuşu üreme davranışlarının incelenmesi	31
5.3.2. Lévy uçuşu.....	31
5.3.3. Guguk kuşu algoritması	33
6. UZMAN SİSTEM.....	35
6.1. Uzman Sistem	35
6.2. Uzman Sistemlerin Avantajları	36
6.3. Uzman Sistemlerin Dezavantajları	38
6.4. Uzman Sistemlerin Tarihçesi.....	39
6.5. Uzman Sistem Uygulama Alanları	40
6.5.1. Kimyada uzman sistemler	40
6.5.2. Tarımda uzman sistemler.....	41
6.5.3. Eğitimde uzman sistemler	41
6.5.4. Endüstride uzman sistemler.....	42

	Sayfa
6.5.5. Tıpta uzman sistemler.....	42
7. ÖNERİLEN ALGOTİRMA VE YÖNTEM	45
7.1. Rota Takip Kullanıcı Arabirimi	51
7.2. Rota Takip Bilgi Tabanı.....	52
7.3. Rota Takip Çıkarım Mekanizması ve Açıklama Ünitesi	57
8. SONUÇLAR.....	59
KAYNAKLAR	63
ÖZGEÇMİŞ	69

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 5.1. Yapay zekâ uzman sistem karşılaştırması	26



ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 4.1. Araç rotalama yöntemleri	15
Şekil 4.2. Tasarruf yöntemindeki müşteri birleştirilmesi.....	20
Şekil 5.1. Lévy uçuşu.....	32
Şekil 6.1. Uzman sistem genel yapısı	36
Şekil 7.1. Guguk kuşu arama algoritması	48
Şekil 7.2. Rota takip uzman sistem çalışma yapısı	50



RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 7.1. Guguk kuşu algoritması programı	49
Resim 7.2. Guguk kuşu algoritması sonuç.....	49
Resim 7.3. Rota takip uzman sistem ana menü	51
Resim 7.4. Rota takip uzman sistem bilgi giriş ekranı	51
Resim 7.5. Rota takip uzman sistem bilgi tabanı görüntüsü.....	52
Resim 7.6. Rota takip uzman sistem kural tabanı	54
Resim 7.7. Rota takip uzman sistem kural listesi	55
Resim 7.8. Nitelik kural tabanı	55
Resim 7.9. Hassas kural tabanı	56
Resim 7.10. Kural tabanı karşılaştırma.....	56
Resim 8.1. Rota takip uzman sistem iş sırası listesi	60

KISALTMALAR

Çalışmada kullanılan kısaltmalar detayları ile beraber aşağıda sunulmuştur.

Kısaltmalar	Açıklamalar
ARP	Araç rotalama problemi
DKA	Değişken komşuluk arama
EDTARP	Eş zamanlı dağıt topla araç rotalama problemi
KARP	Kapasiteli araç rotalama problemi
KKS	Karınca koloni sistemi
GKA	Guguk kuşu algoritması
GSP	Gezgin satıcı problemi
US	Uzman sistem
ZKEDTARP	Zaman kısıtlı eş zamanlı dağıt topla arp

1. GİRİŞ

Hizmet sektörü, lojistik firmaları, dağıtım optimizasyonuna ihtiyaç duyulan diğer sektörlerde hizmetin devamlılığı, müşterilerin artan beklentilerini karşılayabilmek, değişen hizmet yapıları ve önceliklendirmeler gün geçtikçe çözümlenmesi zor problemler haline gelmiştir. Bu kapsamda çalışan firmalar bu problemlerin çözümlerine ciddi emek harcayarak etkili ve maliyeti düşük çözümler üretmek durumunda kalmıştır. Üretilen çözümlerin hem müşteri memnuniyeti artırması hemde en optimal maliyetle yapılması şirketlerin ana odak noktasını belirlemektedir.

Gezgin satıcı problemi ya da araç rotalama problemi şirketlerin bu kapsamdaki sorunlara çözüm bulmak için en fazla uğraştıkları konulardan bir tanesidir. Posta dağıtımının yapılması, Beyaz eşya servis hizmetinin verilmesi, online alışveriş uygulamalarındaki ürünlerin müşteriye etkin bir şekilde ulaşması, telekomünikasyon servis sağlayıcılarının müşterilerinin hizmetleri için sundukları teknik servis hizmeti vb. dağıtım işlemlerinde maliyet ve müşteri memnuniyetini sağlayacak en önemli konulardan birisi araç rotalama problemidir.

Araç rotalama probleminin çözümlerinde ana başlık olarak kesin ve sezgisel yöntemler olmak üzere 2 farklı yöntem geliştirilebilmiştir. Kesin çözümler en makul çözümü geliştirmek ve bunu diğer çözümlere göre en iyi olduğunu ispatlamak üzerine kurgulanmış çözümlerdir. Fakat bu çözümlerde karmaşık ve çok değişkenli problemler için beklenilenin altında kalmıştır. Sezgisel çözüm yöntemleri ise kesin çözüm üreten yöntemlere göre süresel olarak daha kısa zamanda çok daha kaliteli, efektif çözümler ortaya koyabilmiştir. Bu yöntemlerden birisi olan Guguk kuşu algoritması 2009 yılında guguk kuşlarının sezgisel uçuş şekillerine göre geliştirilmiş ve 2014 yılında gezgin satıcı problemi için uygulanmış, verimli ve başarılı sonuçlar elde edilmiştir.

Şirketlerde bu ve diğer problemlerin çözümlerini sağlayabilmek, sevk ve idare işleminin hızlı yapılabilmesi, müşteri memnuniyetinin sağlanabilmesi, önceliklendirilmenin doğru yapılabilmesi için personel çalıştırma zorunluluğu oluşmuştur. İnsan gücüyle bazı kritik kararların alınabilmesi, önceliklerin belirlenmesi, maliyetin azaltılarak müşteri memnuniyetinin artırılması hedeflenmektedir. Uzman sistemler ise belirlenen kısıtlar

dahilinde kurallara uygun olarak bu kararın insanlar yerine daha verimli bir şekilde verilmesini sağlamaktadır.

Bu tez çalışmasında yapay zekâ algoritması sayesinde uygun mesafe analizi yapılarak çıkan sonuçların Uzman bir sistem aracılığıyla şirketin belirlediği kıstaslar eşliğinde en optimal çözümleri üretebilmesi, fayda maliyet kıstaslarında yarattığı etkiler incelenmiştir.

Tezin ilk bölümünde tez ile ilgili yapılan literatür araştırmasına yer verilmiştir. İkinci bölümde Araç Rotalama Problemi ile ilgili detaylar verilmiş, problemin matematiksel karşılığı, bileşenleri, rotalama çeşitleri, rotalama yöntemi ile ilgili kullanılan yöntemler anlatılmıştır. Üçüncü bölümde yapay zekâ kavramı, yapay zekâ algoritmaları, guguk kuşu algoritması anlatılmıştır. Dördüncü bölümde Uzman sistemler hakkında bilgi verilmiş, avantaj ve dezavantajlarından bahsedilmiştir. Son bölümde önerilen algorithmadan bahsedilmiş, uygulamanın çalışma mantığı ve çıktıları paylaşılmış, sonuçların yorumlanması ve gelecek çalışmalar anlatılmıştır.

2. LİTERATÜR TARAMASI

Günümüzde şirketler devamlılıklarını sağlayabilmek, karlılık seviyeleri ve şirketlerinin değerini artırabilmek için birçok konuda optimizasyon yapmak zorunda kalırlar. Lojistik sistemleri optimizasyonu, yalnızca taşıma maliyetlerini azaltmak için değil aynı zamanda servis kalitesini artırmak ve çevreyi korumak için önemli olduğundan pek çok şirket için başlıca önem arz eden konulardır.

Dantzig ve Ramser (1959) tarafından ortaya atılan Araç Rotalama Problemi (ARP) bu optimizasyon sürecinin kritik adımını oluşturmaktadır. ARP noktalara belirli bir seri ile hizmet sunulması için araç filosunun izlediği optimal rota ile ilgilidir. ARP'nin genişletilmiş bir versiyonu olan Eş Zamanlı Topla Dağıt Araç Rotalama Problemi (ETDARP) Min (1989) tarafından önerilmiştir ve eşzamanlı olarak dağıt ve topla taleplerinin karşılanabilmesi düşünülmüştür. Bu sebepten araçların rota içindeki dolulukları değişir ve en optimal çözümün kontrolünün sağlanması zorlaştırmaktadır. Bu nedenle rota üzerindeki her noktadan sonra araç kapasitesi kontrol edilmeli ki böylece araç kapasitesi aşılmamalıdır. Meşrubat sektöründe, dolu olan şişelerin müşterilere teslimi ve boş şişelerin geri toplanması aynı araç tarafından gerçekleştirildiği için bu probleme iyi bir örnektir. Etdarp'nin bir uzantısı olan, zaman kısıtlı eş zamanlı topla dağıt araç rotalama problemi (ZKETDARP), rotayı gezecek araçların tekrar ana depoya ulaşmaları için bir zaman kısıtlaması içermektedir. Taşıma esnasında taşınan malzeme bozulmadan ulaşması için bir zaman kısıtı olduğundan, diğer bir gerçek yaşantı örneği süt taşımada meydana gelir.

Etdarp ve Zketdarp'nin matematiksel formülleri sırası ile Alfredo Tang Montané ve Galvão (2006), Polat ve diğ. (2015a) çalışmalarında bulunabilir. Etdarp problemi Min (1989) tarafından ortaya atıldığında bir küçük boyutlu kitap taşıma vakası üzerine çalışılmıştır. Üzerine çeşitli araştırmalar yapılan bu problem bir merkez kütüphane ve yirmi iki yerel kütüphane arasında kitap dağıtma, toplama operasyonlarını içermektedir. Literatürdeki bu çalışmalar kesin, sezgisel tek çözüm temelli ve popülasyon temelli metasezgisel olarak sınıflandırılabilir.

Ana problem NP-Zor bir problem çeşitinde bir problem olduğu bilindiğinden, çok az sayıda kesin yaklaşım düşünülmüştür. İlk kesin çözüm metodolojisi, dal-sınır yaklaşımı temelli kesin dinamik bir uygulama ve uzay konumunu gevşetme prosedürü ile Dell'Amico ve diğ. (2006) tarafından geliştirilmiştir ve sadece 40 müşteriye kadar, hesaplama zamanı yüksek

olmasına rağmen optimal çözebilmektedir. Diğer kesin çözüm yaklaşımları, dal ve kesme temelli algoritma Subramanian ve diğ. (2011) ve dal-sınır ve kesme temelli algoritma Subramanian ve diğ. (2013) 100 müşteriye kadar probleme çözüm sunabilmektedir. Kesin çözüm metodolojileri optimal çözümlerin kanıtlamasına ve iyi bilinen veri setleri için alt sınırlar yaratmasına rağmen, problem boyutu arttığında hesaplama karmaşıklığı da arttığından Tovey (2002) kesin çözüm algoritmaları büyük boyutlu problemleri kabul edilebilir zaman içerisinde çözemeyebilir. Bu sebepten, çözüm metodu için büyük oranda sezgisel ve metasezgisel yöntemler oluşturulmuştur. Dethloff (2001) Etdarp için bir matematiksel formülasyon önermiştir ve ters lojistik operasyonları için ekleme temelli bir sezgisel geliştirmiştir. Salhi ve Nagy (1999) Clarke ve Wright'ın kaydetme sezgisellerine (Clarke ve Wright 1964) yakın bir metotla aynı rota üzerine iki müşteri ilave eden ekleme sezgiseli üzerinde çalışma yapmışlardır. Nagy ve Salhi (2005) geliştirdikleri yaklaşımla geçici süreliğine uygun olmayan çözümlere de olanak sağlamışlardır ve 2-opt, 3-opt, shif, Exchange, reverse ve perturb gibi fonksiyonlardan yararlanıp kapasite aşımı halini bertaraf etmişlerdir. Sezgisel yaklaşımlar bir olasılık metodu barındırmadığı müddetçe, bir iyileşme rutini bulundurmaz bundan dolayı metot her zaman aynı netice ile sonuçlanır. Bu nedenle, ilk çözüm yapımında, sezgisel yöntemler ya saf rasgeselliği veya yenilemeli arama algoritmaları gömülü yapıcı bir alt metot olarak karşılaştırıldığında daha iyi bir başlangıç noktası sağlamak noktasında kullanılmaktadır.

Tek çözüm temelli sezgiseller Etdarp'nin çözümü için daha fazla tercih edilmiştir. Mevcut çözüm yöntemleri içerisinde, tabu arama yöntemi bu problemi çözmek için uygulamada en fazla kullanılan yaklaşımdır. Crispim ve Brandao (2005) tarafından uygulamaya alınan ilk tabu arama çalışmasında, tabu arama ve değişken komşuluk metodu birlikte kullanılarak, başlangıç çözümünde olası olmayan çözümlere izin verilmiş olup, çözüm uygun olana kadar ekleme ve takaslama metotları ile ceza uygulaması yapılmıştır. Rota-arası ve rota-içi komşuluk yapıları için *interchange*, *relocation*, *crossover* ve *2-opt* metotları kullanan Alfredo Tang Montané ve Galvão (2006), ceza şemalı bir tabu arama algoritması önerisi sunmuştur. Bianchessi ve Righini (2007) yerel arama sezgiseli temelli değişken komşuluk yapıları tabu aramayı nokta ve yay değişimini de kullanarak önermiştir. İteratif, müşteri tehcir, müşteri alışverişi, rota değişim işlemleri gibi komşuluk yapılarını kullanıp kazanca dayalı yapıcı sezgisel tarafından oluşturulan başlangıç çözümü geliştiren Zachariadis ve diğ. (2009), güdümlü yerel arama stratejileri ile birlikte tabu arama algoritması sunmuştur. İyi çözümleri hafızada tutan Zachariadis ve diğ. (2010) tabu arama temelli bir yaklaşım

sunmuşlardır. Wassan ve diğ. (2008) verimli bir dengeyi sağlamak amacıyla dinamik gerçekleştirilen aramada yoğunlaşması ve çeşitlendirilmesi arasında tabu listesi boyutunu kontrol ve *shift*, *swap*, *lokal shift* ve iyileştirme işlemlerini ters olarak komşuluk yapıları kullanan bir mekanizma ile bir tabu arama algoritması önermiştir. Komşulukları olasılık temelli seçen Ropke ve Pisinger (2006) büyük bir komşuluk arama yaklaşımı geliştirmiş ve birkaç geri çekmeli Etdarp'yi çözmüştür. Ağırlık kazanç temelli bir sezgiselle başlangıç çözümü oluşturan Zachariadis ve Kiranoudis (2011), tabu arama algoritması geliştirmiştir. Subramanian ve diğ. (2010), değişken komşuluk prosedürünü içeren bir çoklu-başlangıç meta sezgisel yaklaşımı, bir rasgele komşuluk sıralama prosedürü ve bir yineleyen lokal arama yapısı ile birlikte önerilmiştir. Etdarp problemini çözmek için rota-arası ve rota içi operatörlü lokal arama algoritması Jun ve Kim (2012), Li ve diğ. (2015) ve Avcı ve Topaloglu (2015) tarafından önerilmiştir. [1-26]

Son dönemlerde Polat ve diğ. (2015a) değişken komşuluk arama algoritması temelli bir değişken mekanizmasını, zaman limiti kısıtı olmayan Etdarp probleminin çözümü için önermiştir. Etdarp probleminin çözümü için Popülasyon temelli meta sezgisel algoritmalar da uygulanmıştır. Etdarp probleminin çözümü için parçacık sürü optimizasyonu algoritması, Ai ve Kachitvichyanukul (2009), Goksal ve diğ. (2013) tarafından önerilmiştir. Ai ve Kachitvichyanukul (2009) çözümün projeksiyonu için gerçek değerli kodlama mekanizması ve iyileştirme için ucuz ekleme sezgiseli ve *2-opt* metodu kullanırken, Goksal ve diğ. (2013) çözümün gösterimi için permütasyon kodlama kullanmış ve her iterasyonda iyileştirme için rasgele bir parçacığa değişken komşuluk ve lokal arama algoritmaları uygulamıştır. Gajpal ve Abad (2009) ve Çatay (2010) çalışmalarında Etdarp probleminin çözümü için karınca kolonisi optimizasyonu algoritması önerilmiştir, Clarke ve Wright'ın kurtarma algoritmasını (Clarke ve Wright 1964) her ikisi de kullanmış ve *2-opt*, müşteri ekleme, *interchange* ve *sub-path exchange* prosedürleri ile kısmi arama gerçekleştirmişlerdir. Salhi ve Nagy (1999) tarafından alana kazandırılan Zketdarp için karşılaştırmalı problem örnekleri ilgiye değer bir konu olmasına rağmen alanda pek ilgi çekmemiştir. Daha önce konu edilen çalışmalar içinde Etdarp örneklerine zaman kısıtını ekleyen çalışmalar, Ai ve Kachitvichyanukul (2009); Alfredo Tang Montané ve Galvão (2006); Çatay (2010); Dethloff (2001); Gajpal ve Abad (2009); Li ve diğ. (2015); Nagy ve Salhi (2005); Polat ve diğ. (2015a) ; Ropke ve Pisinger (2006); Wassan ve diğ. (2008). [1-26].

Kural tabanlı uzman enformasyonun güç optimizasyonunu zenginleştiren etkili optimizasyon araçlarını kapsayarak Uzman ve akıllı sistemler, karar destek sisteminde önemli başarılar elde edebilir. Bu çalışmada amacımız, Etdarp için etkili bir şekilde ve zaman sınırlaması kısıtlaması olmadan hızlı bir çözüm yaklaşımı belirlemektir. Çözüm yaklaşımı için Guguk Kuşu algoritmasının ürettiği sonuçlara Uzman sistemin müdahale edebildiği, hibrit bir yaklaşım ile en iyi çözümleri belirlemektir. Amacımız sadece en optimal çözüm değil, Uzman sistemin belirlediği kriterlere uygun en iyi çözümü oluşturmaktır. Uzman sistem metasezgisel algoritmadan aldığı sonuçla kendi kriterleri için oluşturduğu çözümün fayda maliyet analizi içerisinde başarılı olması hedeflenmektedir.



3. ARAÇ ROTALAMA PROBLEMİ

ARP, lokasyon olarak farklı noktalarda bulunan müşterilerine bir veya birkaç noktadan hareket eden araçların en optimize çözümle ulaşabilmesi için hazırlanacak rotaların problemine denir [27].

3.1. Araç Rotalama Problemi

ARP ile ilgili ilk çalışmalar Dantzig ve Ramser tarafından 1959 yılında yapılmış ve günümüze kadar üzerinde çok fazla optimizasyon çalışmaları yapıldığı görülmüştür [28].

Mevcut bir ARP’de farklı noktalardaki müşterilerin talepleri depolarda yer alan araçlar ile karşılanmaya çalışılmaktadır. Amaç bu işlemi yaparken müşteri gereksinimlerini etkin bir şekilde en az sürede, en optimum mesafe ve düşük maliyetle gerçekleştiren rotayı tayin etmektedir. Araç rotalaması yapılırken aşağıdaki hususlara dikkat edilmesi gerekmektedir:

- Ağ içerisinde yer alan noktalardan gelen taleplerin tamamı karşılanmalıdır.
- Ağ içerisinde bulunan varış noktalarına tek bir taşıt ile bir kez uğraması gerekir.
- Güzergâh başlangıç noktasında başlayıp, aynı noktada sonlanacak şekilde yapılmalıdır.
- Güzergâh üzerinde bulunan müşterilerin talep miktarları toplamı aracın kapasite toplamını geçmemelidir.
- Tüm güzergâh üzerinde sadece bir taşıt görevlendirilmelidir.
- Belirlenen güzergahın amacı taşıtların alacağı tüm mesafenin minimum seviyede olmasıdır [29].

3.2. Araç Rotalama Problemi Temel Bileşenleri

Araç rotalama problemlerinin ana bölümlerini; Dağıtım/ toplama noktaları, taşınacak malzemenin tipi, Araç filosu ve talep yapısı oluşturur.

Talep Yapısı

ARP’de talepler dinamik veya statik olabilir. Statik talepte önceden talepler bilinmekle birlikte, talep durumu dinamik olan durumlarda bazı noktalardaki talepler bilinebilsede, bazı

noktalarda ise araç güzergâh içerisinde ilerlerken belirlenmektedir. Bu çalışmada için statik yapısı tercih edilmiştir.

Malzeme Tipi

Taşıtlarla farklı malzemeler taşınır. Tehlikeli maddeler, gazete dağıtımı, Gıda maddeleri, çöp toplama gibi malzemeler teslim edilmesi basit olan ürünler olarak ifade edilir ve problemde ekstra bir zorluk oluşturmazlar. Öğrenci servisleri; eşitlik, etkinlik, güvenlik gibi ekstra bazı amaçlardan dolayı daha zorlu bir probleme sahiptir. Tehlikeli maddeleri bulunduran taşıtların güzergâhlarının oluşturulmasında şehrin coğrafi yapısı önem arz etmektedir [30].

Dağıtım/Toplama noktaları

Birçok ARP’de, Depo toplama noktaları iken, müşterilerin bulunduğu yer dağıtım noktalarıdır. Örnek olarak fabrikalardan toptancılara tüketim mallarının dağıtımı verilebilir.

Depolar çoğunlukla taşımacının güzergaha başladığı ve rotasını tamamlayıp geri döndüğü alandır. Problem depo sayısına göre gruplara ayrılır. Tek ve çok depolu diye isimlendirilir. Birden fazla başlangıç noktasının olduğu problemlerde, her bir başlangıç noktası kendine ait taşıtlar ile işlerini idame ettirebilir, problem bu durumda birkaç ayrı tek başlangıç noktalı ARP olur. Araç depodan, başka bir depoya yükleme/boşaltma gerçekleştirebilir. Bu kapsamda problemin bütün olarak değerlendirilmesi gerekmektedir.

Dağıtım noktalarının belirli ve önceden bilinmesi durumunda araçların hangi depolara hizmet sağlayacağı tanımlanmalıdır. Dağıtım noktaları diğer durumda potansiyel yerler içinden belirleneceğinden ötürü ekstra bir yerleştirme kararı gerekmektedir.

Dağıtım ve toplama noktaları bazı araç rotalama problemlerinde aynıdır. Örneğin bir şirketin sabah işe gidişinde personel servisi personelin evi için toplama noktası, iş yeri dağıtım noktası iken akşam dönüşte personel evleri dağıtım noktaları iş yeri toplama noktası/depo ise toplama noktasıdır [30].

Araç Filosu

Genel olarak araç rotalama problemlerinde taşıtların kapasitesinin belli olduğu ve genellikle araçların standart bir hacime sahip olduğu düşünülür. Filo standart değilse filodaki taşıtların taşıma hacimleri değişiklik gösterecektir. Bu kapsamda taşıt tipinin, hangi güzergaha gideceğinin tespit edilmesini, bu durumda ilave bir karar almayı gerektirmektedir. Araçların bunlar dışındaki özellikleri arasında yakıt tüketimi, taşınacak hızı ve malzemeye uygunluğu sayabiliriz. Diğer özelliklerin rotalama kararlarına dolaylı bir etkisi vardır.

3.3. Araç Rotalama Problemi Matematiksel Modeli

Araç Rotalama Probleminin formülize edilmiş hali aşağıdaki gibidir [31].

Parametreler:

Q_k : taşıt kapasitesi,

N : müşteri veya nokta sayısı,

q_i : i ($i > 0$)müşterisinin talebi,

d_{ij} : i ve j müşterileri arasındaki uzaklık,

Değişkenler:

x_{ij} , $i \neq j$: araç i " den j " ye gidiyorsa "1", yoksa "0" olmak üzere,

($i, j \in \{ 0, \dots, n \}$ ve "0" başlangıç deposu iken)

Amaç Fonksiyonu:

$$\text{Min } \sum_{i=0}^n \sum_{\substack{j=0 \\ i \neq j}}^n d_{ij} x_{ij} \quad (3.1)$$

Kısıtlar:

$$\sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1 \quad \forall i, j \in \{1, \dots, n\} \quad (3.2)$$

$$\sum_{i=0}^n \sum_{j \in S} x_{ij} \leq |S| - 1 \quad (3.3)$$

$$\sum_{i=0}^n \sum_{\substack{j=0 \\ T \in S}}^n x_{ij} \leq |T| - n \quad (3.4)$$

Problemimizin amaç fonksiyonu için Eş.3.1. kullanılmaktadır. Burada Eş.3.2. kısıtları gidilen müşteriden ayrılma öncelikleridir. Eş.3.3. önceliği depodan başlamayan ve depoda bitmeyen turları devre dışı bırakmayı sağlayacaktır.

Son olarak, Eş.3.4. kısıtı taşıtlardaki yük durumunlarını denetlemeyi sağlayacaktır. Bu kısıt, başlangıç noktası dahil olmak üzere her T müşteri kümesine eklenir. Bu kümelerin her biri,

$$\sum_{i \in T} q_i < Q_k$$

Şartını sağlaması gerekir.

Ayrıca n ise aşırı doluluğu ortadan kaldırmak için T grubundan çıkarılması gereken en az sayıda müşteriyi ifade eder. [31]

3.4. Rotalama Türleri

Rota süresi, başlangıç noktasından çıkan bir taşıtın depoya geri dönüşü için ihtiyaç olan zaman olarak tanımlanmıştır.

3.4.1. Rota süresi sınırsız rotalama problemleri

Tek araçlı gezgin satıcı problemleri, Rota süresi sınırsız güzergâh problemleri arasında verilebilecek en iyi örnektir. Bu sorun, bir taşıt ile bir "Hamiltonion tur" oluşturulması esasına temel almaktadır. Bir Hamiltonion tur, bir $H = [D, B]$ ağında, hiçbir düğümü atlamadan tam bir turu ifade etmektedir. Bu ağda, (D), noktalar kümesidir; (B) kollar kümesini ifade etmektedir. Problemin amacı, tam turun maliyetini ya da tam tur toplam mesafesini minimize etmektir. Belirtilen problemde, noktalara gidilecek zamanlama ve toplam seyahat süresi ile ilgili girdiler ve kısıtlar bulunmamaktadır [32].

3.4.2. Rota süresi sınırlı rotalama problemleri

Rota süresinin sınırlandırıldığı problemlerde, taşıtların aldıkları toplam mesafe minimize edilmektedir. Depodan çıkan araçlar belirli noktalara (düğümlere) gittikten sonra aynı başlangıç noktasına geri gelme durumundadırlar. Bir aracın bir başlangıç noktasından diğer

başlangıç noktasına olan güzergâh süresi, en maksimum güzergâh süresini geçmemelidir. Bu kısıt, noktalar havuzuna gidecek olan taşıt sayısının tespit edilmesinde kritik bir öneme sahiptir.

Rota süresi ile beraber taşıt miktarında kısıtlı olduğu problemlerde, bazı noktalara gidilmemesi durumuna rastlanılabilir. Eğer bütün noktalara bir kere gidilmesi ve taşıt miktarını artırılması mümkün değilse, maksimum rota süresini aşmak gerekebilir. Bu durumda geçilen her birim zaman için ceza maliyeti hesaplanıp, toplam ceza maliyetinin azaltılması için optimizasyon yapılabilir. Bu yaklaşım taşıt sayısının kısıtlı olmasını bir "katı kısıt", rota süresinin belirli olmasını bir "gevşek kısıt" yapar. Bir "katı kısıt" mutlaka yerine getirilmesi gereken, bir "gevşek kısıt" ise uyulması belirtilen kısıtlardır. Rota süresinin sınırlandırılması problemin yapısını, önemli düzeyde etkilenmektedir [32].

3.4.3. Zaman pencereci rotalama problemleri

Zaman pencereci ARP'ler, olabilecek en düşük maliyetli rota gruplarının belirlenmesi, merkez bir başlangıç noktası belirlemeyi ve gelen isteklere bağlı olarak araçların rotalamasını içermektedir. İlk durumda, yolcular, mevcut hacimler dikkate alınmadan taşıtlar vazifelendirilmektedirler.

Zaman penceresi, yumuşak veya sert olarak ayrılırlar. Sert bazlı olaylarda, eğer bir taşıt gidilmesi gereken noktaya belirlenen zamandan önce ulaşırsa, taşıt müşteri gelinceye kadar beklemedilir. Ancak, taşıtın her bir noktada, hizmete başlaması için belirlenen süreden sonra varılmasına izin verilmemektedir. Yumuşak bazlı durumlarda, zaman kısıtları bir maliyet belirlenerek aşılabilmektedir [33].

Zaman bazı, genellikle doğal olarak esnek zaman çizelgeleriyle çalışan yapılarda karşımıza çıkan bir durumdur. Randevulu ev içi hizmetler, posta iletileri, işyeri servisleri rotalama ve çizelgeleme Sert zaman bazlı spesifik örnekleri oluşturur [33].

Zaman pencereleri iki kümede incelenebilir:

- Çift sınırlı zaman pencereleri,
- Tek sınırlı zaman pencereleri

Çift sınırlı bir zaman penceresinde, noktada olunabilecek ilk ve son zaman belirlenmiştir. Bir taşıt bir noktaya belirlenen zaman kısıtları içinde gitmelidir. Belirtilen zamandan önce noktaya giden bir taşıtın, belirtilen zamana kadar bekleyeceği esas alınır. Aracın, belirlenen zamandan sonra noktaya gitmesine müsaade edilmez. Tek sınırlı zaman pencereleri, uğranılabilecek "ilk zamanı belirli olanlar" ve "son zamanı belirli olanlar" şeklinde iki türde belirlenmiştir [32].

3.4.4. Eşzamanlı rotalama ve çizelgeleme problemleri

Bir toplu taşıma seferi için ilk başlangıç zamanı /yeri ve son bitiş zamanı/yeri önemlidir. Filodaki her bir seferin, hangi araç ile gerçekleştirileceği belirlenmesi işlemine, "eşzamanlı rotalama ve çizelgeleme" olarak ifade edilmiştir. Eşzamanlı araç rotalama ve çizelgeleme, her bir taşıtın gerçekleştireceği seyahat “sıralamasını” gösteren bir iyileştirme problemi olarak ifade edilmiştir [32].

- Bir eşzamanlı rotalama ve çizelgeleme probleminde, genel olarak dikkat edilmesi gereken kısıtlar şunlardır:
- Belirli bir zaman çizelgesinde her bir görev tek bir araç ile ve sadece bir kez yapılmalıdır.
- Bir başlangıç noktasından ayrılan her bir taşıt, görevlerini gerçekleştirdikten sonra tekrar aynı noktaya geri gelmelidir.
- Depolarda bulunan taşıtlar ile görevlerin tümü gerçekleştirilmelidir.
- Görevlere, görevleri için tayin edilen araç tipleri ile hizmet edilmelidir.
- Her bir taşıt, belirli bir zaman dilimi sonrasında başlangıç noktasına geri gelmelidir.

Problemler, buradaki kısıtların bir kısmını veya tamamını içermektedir. Bununla beraber, problemin yapısı değişik sınırların göz önünde bulundurulmasını gerekli kılabilir. Bu kısıtların bazılarının katı, bazılarının ise gevşek kısıtlar olması mümkündür [32].

3.4.5. Kapasiteli araç rotalama

ARP'nin en genel hali (Kapasiteli Araç Rotalama Problemi (KARP)) olan bu problem; merkezi bir başlangıç noktasında dağıtım işini gerçekleştiren t tane T kapasiteli F hızına sahip taşıtların o belirlenen bölgede homojen bir şekilde ayrı yerlere dağılmış m tane

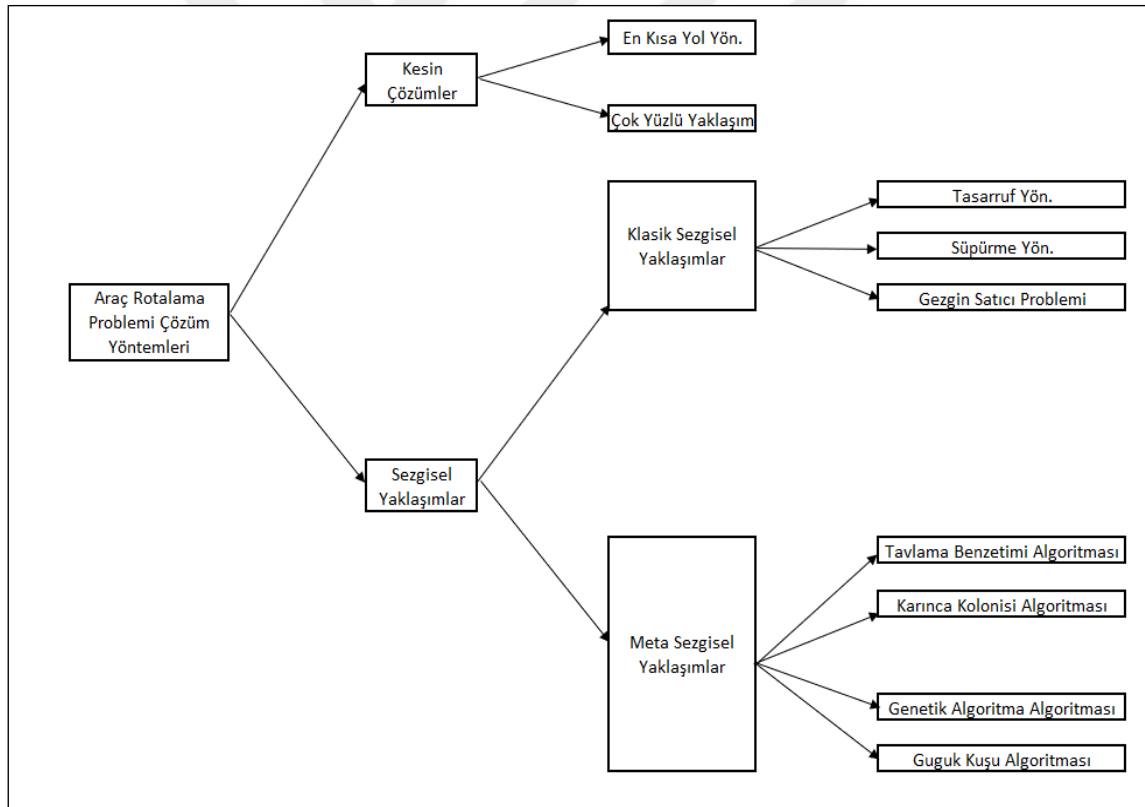
müşterinin taleplerini en az D maliyetle karşılaması için yapması gereken en uygun P rotalarının tespit edilmesi olarak ifade edilmiştir. Noktalar arasındaki uzaklıklar karşılıklı gidiş-geliş olarak tahmin edilmektedir. Taşıtların bekleyen tüm taleplere karşılık vermesi durumunda k tane rota tespit edilir. Güzergahların trafik durumları ölçümlenmek istenirse kapalı olan yolları gösteren düğümler düşünülmeyle o anki ölçüme göre rotalar tespit edilebilir. Ayrıca yollar çift yönlü olacak şekilde kabul görse de tek yönlü olarak da düzenlenebilir [34].



4. ARAÇ ROTALAMA YÖNTEMLERİ

Değişen çevre faktörleri, artan rekabet koşulları ile birlikte oluşan kısıtlar Araç rotalama problemlerinin çözümünü daha da zor hale getirmektedir. Araç rotalama problemlerinde farklı hacimlere sahip çok sayıda taşıt, Zaman aralıkları, rota üzerinde istenilen seyahat süresi, taşıt sürücülerinin dinlenme zamanları gibi kısıtlar karşımıza çıkmaktadır [32].

Bir ağ için ARP o problem için hazırlanmış bir yöntem ile çözülebilmektedir. En kısa yol yöntemi en fazla tercih edilen çözüm metotlarından birisidir. Süpürme metodu, gezgin satıcı yöntemi ve tasarruf metotları da sıkça kullanılan diğer yöntemlerdir. Kendine özgü amaçlar ve kısıtlar taşıyan bu yöntemler, karmaşık yapılı, çoklu orjin ve varış düğümlerine sahip ARP doğrusal programlama yöntemi ile çözülmektedir.



Şekil 4.1. Araç rotalama yöntemleri

4.1. En Kısa Yol Yöntemi

Genel amacı müşteriler arasındaki uzaklık ölçümünün doğru bir şekilde yapılması ve taşıt hacimlerine göre müşteri atanması olan En Kısa Yol Yönteminin süreçsel aşamaları şu şekildedir (Breedam, 2002):

1. Merkezi birimden güzergaha bağlanmaktadır. Merkeze en yakın müşteriye ilk araç tahsis edilir.
 2. Güzergâh üzerinden belirlenen müşteriye en yakın, daha önce güzergaha dahil edilmemiş müşteriler gözlemlenir. Eğer merkeze yakın birden fazla müşteri varsa, ikisi içinde ayrı bir rota çizilerek ayrı bir çözüm yolu oluşturulur.
 3. Eğer müşteri doğrudan merkezi nokta ile irtibatı bulunmuyorsa (Oluşturulan rota içerisinde müşteriler başlangıç noktasından sonra bulunmuyorsa) ve müşteriye en yakın diğer müşteri ile ana nokta eşit uzaklıkta ise rotalama için yine iki ayrı çözüm oluşturulur.
- Güzergahta ilk olarak müşteri başka müşteriye gidişte taşıt hacim kısıtını sağlıyorsa bağlanır. Ardından çözüm kümesinde yeni bir yol oluşturularak müşteri doğrudan merkezi noktaya gönderilir.
4. Ayrı ayrı hesaplanan çözümler arasından en uygun çözüm seçilmektedir.

Ancak bir başlangıç noktasından birden fazla bitiş noktasına ya da çok sayıda başlangıç noktasından çok sayıda bitiş noktasına en az mesafenin bulunması durumunda yapılacak işlemler çok karmaşık bir hal almaktadır. Böyle zamanlarda çözüm doğrusal programlama ile yapılmaktadır [35].

4.2. Süpürme (Sweep) Yöntemi

1974 yılında Gilbert ve Miller tarafından geliştirilen Süpürme algoritmasında depo merkezli bir doğrunun döndürülmesi ile rotalarda yer alacak müşterilere ulaşılmaktadır. Döndürme sırasında doğrunun üzerinden geçtiği müşteriler bir gruba ayrılır ve kapasite veya uzaklık kısıtı aşıldığında grup kapatılarak yeni bir grup ile ilerlenir. Genel olarak Gezgin Satıcı

Problemi (GSP) gibi çözülerek rotalar belirlenir ve oluşturulan nokta gruplarına merkez depo da eklenir [36].

Süpürme Yönteminde “müşterilerin koordinatları Öklidyen forma (x, y) yerine, θ açı ve ρ doğru uzunluğu olmak üzere polar formatta (θ_i, ρ_i) tutulur. Böylece talep noktaları θ açısına göre küçükten büyüğe sıralanarak, hacim ve mesafe kısıtları göz önünde tutularak sınıflandırılır.” Genel olarak Süpürme Algoritmasının çözüm adımları aşağıdaki belirtilmektedir [37].

- 1- Coğrafi konum üzerinde başlangıç ve müşteri noktalarının yeri belirlenir ve koordinatlar polar formata (θ_i, ρ_i) çevrilir. Rotaya atanmamış rastgele bir taşıt tespit edilir.
- 2- Saat yönünün tersine doğru Depodan yatayla 0° açı ile başlanarak taranmaya gerçekleştirilir. Tarama sırasında bir müşteri ile karşılaşırsa ve müşterinin talep ölçüsü aracın hacmini aşmıyorsa müşteri taşıta tayin edilir. Karşılaşılmazsa benzer devam edilir. Eğer taşıta her iki yönde de müşteri tayin edilmiyorsa, diğer araçla devam edilir.
- 3- Eski taşıtın bulunduğu yerden gözlemleme sürdürülür. Eğer daha evvel güzergahı belirlenmeyen müşteri ile karşılaşırsa ve müşterinin taşıta atanması hacim kısıtının açılmasına sebep vermiyorsa, müşteri taşıta atanır. Bu süreç boşta bir talep noktası bulunmayınca kadar (tüm müşteriler için güzergâh belirleninceye kadar) devam eder.
- 4- Her lokasyona taşıtlar tayin edildikten sonra kümeler elverişli bir biçimde optimize edilerek rotalar tespit edilir.

Süpürme yöntemi Tasarruf algoritmasına göre çalıştırılması daha kolay olmasıyla birlikte, çoğu araç rotalama problemi için kesinlik ve hız bakımından Tasarruf metodu daha kullanışlıdır. Bunun nedeni grupta gerçekleştirilirken noktalar arasındaki mesafenin göz önünde bulundurulmaması ve güzergahların belirlenen bir lokasyondan başlayarak yapılmasıdır. Ek olarak bu yöntemin yeni kısıtlarla bütünleşik olması zor olmakla birlikte yalnızca iki boyutlu Öklidyen sistemde çözümler üretmesi uygulanmasını sınırlı hale getirmektedir [36].

4.3. Gezgin Satıcı Problemleri

Bir satış faaliyeti gösteren personelin m adet lokasyonu en az sürede dolanmasını gerçekleştirecek şekilde lokasyon sırasını belirleme problemine gezgin satıcı problemi denmektedir. GSP’de tekrar başlangıç noktasına geri dönmek üzere her bir şehire sadece bir kere ve en kısa yoldan rotasını tamamlaması beklenir.

Gezgin satıcı problemi kombinasyonel optimizasyon bir NP-Zor bir problem olup, yöneylem araştırmalarında ve teorik bilgisayar bilimlerinde çalışılmıştır. Listesi verilen Şehirler ve ikili mesafelerde, amaç tam olarak her şehirde en kısa turu bularak bir kere ziyaret etmektir.

Bir matematik problemi olarak 1930 yılında formüle edilen sorun, optimizasyon konusunda en çok çalışılan problemlerden birisidir. Hesaplama olarak zor olmakla birlikte problem, birçok sezgisel ya da kesin algoritma ile probleme çözümler sağlanmıştır.

GSP ve ARP lokasyon sayısı fazlalaştıkça problemin zorluğu ve Hesaplama karmaşıklığı hızlıca artmaktadır. M Şehirli bir problemde olurlu turların sayısı $(m-1)!$ ’dir. İlk Şehir için $(m-1)$ tane, ikinci Şehir için $(m-2)$ tane opsiyon bulunmaktadır.

Çoğu GSP sezgisellerinden ARP için geliştirilen sezgiseller belirlenmiştir. Gezgin satıcı sezgiselleri; tur geliştirici algoritmalar, tur kurucu algoritmalar ve karışık algoritmalar olmak üzere üç grupta tanımlanır. Tur kurucu algoritmalar, mesafe matrisinden ötürü gerçeğe çok benzer güzergahlar meydana getirir. Tur geliştirici algoritmalar mevcut bir makul çözümü şekillendirirler. Karmaşık yöntemlerde ise tur kurucu yöntemlerden biri ile bir başlama çözümü tespit edip, tur geliştirici yöntemlerin bir veya birkaçı ile mevcut çözümü şekillendirirler. [30]

GSP’nin matematiksel açıklamaları aşağıdaki gibi ifade edilmiştir.

$$\text{Min } \sum_{i=1}^n \sum_{j=1}^n d_{ij} \cdot x_{ij} \quad (4.1)$$

$$\sum_{i=1}^n x_{ij}=1 \quad j = \{1, \dots, n\} \quad (4.2)$$

$$\sum_{j=1}^n x_{ij}=1 \quad i = \{1, \dots, n\} \quad (4.3)$$

$$x = (x_{ij}) \in S \quad (4.4)$$

$$x_{ij} = 0 \text{ veya } 1 \quad j = \{1, \dots, n\} \quad i = \{1, \dots, n\} \quad (4.5)$$

n : Dağıtım ağı içindeki düğüm sayısı

d_{ij} : i, j yayının uzunluğu

S : Alt tur eliminasyon kısıtı

$$x_{ij} = \begin{cases} 1, & \text{eğer } i, j \text{ yayı turun içinde ise} \\ 0, & \text{aksi takdirde} \end{cases}$$

GSP için belirlenen amaç fonksiyonu Eş.4.1. de belirtilmiştir. Eğer yalnız Eş.4.2. ve Eş.4.3. küme kısıtı hesaba dahil edilirse problem bilinen atama problemine girmektedir. Eş.4.4. kısıtı esas zorluk oluşturan kısımdır. Bu kısıt, atama kısıtlarını yeterli kılan çözümlerini engellemek için kullanılan alt tur eliminasyon kısıtından oluşmaktadır. Eş.4.5. kısıtı ise başlangıç ve araçların herhangi birine gidileceğini belirtmektedir.

$$S = (x_{ij}): \sum_{i \in T} \sum_{j \in T} x_{ij} \geq 1$$

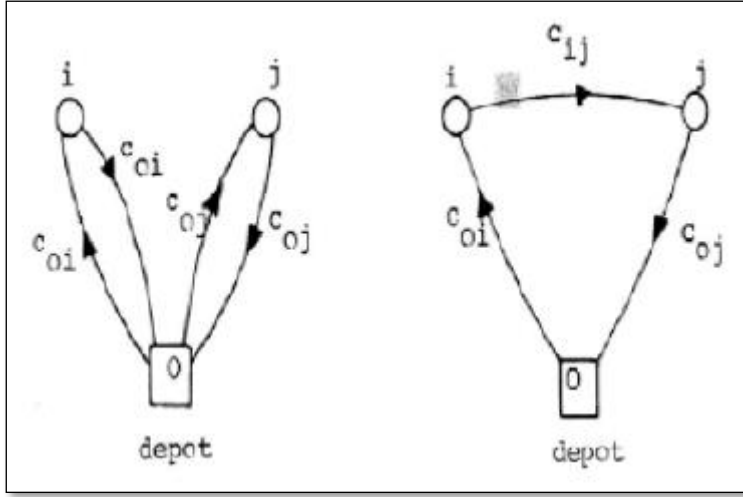
(V 'nin boş olmayan her bir T alt kümesi için)

V : Düğüm Seti

Alt tur eleme kısıtlarının ilk partisi, noktaların hepsi için uygun bir alt kümesinin, aynı alt kümenin tamamlayıcısı ile alakalı olduğunu taahhüt etmektedir. Bu durum, sistemin diğer noktalar ile ilişkili olmayan herhangi bir uygun alt kümenin, diğer düğümlerden bağımsız olarak bir alt tur meydana gelmesine engel olur [38].

4.4. Tasarruf Algoritması

ARP problemlerini çözmek için geliştirilen yöntemlerden birisi de Tasarruf yöntemidir. Bilinen sezgisel yöntemlerden genel olarak en iyisi olan Tasarruf Yöntemi 1964 yılında Clarke ve Wright tarafından geliştirilmiştir. Bu algoritma, her bir nokta ikilisi arasındaki maliyet tasarrufunu hesaplamak suretiyle oluşur. Maliyet tasarrufları hesaplanarak iki nokta arasına bir nokta ilave edilir. Şekil 4.1'de görüldüğü gibi i ve j . müşteri farklı turlardadır, i . müşteriden sonra j . müşteri ilave edilerek turlar birleştirilmektedir.



Şekil 4.2. Tasarruf yöntemindeki müşteri birleştirilmesi

$$S_{ij} = (c_{0i} + c_{i0} + c_{0j} + c_{j0}) - (c_{0i} + c_{ij} + c_{j0}) = c_{i0} + c_{0j} - c_{ij} \quad (4.6)$$

Denklem Eş.4.6.'deki tasarruf miktarı (S_{ij}) müşteri i ve j . müşterinin farklı turlarda değil aynı turda hizmet almasından doğan bir maliyetten tasarruf edilmesi sağlamaktadır. Bu maliyet tasarrufu iki bağımsız turun entegre edilmesinden oluşmaktadır. Tasarruf yönetiminde her zaman, en büyük optimizasyonu sağlayan (i,j) ikilisi, müşteri istekleri ve araç hacmi kısıtları göz önünde bulundurularak tercih edilir. Bütün müşterilerin araçlara atanmasına kadar bu işleme devam edilir [39].

Clarke-Wright'ın tasarruf yöntemi, kolay anlaşılması ve diğer araç rotalama problem algoritmalarına göre esnetilebilir olabilmesi sayesinde yaygın bir kullanım alanı bulunmaktadır. Bu yöntem Gaskell (1967) ve Yellow (1970) gibi pek çok araştırmacı tarafından üzerinde çalışarak, geliştirilmiştir [40].

4.5. Doğrusal Programlama Yardımıyla Araç Rotalama

Karışık yapıli ağ yapılarında araç rotalama işlemi için yukarıda belirtilen rotalama metotları yeterli olmamaktadır. Bundan dolayı da araç rotalamanın ana unsurları olan toplam güzergâh mesafesinin minimize edilmesini mümkün kılmamakta, bu nedenle servis seviyesi düşmekte ve yüksek seviyede maliyetler çıkmaktadır. Benzer durumlarda doğrusal programlama araç rotalamada uygun bir sonuca varmak amacıyla kullanılmaktadır.

Doğrusal programlama yardımıyla araç rotalamada kısıtları müşteriler, araçlar, depo ve müşteriler ile başlangıç noktası arasındaki ağ bağlantıları oluşturmaktadır. $N+1$ adet müşteri noktası ve K adet araç bulunmaktadır. Başlangıç noktası 0, müşteri olarak açıklanır. Araçlar rotaya başlangıç noktasından başlayarak, müşteri noktalarını ziyaret eder ve depoya tekrar ulaşırlar. Ağ içerisindeki rotaların miktarı kullanılan araçların miktarı ile aynıdır. Bir araç sadece bir rota üzerinde faaliyette bulunabilir. Rotalama oluşturulurken şebeke içerisindeki tüm müşterilerin ziyareti taşıtlardan sadece biri ile yapılır. Tüm araçlar aynı kapasitededir ve müşteri talepleri değişkenlik gösterebilmektedir. Aracın kapasitesi rota üzerinde bulunan tüm müşterilerin taleplerinin toplamı ile eşit ya da az olmalıdır. Bunun anlamı taşıtların hacimleri üzerinde yüklenme yapılmamasıdır. Doğrusal programlama sayesinde bir ARP şu şekilde formülize edilebilir:

Parametreler:

K : Toplam araç sayısı

N : Toplam müşteri sayısı

C_{ij} : i noktasından j noktasına taşıma maliyeti

M_i : i müşteri noktasındaki talep

q_k : k aracının kapasitesi

Değişkenler:

x_{ij} : eğer k aracı i noktasından j noktasına giderse 1, aksi halde 0

$$\text{Min } \sum_{i=0}^N \sum_{j=0, j \neq i}^N \sum_{k=1}^K C_{ij} \cdot x_{ijk} \quad (4.7)$$

$$\sum_{k=1}^K \sum_{j=1}^N x_{ijk} \leq K \quad i = 0 \text{ için} \quad (4.8)$$

$$\sum_{j=1}^N x_{ijk} = \sum_{j=1}^N x_{ijk} \leq 1 \quad i=0 \text{ ve } k \in 1, \dots, K \quad (4.9)$$

$$\sum_{k=1}^K \sum_{\substack{j=0 \\ j \neq i}}^N x_{ijk} = 1 \quad i \in 1, \dots, N \quad (4.10)$$

$$\sum_{k=1}^K \sum_{\substack{i=0 \\ i \neq j}}^N x_{ijk} = 1 \quad j \in 1, \dots, N \quad (4.11)$$

$$\sum_{i=1}^N M_i \sum_{\substack{j=0 \\ j \neq i}}^N x_{ijk} \leq q_k \quad k \in 1, \dots, N \quad (4.12)$$

$$x_{ijk} \in \{0,1\} \quad (4.13)$$

Yukarıdaki doğrusal programlama modelinde Eş.4.7. modelin amacını yani toplam seyahat mesafesinin ve toplam rotalama maliyetinin en düşük seviyede tutulmasını açıklamaktadır. Eş.4.8. oluşabilecek maksimum K rota sayısını ifade etmektedir. Eş.4.9. her bir güzergahın başlangıç noktasından başlayıp yine başlangıç noktasında sona erdiğini ifade etmektedir. Eş.4.10. ve Eş.4.11. müşteri lokasyonlarının her birinin yalnızca bir taşıt ile sadece bir defa ziyaret gerçekleştirildiğini belirtmektedir. Eş.4.12. üzerinde bulunan toplam talep hacminin rotanın tamamlanması sonucu ortaya çıkacak olan taşıtın toplam hacminden fazla olmayacağını belirtmektedir. Eş.4.13. X_{ijk} değişkeninin 0 ya da 1 değerini alabileceğini ifade etmektedir [41].

Hizmet faaliyeti için sabit sayıda yere gitmesi gereken taşıtların etkin ve verimli bir biçimde kullanılması Hizmet sektöründe karşılaşılan en önemli sorunlardan birisidir. ARP olarak bilinen bu tür problemlerde araç hacmi kısıtı içerisinde toplam gezinti mesafesini ve dolayısıyla da toplam maliyeti en aza indirecek uygun rotayı tespit etmek gerekmekte olup, Araç rotalama problemi uzun bir süre hem araştırmacıların hem de uygulamacıların dikkatinde olmuştur. Bunun sebebi problemin zor ve aynı zamanda uygulama alanının geniş olmasından kaynaklanmaktadır [29].

4.6. Tavlama Benzetimi Yöntemi

Kirkpatrick ve arkadaşları tarafından ilk kez 1983 yılında bir optimizasyon tekniği olarak ortaya çıkmış ve Metropolis Algoritması olarak da adlandırılmıştır (Tosun vd., 2009:2). Isıtılan metallerin soğutulması işlemine benzetilen bu yöntem bu haliyle doğal bir yöntem olarak görülmektedir. Tavlama benzetimi, mevcut bir çözümde sürekli olmayan bozulmalar üreten arama hareketlerini kabul eden olasılıksal bir meta-araştırma algoritması olarak yerel

bir minimum noktadan uzaklaşmanın bir yolu olmuştur. (Oliveira vd. 2006:2). Arıkan ve Erol(2005), esnek imalat sistemlerinde parça seçimi ve makine yükleme problemini, Şahin(2008), dinamik ses düzenleme problemine, Zhan vd., (2017), gezgin satıcı problemi ile ilgili çözümü tavlama benzetimi metodu ile çözüm bulmaya çalışmışlardır.

4.7. Karınca Kolonisi Yöntemi

Karınca Kolonisi Algoritması, 1991 yılında Colorni ve arkadaşları tarafından geliştirilmiş bir meta-sezgisel algoritma çeşididir. Algoritmanın dayanağı, ismini de aldığı karıncaların gerçek yaşamdaki durumudur. Karıncaların beslenme alışkanlıklarından yola çıkan metot, karıncaların yuvalarından gıda kaynağına giden en kısa yolu, görsel belirti olmadan bulabilmeleri temeline dayanmaktadır

Aday çözümler bir dizi değişken, rastlantısal yöntemlerle üretilir, daha önce bilinen çözümler kullanılarak mevcut çözümler olasılıksal olarak revize edilir. Karınca Kolonisi Algoritması, ortaklaşa böcek davranışlarından esinlenen algoritmalar arasında en çok tercih edilen yöntemlerden birisidir. Luan vd.(2018), tedarikçi seçim problemini çözmek, Kumar vd., (2019), insansı robotların yolunu optimize etmek, Xu, (2019), turist rehberlerinin en uygun rotayı bulmalarını elde etmek için karınca kolonileri algoritması kullanarak optimizasyon çözümleri gerçekleştirmişlerdir.

4.8. Genetik Algoritma Yöntemi

Genetik ve evrimsel algoritmalar, Holland (1975) ve Rechenberg (1973) tarafından, Darwin'in 1859 yılında öne sürdüğü evrim teorisinden ilham alınarak şekillendirilmiştir. Genetik Algoritma (GA), doğanın temel prensiplerini uyarlayarak, çok ayrı çeşitteki problemlerde başarıyla gerçekleştirilmiştir. (Rathlauf, 2006:15-16). Değerlendirme, seçim ve gen oluşumu olmak üzere üç ana kuralı bulunan GA, her biri belli olan bir soruna ilişkin muhtemel bir çözümü simgeleyen ayrı ayrı karakter dizileri popülasyonu ile çalışır (Louis ve Li, 2000:205). Her nesilde farklı bireyler alınır ve yeni bir popülasyonun oluşması için mutasyona uğratılır bu şekilde popülasyon algoritmanın sonraki tekrarlarında uygulanır. Probleme ilişkin değişkenlerin kodlanması ve Algoritmaya bildirilen bir amaç fonksiyon ile çeşitli karmaşık problemlere GA sayesinde çözüm oluşturulabilir. (Çalışkan vd, 2016:22; Goldberg, 1989:1-2). Genetik algoritma ile finans, üretim, araç rotalama problemi, Çinli postacı problemi, arama problemleri, çizelgeleme, mekanik öğrenme, otomatik

programlama ve bilgi sistemleri, ekonomik ve sosyal sistemler gibi birçok problemin çözümü bulunmaya çalışılır. (Emel ve Çağatay, 2002:138-147).

4.9. Guguk Kuşu Arama Algoritması

Yang ve Deb (2009) tarafından Optimizasyon problemlerinin çözümü için geliştirilen GKA algoritması, Quaarab ve ark. (2014) tarafından GSP probleminin çözümü için uygulanmıştır. 2009'da geliştirilmesiyle beraber, planlama, yönlendirme, rastgele değişkenleri seçme, sınır ağlarında izleme ağırlıkları, üretim, durum izleme, sağlık izleme vb. çok sayıda uygulamada tercih edilmiştir. Çeşitli çalışmalarda bu algoritmaya temel alan yeni metasezgisel metotlar tavsiye edilmiştir. Bu algoritma, bazı guguk kuşu türlerinin zorunlu kuluçka paraziti davranışı ile bazı kuşların ve meyve sineklerinin Levy uçuş davranışının bereber kullanılmasına esas almaktadır. (Quaarab ve ark., 2014).

5. GUGUK KUŞU ALGORİTMASI VE YAPAY ZEKÂ

Zekâ; gerçekleri, tavsiyeleri ve bunlar arasındaki bağlantıları anlama, düşünme, akıl yürütme ve kavrama yeteneklerinin tümüdür. Yapay zekâ ise, adından da anlaşılacağı gibi bilgisayara düşünme yetkinliği kazandırır. Yapay zekâ; yeni bilgileri yakalar, insanın düşünme ve öğrenme sürecini taklit ederek harekete geçer ve bilgileri gelecekte kullanıma hazır hale getirir.

Yapay zekâ için en iyi örnek, insandır. Çünkü insan, son derece donanımlı ve kendini sürekli güncelleyebilen öğrenme makinesidir. İnsanın yanlışları, öğrenme limitleri ve unutma şekli dünyanın en geniş, en esnek ve en verimli veri tabanını ve bilgi alma sisteminin oluşmasını sağlayacak niteliktedir. (James, 1984).

İnsan beyni, beynin çalışma şeklini değiştirmeden ve beyindeki diğer bilgilere müdahale etmeden yeni bilgiler öğrenir. Yapay zekâ programları da benzer bir şekilde çalışır.

Yapay zekânın amacı, basitçe sadece karar vermeyi desteklemeyi amaçlamaz. Ana hedefi, kendi kararlarını verebilecek makineler oluşturmaktır. Bu makineler, karar vermek için zekâlarını insan benzeri bir şekilde kullanmalıdırlar. Deneyimden öğrenebilmeli, bilgisinin sınırlarını bilmeli, gerçek bir yaratıcılık ortaya koyabilmelidir.

5.1. Yapay Zekânın Tanımı ve Tarihçesi

Yapay zekânın çeşitli kaynaklardaki tanımlarından birkaç örnek vermek gerekirse;

"Yapay zekâ, insan düşüncesine benzer bir yapıda belirli bir alanda problem çözmeye yaklaşan, doğal dili anlama, öğrenme, sonuç çıkarma ve problem çözme yeteneğine sahip disiplinler arası bir bilimdir." (Akhun, 1992)

"Yapay zekâ, zeki davranışın araştırılmasıdır." (Genesereth, 1987)

Bu tanımlarda geçen "zeki davranış" durumu sadece problem çözme, satranç oynama ve matematiksel zorunlukları gibi yüksek oranda zekâ gerektiren hareketler ile sınırlı olmayıp; cisimleri tanıma ve basit bazı yazıları anlanlandırabilme gibi ortalama bir zekânın gerçekleştirebileceği işlemleride kapsamaktadır. Hatta cisimlerin ve seslerin algılanması

gibi sorunları çözmeye ile direkt alakalı olmayan bazı işlemlerin de yapay zekânın ilgi alanına dahil olduğu söylenebilir (Garnham, 1988).

Çizelge 5.1.'de yapay zekâ ile geleneksel programların genel kabuller üzerinden bir karşılaştırması değerlendirilmiştir. (Özdemir ve diğerleri, 1992).

Yapay Zekâ	Geleneksel Bilgisayar Programlama
Gözlemlenemez çözüm adımları	Gözlemlenebilir çözüm adımları
Öncelikle sembolik işleme	Öncelikle sayısal işleme
Genelde alan bilgisinden ayrı kontrol yapısı	Bilgi ve kontrol iç içe
Düzeltilme, güncelleştirme ve genişletme kolay	Düzeltilmeler oldukça zor
Bazı doğru olmayan çözümler ile sıkça karşılaşılabilir.	Doğru çözümlere ulaşmak gereklidir
Kabul edilebilir, tatminkâr çözümler kabul edilir.	Genellikle en iyi çözüm kabul edilir

Çizelge 5.1. Yapay zekâ uzman sistem karşılaştırması

Yapay zekâ programlarında bazı yapısal özellikler ve teknikler sıklıkla kullanılmaktadır. Bu teknikler, tek başlarına oldukça basit olsalarda bir arada kullanıldıklarında birbirlerini pekiştirmekte ve uyumlu bir birliktelik oluşturabilmektedir. Bu unsurlar şu şekilde sıralanabilir:

- Veri tabanları (Data Structures)
- Arama (Search)
- Hüristik (Heuristic)
- Otomatik çıkarm (Automatic Reasoning)
- Doğal dil (Natural Language)

John McCarty, 1956'da Amerika Birleşik Devletleri'nde düzenlenen "Makine Zekâsı (Machine Intelligence)" kongresinde, yapay zekâ terimini ortaya atmıştır. O zamandan beri terim, dünya çapında kabul edildi ve dilimize yerleşti.

Öte yandan yapay zekâ araştırmaları ise, program yüklenebilen bilgisayarların ortaya çıkmasıyla başladı. İlerleyen tarihlerde, bilgisayarların da öğrenebileceği ve düşünebileceği

fikri ortaya çıkmıştır. Bu gelişmelerle birlikte 1958 yılına gelindiğinde, yapay zekânın yine John McCarty tarafından geliştirilen LISP (İlk Yapay Zekâ Programlama Dili) ile başladığı kabul edilmiştir.

1964'da hayata geçen ELIZA adlı program Joseph Weizenbaum tarafından geliştirilmiş ve psikanalizi çok iyi uygulamıştır. İlerleyen yıllarda ise genel hedef olarak zeki programlara hedeflenmiştir. Bu durum, daha fazla kullanılabilen bilgi işlem kaynaklarını gerektirmiştir. Çünkü hafızayı doldurma ya da çok yavaş çalışma gibi bazı programdan kaynaklı sorunlar ortaya çıkmıştır.

1970'li tarihlere gelindiğinde, hız ve hafıza yönünden daha yüksek kaliteli bilgisayarlar üretilmiş fakat yapay zekâ çalışmaları için bu bilgisayarlar da yetersiz kalmıştır. Bu nedenlerle ve verimsizlikle birlikte yapay zekâ çalışmalarının etkinlikleri yitirilmiştir.

1972'de Marseille Üniversitesi'nde LIPS'e benzeyen bir PROLOG dili geliştirilmiştir. Yeni geliştirilen bu PROLOG dili, basit bir yazılım meydana getirilmiştir. Bu sayede, 1980'lerde PROLOG Avrupa'da; LISP ise Amerika'da popülerdir.

1970'li yılların sonuna doğru doğal dil işleme, sorunların çözümü ve bilgi anlatımları gibi bazı kazançlar sağlanmıştır. Yapılan bütün çalışmaların sonucunda ilk ticari yapay zekâ programı olan uzman sistemleri meydana gelmiştir.

Yapay zekâyâ büyük bir ilginin başlaması 1980'li yıllara dayanmaktadır. Bu tarihlere gelindiğinde, bilgisayar bilimindeki ve mühendislikteki çözümlenemeyen tüm problemlerin yapay zekâ ve uzman sistemler tarafından çözülebileceği kanısı yaygınlaşmıştır. Birçok iyi yetişmiş bilgisayar bilimcisi, mühendis ve felsefeci yapay zekâ üzerine ciddi çalışmalarda bulunmuştur.

1981 yılında ise Japonlar, beşinci nesil bilgisayarlarında PROLOG dilini temel olarak kullanmaya hazır olduklarını ve bu kapsamda çalışacağını açıklamışlardır. O zamandan günümüze PROLOG, yavaş yavaş üniversite ve araştırma kuruluşlarındaki sistem bilgisayarlarından günlük hayatta kullanılan mikrobilgisayarlara yer bulmaya başladı. Mikro-PROLOG ve Turbo-PROLOG programları bunun başlamasının kanıtı olarak düşünülebilir (Schildt, 1987).

1980'li yılların sonuna doğru ise yapay zekâ ve uzman sistemler konusunda beklenen gelişmeler yaşanmamıştır. Başlangıçtaki istek, ardında çözümlenemeyen problemlere yönelik birçok sorular bırakarak azalmıştır. Yapay zekânın, varolabilmesi konusunda ciddi şüpheler oluşmuştur. Diğer taraftan, bulanık mantık ve nesne tabanlı programlama gibi bazı başarılı dalların, yapay zekâdan esinlenerek ortaya çıktıklarını tespit edilmiştir. (Erginbilgiç, 1995).

1980'lerin sonunda bilgisayar bilimcileri bu bilim dalının gerçek değerini anlamaya başlamışlardır. Yapay zekânın sadece, bilgisayar üzerindeki programlarındaki yeni bir veri seti olduğu ve düşünüldüğü gibi, tüm çözümlenmesi imkânsız problemler için bir alternatif olmaktan oldukça uzak kaldığı görüşü yayılmaya başlamıştır. Bu teknolojinin, kural tabanlı uygulamalar ve nesne tabanlı programlama gibi artılarının kabul edilmesi ve sadece ihtiyaç olması durumunda eldeki problemin durumuna göre kullanılması gerekmektedir. (Erginbilgiç, 1995).

5.2. Yapay Zekânın İlgilendiği ve Kullanıldığı Alanlar

Bir problemi çözerek ortadan kaldırmak için bilgisayar desteği alınmasına karar verildiğinde, problem matematiksel fonksiyonlarla tanımlanamıyorsa ya da çalışmanın sınırları belirlenemiyorsa, yapay zekânın kullanılması gerekiyor demektir. Bu durumda bilgisayarı, uzmanın probleme yaklaşımını benimseyecek ve bu şekilde tepkiler verebilecek şekilde programlamamız gerekmektedir. Eğer kişilerin bilgileri hangi sırayla işlediğini bilmek tam olarak mümkün değilse, bu düzeni yaklaşık olarak tahmin etme yoluna gidilmiştir.

Yapay zekânın ilgi alanları şu şekilde sıralanabilir:

- Bilgisayarın görmesi (Computer Vision) Robot çalışmalarında kaydedilen gelişmelerin sonucu olarak ilgi, bu makinelerin etrafındaki cisimleri fark etme ve tanıma yetenekleri üzerine geliştirilmiştir.
- Konuşma sesinin meydana getirilerek üretilmesi, (Speech Production)
- Ses algılama (Voice Recognition) Makine harici sesleri (insan sesi ve diğer herhangi bir makinenin sesi) algılayabilmesi

- İnsan dilinden algılayabilme ve çeviri yapabilme (Language Understanding and Translation)
- Düşünme, sonuç çıkarma ve problem çözme (Thinking, Reasoning and Problem Solving) Bu konu yapay zekânın en geniş kapsamlı çalışma alanlarından biridir. Bu alandaki bazı başlıkları şu şekilde örnekleyebiliriz: Matematik teorem ispatlama, zekâ oyunları, öğrenen makineler, uzman sistemler.

Bu alanlara ek olarak bazı yapısal elemanlar da araştırma konusu olmuştur. Bunlar içerisinde arama (Searching), doğal dili işleme (Natural Language Processing), desen karşılaştırma ve algılama (Pattern Matching and Recognition), mantık (Logic). Belirsizlik ve bulanık mantık (Uncertainty and Fuzzy Logic) örnek gösterilebilir (Agrawala,1977).

Yapay zekânın bazı uygulama alanları aşağıda kısaca sunulmuştur;

Arama

Arama ifadesi yapay zekâ çerçevesinde herhangi bir soruna karşı üretilecek çözümleri tespit etme olarak kullanılabilir. Burada anlatılmak istenen durum veri yığını içerisinde anlamlı bir parçacığın bulunması şeklinde değildir Şehir seçenekleri arasındaki en kısa yolun ya da matematiksel bir ifadenin ispatlanması gibi soruların çözümü, bu çalışma alanında yapılan çalışmalara örnek teşkil edebilir. Bu kapsamda yapılan çalışmalar sonucunda birçok arama yöntemi keşfedilmiştir (Schildt, 1987).

Doğal dili işleme

Yapay zekânın en çok merak edilen alanlarından biridir. Birçok çalışmanın bu alanda yapılmasının ana nedeni bu alanda yapılan araştırma sonuçlarının, inşaların bilgisayar ile yapabildiği iletişimi bilgisayarlara da kazandırabilmektir. Buradaki en önemli problem insanların çok karmaşık ve yüksek boyutlardaki dil yeteneğidir (Schildt, 1987).

Desen gösterimleri

Robotlat ve görsel süreçlerin analizleri ile ilgilenen uygulamalar için Desen işleme ve şekillerle çalışma çok önemlidir. Örneğin, “Bir televizyon ekranı üzerindeki nesnelerin

başlangıç ve bitiş yerlerini veya hangi nesnelerin hangi nesneler üzerine konulabileceğini bilgisayara nasıl tanıttırabiliriz?" sorusu düşünüldüğünde benzer çalışmalar ihtiyacın olduğu görülür. Gölgelemlerin nesnelerden ayırdedilebilmesi, bu çalışmaların odak noktasıdır (Turban, 1990).

Robotlar

Robot denemeleri, yapay zekâda elde edilen hareketlerin kontrol edilmesi amacıyla yapılmıştır. Burada kontrol mekanizmaları geliştirilerek, robotların insans bir şekilde hareket edebilmesi sağlanması amaçlanmıştır (Turban, 1990).

Öğrenme

Yapay zekânın ilgi çeken bir çalışma alanında bilgisayarların öğrenme özelliklerinin geliştirilebilmesidir. Bu kapsamda gerçekleşen çalışmalarda, hatalardan ders çıkarabilen, yanlışlarını öğrenebilen ve hataları tekrar etmeyen bilgisayar yazılımları oluşturmak amaçlanmıştır. Öğrenmenin diğer bir anlamı da tecrübelerden yararlanabilir bilgisayarlar geliştirmektir (Schildt, 1987).

Mantık

Yapay zekânın diğer bir çalışma alanı ise mantığın kurallarını kullanarak bir probleme karşı üretilen çözümün mantıksal olarak doğru olduğunun anlaşılabilirdiği uygulamaları varetmektir. (Turban, 1990).

Belirsiz ve bulanık kümeler

İnsanlar bazen kararlarını elindeki verilere güvenmese belirlemek zorundadır. Bu şekilde durumlar için çok farklı sonuçlar doğurabilecek kararlar alınabilir. Bilgisayarlara bu durumu öğretebilme konusu bu alan için düşünülebilir (Turban, 1990).

Uzman sistemler

"UZMAN SİSTEMLER" olarak isimlendirilen uygulamalar, aşağıda detaylı olarak değinilecektir.

5.3. Guguk Kuşu Arama Algoritması

2009 yılında Xin-She Yang ve Suash Deb tarafından guguk kuşunun yaşam döngüsünden ilham alınarak geliştirilen Guguk kuşu algoritması üst sezgisel (metaheuristic) bir algoritmadır [54]. Bu algoritma bazı guguk kuşu türlerinin kuluçka asalaklığı ve Lévy uçuşu (Lévy flights) adı verilen rastgele yürüyüşlerini temel almaktadır.

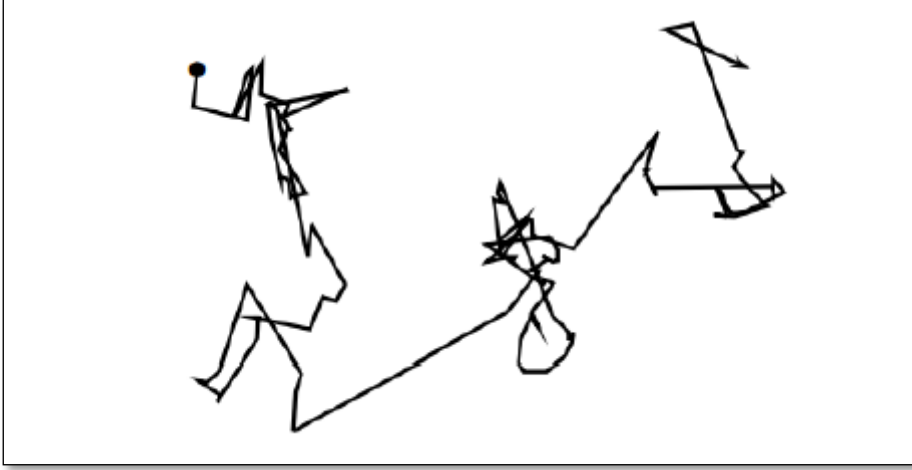
5.3.1. Guguk kuşu üreme davranışları

Guguk kuşları doğal yaşamda yabani çoğalma yöntemleri ile dikkat çeken kuş türüdür. Türlerin büyük bir kısmında kuluçka problemi olup, yumurtalarını başka kuşların yuvalarına terk ederler. Konak yuvanın sahibi dişi kuşun yumurtlamaya başlamasını bekleyen Dişi guguk kuşu, çoğunlukla bir yumurtasını kuluçkaya yatmadan önce başka yuvalara bırakır. [55]. “Ani” ve “Guira” gibi bazı guguk kuşu türleri kendi yumurtasını bırakacağı yuvanın sahibi diğer kuşa ait yumurtalardan birini alarak bıraktığı yumurtanın yaşama ihtimalini yükseltmeye çalışır. Yuvasına yumurta bırakılan kuş eğer yuvasında ona ait olmayan yumurtayı fark edebilirse, ya yuvasını terk eder yada bulduğu bu yumurtayı yuvasından dışarıya çıkartır [55]. Yeni Dünya türlerinden kuluçka asalağı “Taparak” gibi bazı guguk kuşu türleri, seçtikleri farklı bir kuş çeşidine şekil ve renk bakımından benzemeye çalışarak yumurtalarını yuva sahibi olan kuş türünün yumurtalarına benzetmek için uğraşırlar [55, 56]. Böylece kendisine ait yumurtanın farklı kuş türüne ait yuvada yaşama şansını yükseltirler.

Konak kuş tarafından Guguk kuşu yumurtası anlaşılmazsa kuluçka sırasında konak kuşun kendine ait yumurtalarından önce dünyaya gelirler. Yumurtadan çıkan guguk kuşu yavrusunun gözleri henüz görmemektedir ve konak kuşun bulacağı besinleri yuvanın asıl sahibi olan yavruya vermemek için sırtını ve kanatlarını kullanarak kendi için görece zor olan hareketlerle mevcut yumurtaları birer birer yuvadan çıkarır [56].

5.3.2. Lévy uçuşu

Lévy uçuşu, hayvanların besin bulmak amacıyla yaptıkları yöntemlerden birisidir [57]. Bu yöntemle göre, yiyecek araştıran bir hayvan ilk olarak küçük bir alanı bir maddedeki atomların yaptığı “Brown hareketleri” ne benzer olarak uyguladıktan sonra rastgele tayin edilen bir yön belirleyerek geniş bir alanda yiyecek aramaya başlarlar. Bu hareketler neticesinde Şekil 5.1’deki gibi bir desen ortaya çıkmaktadır.



Şekil 5.1. Lévy uçuşu [60]

Lévy uçuşu adım değerleri sürekli bir olasılık dağılımı olan Lévy dağılımına uygun şekilde yapılan gelişigüzel bir yürüyüş olarak tanımlanmıştır. [58]. Sonsuz ortalama ve varyansa sahip Lévy dağılımı Eş.5.1.'teki gibi tanımlanabilir [54].

$$\text{Levy} \sim u = t^\lambda, (1 < \lambda \leq 3) \quad (5.1)$$

Adım değerini tespit etmek için birkaç yöntem vardır. Bunlardan en verimli ve kolay yöntemlerden birisi Mantegna algoritmasını uygulamaktır. Adım değeri pozitif veya negatif şekilde olabilir. Mantegna yönteminde adım büyüklüğü (s), eşitlik Eş.5.2.'daki hesaplanır [59].

$$S = \frac{u}{|v|^{1/\beta}} \quad (5.2)$$

Bu eşitlikteki β parametresi $[1, 2]$ aralığında olup genellikle 1.5 olarak alınırken u ve v , ortalaması sıfır olan normal dağılımlı rassal değişkenlerdir ve Eş.5.3.'deki gibi tanımlanırlar.

$$u \sim N(0, \sigma_u^2), \quad v \sim N(0, \sigma_v^2) \quad (5.3)$$

u ve v 'nin σu ve σv standart sapmaları Eş.5.4.'de verilmiştir.

$$\sigma u = \left\{ \frac{\Gamma(1+\beta) \sin(\frac{\pi\beta}{2})}{\Gamma[(\frac{1+\beta}{2})] \beta 2^{(\beta-1)/2}} \right\}^{1/\beta} \quad \sigma v = 1 \quad (5.4)$$

5.3.3. Guguk kuşu algoritması

Guguk kuşu kendisine ait olmayan bir yuvaya yalnızca bir adet yumurta bırakır ve yuvada bulunan her yumurta bir çözümü, Guguk Kuşuna ait olan yumurta yeni bir çözümü ifade eder. Yeni ve potansiyel olarak daha iyi çözümler olan (guguk kuşu yumurtalarını) Yuvadaki iyi olmayan çözümlerle değişikliğini sağlamaktır buradaki temel amaçtır. Algoritmanın doğru sonuçlar üretebilmesi için diğer yöntemlere göre daha kolay olan tek yumurta taşıması yaklaşımı ile çözüm üretilmiştir [54]. Dolayısıyla bir guguk kuşu yumurtası veya yabancı kuş yuvası bir çözümü ifade eder. Algoritmada nesil konak kuş yuvalarından meydana gelmektedir. Guguk kuşu algoritmasının temeli aşağıdaki üç prensibe dayanmaktadır [54].

1. Guguk kuşları her uçuşunda sadece bir yumurtayı rastgele belirlenen yabancı kuşun yuvasına koyar.
2. Hayatta kalabilen yuvalar yeni nesillere aktarılır. Yuvaların kalitesi probleme göre belirlenen amaç fonksiyonu ile tespit edilir.
3. Yuva adedi değişkenlik göstermez ve konak kuş tarafından fark edilen guguk kuşu yumurtası $pa \in [0,1]$ olasılığı ile gerçekleşeceği düşünülür. Fark edilecek yuvaların yerine rastgele yeni yuvalar tayin edilir.

Lévy uçuşu ile yeni rotalar $x_{(t+1)}$ üretmek için Eş.5.5. kullanılır. Bu eşitlik aslında, gelecek durumu sadece şu andaki mevcut durumuna ($x(t)$) ve geçiş olasılığına bağlı olan Markov zinciri ile tanımlanan rastgele bir yürüyüşü ifade eder.

$$x_{i(t+1)} = x_{i(t)} + \alpha \oplus Levy(\lambda) \quad (5.5)$$

Eş.5.5. eşitliğindeki $\alpha > 0$ olmak üzere adım büyüklüğünü gözden geçirmek için kullanılan bir parametredir. Çoğu durumda $\alpha = 1$ olarak uygulanabilir.

Guguk kuşu algoritmasına benzer bir yürüyüşe sahip olan PSO'ya rağmen, Lévy uçuşu sayesinde uzun adımlar gerçekleştirilebildiği için arama uzayının tespiti konusunda daha etkindir [54].

Yeni çözümlerin bazıları Lévy uçuşu ile en sağlıklı çözüme yakın yerlerde üretilmeye çalışılır. Bu yerel aramayı hızlandıran bu durumla birlikte en iyi çözümden uzak yerlerde

yeni çözümler üretilmesi sağlanır. Dolayısıyla yerel minimuma takılma durumunun önüne geçilir.



6. UZMAN SİSTEMLER

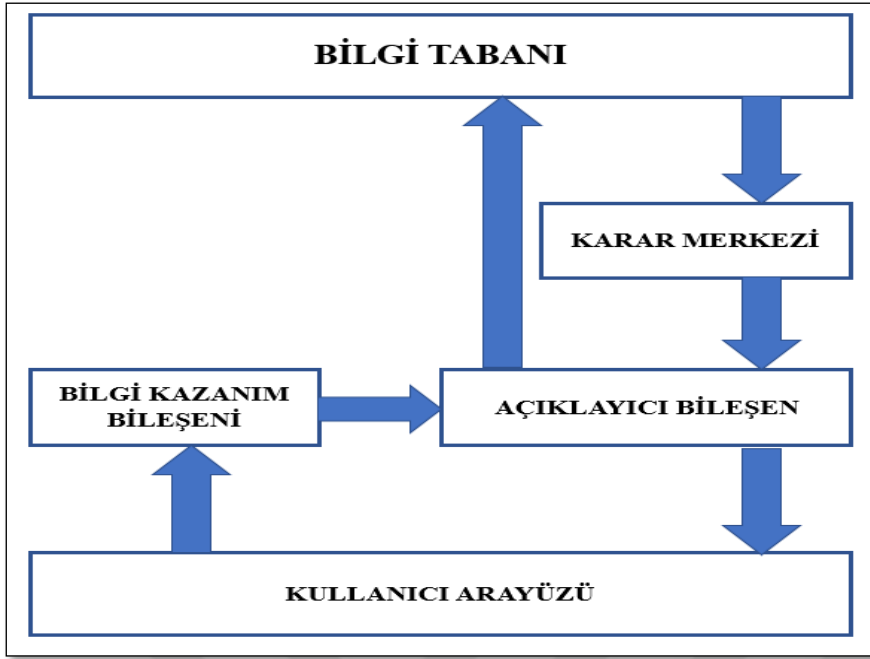
Nabiyev, uzman sistemi (US), “*Bir konuda uzman kişi ya da kişilerce yapılabilen, yargılama ve karar verme işlerini modelleyebilen bilgisayar sistemleridir*” ile açıklamıştır [60]. Edward Feigenbaum ise “*bilgi ve çıkarım prosedürlerini kullanarak uzman bilgisi gerektiren zor problemleri çözen akıllı bilgisayar programları*” tanımı ile açıklamıştır [61].

6.1. Uzman Sistem

Uzman Sistemler bilim adamları tarafından 1970'lerde ortaya çıkmış ve ticari faaliyetlerde 1980 yıllarında ilk olarak uygulanmaya çalışılmıştır. Bir US bir konuda uzmanlaşmış insanların bilgilerinin depolandığı ve bilgisayar ortamında benzer bir algoritmanın oluşturulmasıdır. Bir US'un bir konuda çözüm için ayrılmış tasarlama alanına “domain” denir. Örneğin ilk üretilen US'lardan biri olan DENDRAL'in alanı moleküler üzerinde geliştirilmiş olup bileşik yapısını çözmek için kullanılmıştır.

Uzman Sistemlerde temel olarak konu ile ilgili bilgiler kullanıcı tarafından verilir ve uygulamadan çözüm ve destek beklenir. US'un iki ana noktası bulunur. İlki bilgi tabanıdır (Knowledge-Base). Bilgi tabanı uzmana ait bilgileri ve kuralları içerir. İkinci kısım ise karar tabanıdır (Inference Base). Karar tabanı ise veri havuzundan ve kullanıcıdan aldığı bilgileri birleştirerek problem çözümü için olasılıkların üzerindeki karar merkezidir. [62].

Yapay zekâ seviyesi yüksek sistemlerin hepsinde karşılaşıldığı üzere US'larda bir soruna karşı insanların düşünme, tasarlama ve uygulama faaliyetlerini örnek alır. Uzman tarafından gerçekleştirilen işlerin bir bölümü ya da hepsinin bilgisayar ortamında yaptırabilmek amacıyla kullanılır. Şekil 6.1'de bir uzman sisteme ait çalışma şeması paylaşılmıştır [62].



Şekil 6.1. Uzman Sistem Genel Yapısı

6.2. Uzman Sistemlerin Avantajları

Uzman Sistemlerin kullanılacağı alana göre önemli avantajı aşağıdaki şekilde örneklendirilmiş ve kısaca açıklanmıştır [63-64]:

Üretim

US'lar insan uzmanlardan beklenenden daha kısa bir sürede daha fazla ürün elde edilmesini amaçlamaktadır.

Süreklilik

Makinelerin unutma gibi sorunları olmamasından ötürü her zaman aynı kararlılıkla çalışabilirler.

Maliyet Tasarrufu

Bir insanın zaman içinde eğitilip, uzmanlaştırılarak çalıştırılması yüksek maliyetlere neden olmakla beraber bir defaya mahsus bir US satın alınması sürekli maaş alacak bir uzmana göre daha ekonomik olacaktır.

Kararlılık

İnsan uzmanların kararları günlük olarak değişebilir. Yorgunluk, hastalık psikolojik nedenler vb. sebepler uzmanın kararını etkileyebilmektedir. Uzman Sistemlerin insana özel belirli sorunları olmadığı için daha kısa sürede, daha doğru, düşük miktarda hata payıyla ve her zaman aynı yönde hareket ederek daha doğru kararlar üretebilmektedir.

Çalışılmayan Sürenin Azaltılması

Uzman Sistemler bakım, arıza tespit ve onarımında kullanıldığında daha kısa zamanda işe tekrar kazandırılabilir, arızası engellenebilir.

Yetkinliğin Yaygınlaştırılması

Bir uzmanın her yere gönderilmesi ya da anlık olarak orada bulunabilmesi mümkün olmayabilir, zira ihtiyaç halinde o alanda yetkin uzman tespit edilemeyebilir; oysaki Uzman Sistemler birkaç dakikada çoğaltılabilir.

Eğitim Verme

Kullanıcıların sorun yaşadığı problemlerde verileri analiz edilip, çözümler sunan veya problemleri ön görüp uyarı vererek kullanıcıların eğitilmesini sağlar.

Sağlıklı Öneri Üretimi

Bir insan uzmanın aksine dikkat dağıtabilecek dış etkenler devre dışı bırakıldığında US'lar genellikle daha verimli sonuçlar üretirler.

Güvenirlilik

İnsan uzmanın kaygı ve duygularının olmadığı US'lar insan uzmandan farklı olarak tüm yetkinliğini en doğru sonuçları gerçekleştirmek için kullanabilirler.

Kesin Olmayan veya Eksik Bilgi ile Çalışabilme

Diğer bilgisayar uygulamalarından farklı olarak eksik bilgiler de olsa sonuç üretebilirler.

Veri Analiz Yeteneği

Çok fazla veri yoğunluğu söz konusu olduğunda bile olabilecek tüm durumları analiz edip, olasılıklara göre farklı sonuçlar üretebilme yetkinlikleri mevcuttur.

Verimlilik

Uzman sistemiyle, olası hata ve hilelerin önüne geçilebilir. Bilgilerin farklı seviyelerden süzülerek geçmesi zorunluluğu kalmaz. Bilgi karar için yeterlidir ve zaman kaybına tahammül etmek etmez.

Bu özellikler dışında başka kaynaklardan da ilham alınarak özellikler çoğaltılabilir. Fakat US'ların olumlu yanlarının yanı sıra çeşitli sınırlamaları da vardır.

6.3. Uzman Sistemlerin Dezavantajları

Uzman sistemlerin işletmelerde yayılması, kullanım ve uzmanlık alanlarının genişletilmesiyle ilgili bazı sorunlar şunlardır [63-64]:

1. Bazı konular uzmanlık bilgisi gerektirmez; çünkü bazı insanlar edindikleri bilgilerini bilgi mühendisine aktarmak için zamanları olmayabilir veya uzmanlıklarını başkalarına aktarmak istemeyebilir. Bazen teknolojiye olan güvensizlik ya da işini kaybetme korkusu da buna sebep olabilir.
2. Uzmanların aynı konular üzerinde farklı görüşleri olabilir, eğer tek bir uzman görüşüyle bir bilgi tabanı oluşturulursa bu sadece onun fikirlerini temsil edecektir. Fakat başka bir uzmanın bilgisi ile aynı probleme daha ekonomik ve kullanışlı çözümler getirilebilir.
3. Dar alanlara ve belirli problemlere özgü olan uzman sistemlerinde, alan farklılaştıkça sistemin çalışması verimsiz hale gelebilir.

4. Bir yönetici veya kullanıcı, konuyla ilgili uzmanların varlığı söz konusuysa bir bilgisayar programını kabullenmeyi, para ödemeyi, iş için kullanmayı ve güvenmeyi reddedebilir.

5. Uzmanlar, genellikle kendilerine has yöntem ve kavramlar kullanır. Herkesçe kabul görmüş bazı kavramlar bazı uzmanlar için değişik ifadelerle neden olabilir. Bu durum ise bilgi mühendisinin işini bir hayli zorlaştırmaktadır.

6. Uzman sisteminin iyi bir şekilde oluşturulabilmesi ekonomik açıdan maliyetli ve zaman açısından da uzun süreler gerektirecek bir iştir.

7. İnsanlar yeni konulara daha rahat adapte olarak kolay öğrenebilir ve daha önce karşılaşılmamış sorunlara karşı basit çözümler ortaya çıkarabilir. Bir US için hiçbir şey ifade etmeyen durumlar “Kafasına elma düşmesi veya hamamdaki tasın yüzdüğünü görmesi” insan için yeni bir fikrin habercisi olabilmektedir.

6.4. Uzman Sistemlerin Tarihçesi

1970’lerde ilk US’lar üretilmeye başlanmıştır. DENDRAL, ilk Uzman Sistemler olarak kabul edilir. DENDRAL: “Kimyasal Analiz Uzmanı Sistemidir ve yapısı bilinmeyen kimyasal bileşiklerin moleküler yapısını bulmak için geliştirilmiştir.” Carnegie-Mellon Üniversitesi’nde konuşmayı metne dönüştürmek ve bir yazıcıya göndermek üzere HEARSAY I (1969) ve HEARSAY II (1971) US’ları tasarlanmıştır. Digital Equipment Corp (DEC) tarafından DEC firmasının bilgisayar sistemlerinin konfigürasyonunda kullanılmak üzere XCON isimli US tasarlanmıştır. 1970’lerde Standford Üniversitesi’nde kan enfeksiyonlarının teşhisi için MYCIN geliştirilmiş; fakat hiçbir zaman kullanıma sokulmamıştır. Buna rağmen MYCIN, en çok bilinen uzman sistemdir.

Günümüzde ise pek çok alanda kullanılan gelişmiş uzman sistemler bulunmaktadır. Örneğin; GATES; “havaalanı pist tayini ve izleme için”, HESS; “petrokimya sanayisi ve uzman sistem planlama için”, DustPro; “maden ocağı güvenliği için”, TOP SECRET; “güvenlik sınıflandırması için”, Codecheck bilgisayar programı değerlendirme için geliştirilmiş uzman sistemler olarak sayılabilir.

6.5. Uzman Sistem Uygulama Alanları

Uzman sistemler, insan uzmanlığının gerekli olduğu birçok alanda kullanılabilmektedir. Genel olarak anlamlandırma, tespit etme, eğitim, tahmin, planlama, tasarım, görüntüleme, hata ayıklama ve kontrol amaçlı uygulamalar mevcuttur [65-68].

6.5.1. Kimyada uzman sistemler

Genellikle iki grup olarak ele alınan kimya alanındaki US'lar: organik bileşenlerin analizini yapan US'lar, bileşiklerin sentezinde yer alacak reaksiyon zincirini tahmin eden US'lardır. Kimya alanındaki en çok bilinen uzman sistemlerden aşağıda bahsedilmiştir [69-70].

DENDRAL 1968 yılında organik bileşiklerin moleküler yapısını belirlemesi amacıyla tasarlanmıştır.

CONGEN 1976 yılında bileşikler molekül yapılarını analiz etmek amacıyla tasarlanmıştır.

CRYSTALIS Proteinlerin şekli ve iç yapısının incelenmesini otomatikleştirmek amacıyla tasarlanmıştır.

LHASA Çok karmaşık bilgilerin dönüşümleri için Harvard Üniversitesi tarafından tasarlanmış ve kullanılmıştır.

FALCON Kimyasal üretim sırasında oluşan hataları tespit etmek için kullanılmıştır.

CONPHYDE Kimyasal bir karışım için gerekli şartları belirleyebilmek amaçlı tasarlanmış ve kullanılmıştır.

PICON Ortaya çıkabilecek tehlikeleri önceden görebilen uzman bir danışman olması amacıyla petrol rafinerileri kontrolü için geliştirilmiştir.

HEATEX Kimyasal reaksiyonlardan sonra açığa çıkacak enerjiyi kontrol etmek amacıyla tasarlanmış uzman bir sistemdir.

6.5.2. Tarımda uzman sistemler

Tarım arazilerinin azalması, nüfusun gün geçtikçe artması ve dünyanın küresel ısınmanın etkisinde olması sonucuyla özel tarım yöntemlerine gereksinim artmaktadır. Sürekli olarak tarım uzmanlarından bilgi almak çiftçiler için mümkün olmadığından tarım alanında da birçok uzman sistem geliştirilmiştir [71].

Cuptex, salatalık üretimi, salatalık hastalıklarının teşhisi ve tedavisi için üretilmiştir. Ayrıca Cuptex birçok konuda da (sulama, gübreleme, bakım vb.) uzman bilgisi sağlar. Aynı firma tarafından Citex isimli, portakal ve Tomatex isimli, domates için geliştirilmiş uzman sistemler vardır. Yine aynı firma tarafından geliştirilen Neper Wheat isimli uzman sistem; buğday üretimi, hastalık, haşere tanısı, sulama, gübreleme ve hasat planı için fikirler vermektedir.

Planting, tohum üreticileri ve bitki yetiştiricileri için bir tarım bölgesinde toprak ve iklim bilgilerini girdikten sonra rehberlik yapabilmektedir. Bu alandaki bir diğer US ise uçakla ilaçlama yaparken uçağın hızı, bitkilerin özellikleri, ilaç türü, yükseklik ve rüzgâr gibi bilgiler verildikten sonra ilaçlamanın yapılacağı zaman aralıklarını hesaplamaktadır.

SEPA Çiftliklerde yazın ve kışın ekilecek bitki ve hayvan yemleri için öneri sunan ve bu süreci planlayabilen bir uzman sistemdir.

MAIZE Tasarlanma amacı tarım işçilerine yardımcı olmaktır. Zararlı böcekler, hastalıklar ve benzeri yabancı otlardan nasıl korunulacağına dair tavsiyelerde bulunur. Toprağın gübrenmesi, gübrelerin oranlarının ve karışıklıklarının hesaplamasında yardımcı olabilen bir US'dur.

6.5.3. Eğitimde uzman sistemler

WHY Öğrencilerin coğrafya derslerinde nasıl yağmur yağdığını anlamalarına yardımcı olmak amacıyla geliştirilmiş bir uzman sistemdir.

WEST Oyunlar aracılığıyla öğretime yardımcı olmak amacıyla geliştirilmiş bir uzman sistemdir. Öğrenci, öğretmeni karşısındaymış gibi hisseder.

BUGGY 1970 yılında geliştirilmiştir. Matematik alanında kullanılmaktadır. Öğretmenlere, öğrencilerinin neden başarısız olduklarını söylemesi amacıyla geliştirilmiştir.

Bu uygulamaların yanı sıra öğretim amaçlı SCHOLAR; matematik eğitiminde kullanılan Wumpus ve Excheck, mühendislik eğitiminde kullanılan Ariadna ve Steamer gibi daha birçok uzman sistem profesyonel olarak uygulanmaktadır [72].

6.5.4. Endüstride uzman sistemler

Endüstrinin hemen her alanındaki çözümlere yönelik özel bir uzman sistem bulunabilmektedir. Tasarlama, planlama, karar verme, üretim, dağıtım ve yönetim gibi işlemler için birçok uzman sistem geliştirilmiş ve geliştirilmeye devam edilmek.

Tasarım alanında: “ALADIN, ALEXSYS, CORACALLISTO, CONSTRAINTS, XSEL, XCALIBUR, XCON/R1, MAPCON, EURISCO” gibi US’lar kullanılmaktadır.

Planlama alanında: “Rome, Vicicals, ESP, IMS, EXAPT, FADES, RACE, DEVIZER, IFLAPS, GARI, CAPP, TIPPS, TOM, SIPP, SIPS, KAPPD, TURBO-CAPP, XPLAN, XPLANE, GIPPS, SAPT, CUTTECH, Hi-Mapp, IRPPSS, EXPERT-AG” gibi uzman sistemler kullanılmaktadır.

PILDTEX, SIMMIAS, ROTES, ISI, ESTA; endüstrinin üretim aşaması için üretilmiş uzman sistemlerdir.

Bunların haricinde; dağıtım, servis ve yönetimde kullanılmak için üretilmiş “INET, ASE, CATS, DAPT, IDT, PDS, XSITE, CALLISTO, AZEREKS, ESDARR, ESLS” programların varlığı bilinmektedir [71, 73].

6.5.5. Tıpta uzman sistemler

Tıbbi uzman sistemler (TUS), tıbbi alanlar içerisinde standart soruları ve yanıtları verebilmek amacıyla geliştirilmiş US’ lar olarak ifade edilebilir. TUS bir veya daha çok tıbbi uzmanın tavsiyeleri doğrultusunda geliştirilmiştir. Böylece, doğru sonuçların üretilmesi amacıyla en uygun sorular dikkate alınmıştır. TUS’ in amacı hekime tavsiye ve önerilerde

bulunmaktadır. Amaç asla hekimin yerine geçmek olmamıştır. TUS uygulamaları arasında, tıp eğitiminde kullanılan farklı uygulamalar da yer almaktadır.

Tıbbın her faaliyetinde kullanılmak üzere ortaya çıkmış bir uzman sistem bulunmaktadır. Bu US'ların en çok bilinenleri hakkında detaylı bilgiler aşağıda verilmiştir [61, 72].

MYCIN (1970) Enfeksiyon hastalıklarının teşhisi ve tedavisini amaçlayan Uzman Sistemdir. Herhangi bir ameliyat sonucu hastaya bulaşan enfeksiyonu kısa zamanda teşhis ve tedavi etmek amacıyla kullanılmaktadır.

CASNET Göz tansiyonu hastalığının teşhisi amacıyla geliştirilmiştir. Bu hastalığa yaşamın herhangi bir safhasında yakalanılacakmış gibi yaklaşılır. Diğer göz hastalıkları için de kullanılabilir.

INTERNIST 1970 yılında, iç hastalıklarının teşhisinde kullanılmak amacıyla geliştirilmiştir. Yazılım, hasta ile ilgili doktor görüşü, tahlil sonuçları ve tıbbi geçmiş girilerek olası hastalıkları teşhisi etmek için kullanılmıştır. Daha sonradan geliştirilerek "INTERNIST II" adını almıştır. INTERNIST II, dahiliye bölümünün %75'e yakın hastalıkların teşhisi için gerekli uzman bilgisine sahiptir.

PIP Böbrek hastalarını teşhis etmek amacıyla geliştirilmiştir.

Digitalis Therapy Advisor Diyaliz hastalığının teşhisi ve diyaliz sürecinde doktorların fikir alabildiği bir uzman sistemdir.

LEDI-2 Yoğun bakım hastasının tedavi edilişi ve hastalığının gidişatını kontrol etmek amacıyla kullanılır.

MDES Ülser hastalığının teşhis ve tedavisi amacıyla üretilmiştir. Uygulamanın elde ettiği yüksek başarı klinik deneylerle kanıtlanmıştır.

ANESTEZİ Anestezi uzmanının işini görür, anestezi yöntemini seçer ve nasıl uygulanacağı konusunda fikir verir.

MODIS-2 Tansiyon hastalığının teşhisinde hastaya 30 soru sorarak hastalığın ne olduğunu ve nasıl tedavi edileceğini belirleyen bir uzman sistemdir.

JOSEPH Tıp öğrencileri tarafından EKG'yi okumak amacıyla geliştirilmiştir.

ONCO-HELP Tümörlerin teşhis ve tedavisinde kullanılmak amacıyla geliştirilmiştir. Laboratuvar bilgileri girildikten sonra ürün lokasyonunu, gelişimini ve cinsini tespit ederek yan etkilerini de inceler.

PHARM-2 Eczacıların ve doktorların oluşturduğu bir ekibin girdileriyle, ilaç tedavisini desteklemek amacıyla geliştirilmiştir.

QUAWDS Ayak ve bacak hastalıkları ve felç tanısını belirlemek amacıyla geliştirilmiştir. Hastanın yürüme analizini yapabilmektedir.

XDIS Yaklaşık 300 tane iç hastalık ve bu hastalıkların patolojik semptomları hakkında bilgi içeren, doktorlara yardımcı olmak amacıyla geliştirilmiş bir uzman sistemdir.

SETH Veri tabanında çok fazla zehirli madde ve ön teşhis için gerekli bilgiler bulunan, zehirli veya uyuşturucu maddelerin teşhis edilmesi amacıyla geliştirilmiştir.

Günümüzde yukarıda belirtilenlerin dışında çeşitli amaçlar için geliştirilmiş tıpta uzman sistemler bulunmaktadır. Örneğin: “Poems (ameliyat sonrası acil bakım), Dxplain (tanı belirleme), Oirs (medikal risk yönetimi), Dr. Cad (internet tabanlı tanı destek), Perfex (koroner kalp damar hastalıkları teşhisi)” gibi tıpta uzman sistemler bulunmaktadır.

7. ÖNERİLEN ALGORİTMA VE YÖNTEM

İşletmelerin problemlerini giderebilmek, stabilitesi yüksek ve aynı zamanda zeki bir hibrit sistem geliştirilmiştir. Guguk kuşu algoritması ile hazırlanan, Gezgin satıcı problemi üretilen çözüm sonuçları kullanılarak uzman bir sistem üzerinden gidilecek lokasyon ayarılması yapılan sistem sayesinde hem maliyet analizi kısmı geliştirilmiş, hemde işletmelerin önceliklerini anlık olarak düzeltebilme fırsatı sağlanmıştır.

Teknik servis hizmeti veren bir şirketin iş dağıtımın mekanizmasını geliştirmeye yönelik yapılan uygulama için Uzman Sistemin yapay zekâ ile zeki bir hal alması amaçlanmıştır.

Guguk kuşu algoritması kullanılarak hazırlanan mesafe ölçümü algoritması Uzman sisteme bir girdi oluşturmuş, Uzman Sistemde belirlenen 2 farklı kural tabanına göre sonuçlar gözlemlenecektir.

Guguk Kuşu Algoritması (GKA), guguk kuşunun çoğalma stratejilerinden faydalanılmış, bazı guguk kuşu türlerinin yuvalarında bulunan yumurtaları başka bir yuvaya bırakan guguk kuşu yavrularının hayatta kalma mücadelesinden ilham alınmıştır. [60]

Diğer sezgisel yöntemler gibi, Guguk Kuşu Algoritması da guguk kuşlarının yaşama başlangıç popülasyonu ile başlayan bir süreçtir. İlk olarak guguk kuşları yuvalarına bıraktıkları yumurtalarına sahiptirler. Başka bir yuvaya bırakılan yumurta, bırakıldığı yerdeki yumurtalar ile yüksek bir benzerlik göstermesi durumunda bu yumurtaların bazıları büyüme imkanına sahip olur ve zamanı geldiğinde olgun bir kuş olmayı sağlayabilirler. Benzerlik göstermeyen yumurtalar ise ev sahibi tarafından fark edilerek yok edilirler.

Gelişen bu yumurtalar o bölgedeki yuvaların uygunluğunu ortaya koymaktadır. O bölgede ne kadar çok yumurta hayatta kalırsa o bölgede o kadar kuşun yaşaması sağlanır. Bu nedenle, bu süreç için Guguk Kuşu Algoritmasının süreci iyileştireceği düşünülmektedir. Guguk kuşu algoritması, yumurtaların hayata tutunma oranını en yüksek oranda tutmak için yumurtalarını bırakabilecekleri en uygun alanı arar. Kalan yumurtalar büyüyüp olgun kuşlara dönüştükten sonra, guguk kuşları birkaç toplum oluşturur. Her toplumun içinde yaşayabileceği bir yaşam alanı vardır. Tüm gruplar için, içindeki en iyi yaşam alanı, diğer topluluklardaki guguk kuşları için yeni bir hedef olacaktır. Hedef habitat diğerlerin ayrıştıktan sonra guguk kuşlarıda en iyi habitata doğru göç eder. Guguk kuşları, en iyi

lokasyonun herhangi bir yerinde yaşamaya devam edecektir. Her guguk kuşunun ait yumurtalar ve guguk kuşunun en iyi lokasyona uzaklığı dikkate alınarak, yuvadaki yumurtalar en iyi habitata doğru gönderilecektir. Bu süreç, popülasyonun büyük bir kısmının aynı lokasyon etrafında birikinceye ve maksimum kazanç değerine sahip en iyi nokta elde edilinceye kadar devam edecektir (Zheng ve ark. 2012). [74]

Guguk kuşu algoritmasında her bir yumurta gerçek hayatta bir çözümü temsil eder ve her bir oluşan yeni guguk kuşu yumurtası yeni bir çözümü oluşturduğunu gösterir. Bu sayede yuvada kendine ait daha kötü bir yumurta (çözüm) yerine daha iyi bir yumurta (çözüm) ile yer değiştirerek görece potansiyeli daha fazla olan yumurtanın hayatta kalmasını sağlamaktadır.

Sezgisel algoritmalar genel ve yerel aramaları gözden geçirirken, mantık olarak en makul çözüme temas etmektir. Ancak diğer iyi çözümleri kaçırmaması içinde sürekli arayışta bulunması gerekmektedir. Böyle olduğu için, Guguk kuşu algoritması genel(global) rassal yürüyüş ve yerel rassal yürüyüş yaklaşımlarının bir arada götürülerek kullanıldığı bir algoritma haline gelmiştir. İfade edilen rassal yürüyüşler manevra parametresi tarafından kontrol edilmektedir. Global rassal yürüyüş Levy uçuşu ile takip edilmektedir. Levy uçuşu, yeni nesilleri hayatta tutabilmek için gerçekleştirilir. Guguk kuşu araması anlatılan üreme çeşitliliğine göre kurgulandığı için optimizasyon problemlerinde uygulanabilmektedir. Guguk kuşu arama algoritması uygulamalarda diğer meta-sezgisel yöntemlere kıyas daha iyi sonuçlar ortaya koyduğu bilinmektedir.

Popülasyon Başlatma

Bir optimizasyon problemine çözüm üretmek amacıyla probleme ait değişkenleri bir dizide saklamak gerekir. Terminolojide bu veriler “Kromozom” ve “Parçacık Konumu” olarak adlandırılmaktadır. Guguk Kuşu Algoritması’da bu dizi seti, kuşların yaşadığı ve büyüdüğü doğal yaşam ortamı olarak ifade edilen “ortam” kelimesiyle ifade edilmektedir. [75]

Yumurtlama süreci

Yumurtalar, yumurtlama yarıçapı göz önünde bulundurularak merkez etrafında rastgele yerleştirilir. Yumurtalar farklı yuvalara gönderildikten sonra ev sahibi tarafından fark edilen

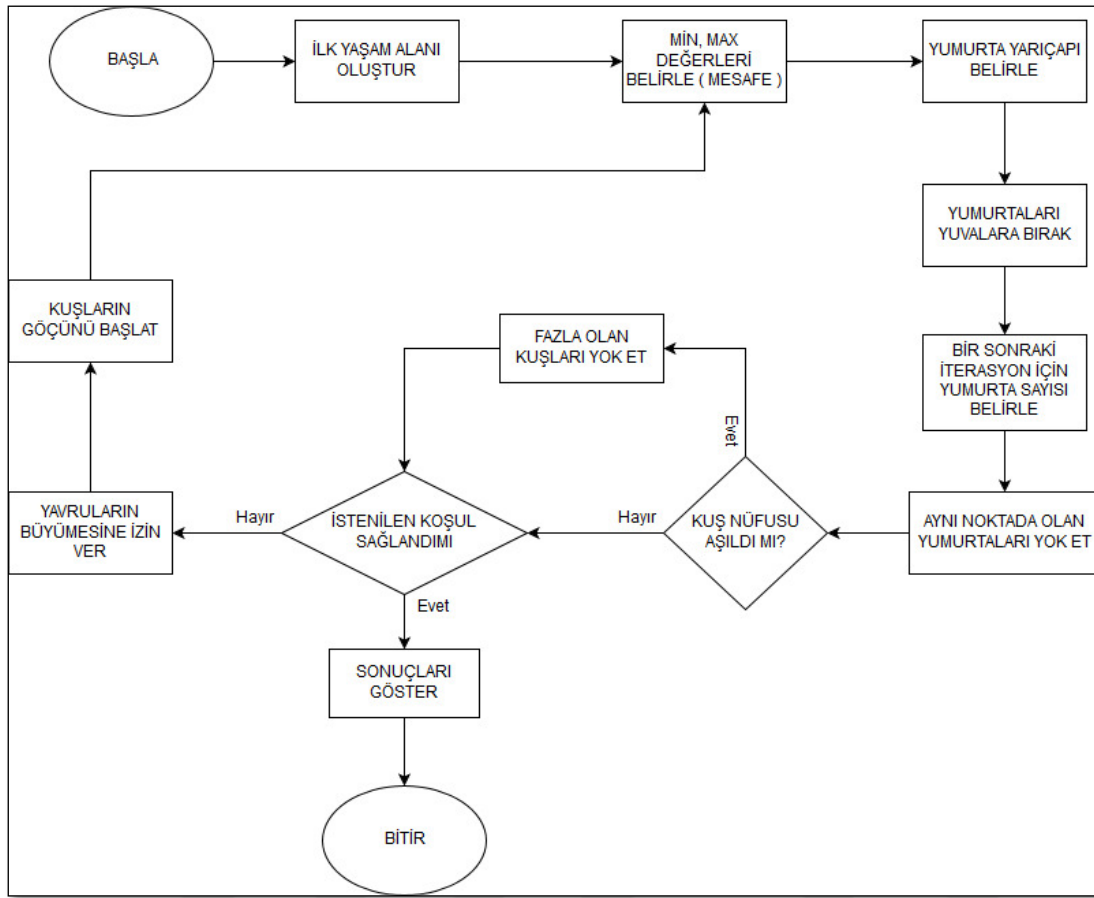
yumurtalar yuvadan gönderilir. Kalan yumurtalar arasında kabul edilmeyen yumurtalar %p (genelde %10) kadarı silinir. [75] Algoritmada yaşamda kalan guguk kuşu sayısı kontrol edilerek kaydedilir. Bu nedenle her iterasyonda en güçlü yumurtalar hayata devam edecektir.

Guguk kuşlarının göç etmesi

Guguk kuşlarının farklı lokasyonlarda kümeleştikten sonra en iyi verime ya da en iyi maliyete sahip olan grup diğer guguk kuşlarını için gidilmesi gereken nokta olarak seçilmektedir. [75] Guguk kuşu gruplarını belirlemek amacıyla K-Ortalama Kümeleme Algoritması (KKA) kullanılmaktadır.

Guguk kuşu optimizasyon algoritmasının sözde kodu:

1. İlk yaşam için başlangıç noktası belirlenir.
2. Her kuş için belirli bir sayıda rastgele yumurtalar belirlenir.
3. Her kuş için en fazla yumurtlama yarıçapı tespit edilir.
4. Belirlenen yumurtlama yarıçapı alanı içinde herhangi bir noktaya yumurta bırakılır.
5. Ev sahibi tarafından fark edilebilen yumurtaların yaşamına son verilir.
6. Fark edilmeyen yumurtaların büyümesine müsaade edilerek, civcivlerin çıkması ve büyümesi sağlanır.
7. Yetişen her kuşun maliyet analizi yapılarak, değerlendirilir.
8. Belirlenen alan içerisinde hayatta kalabilecek kuş adedi sınırlandırılır ve gereksiz görülen alanlar yok edilir.
9. En iyi maliyete sahip kuş grubu tespit edilir ve hedef nokta tayin edilir.
10. Yeni guguk kuşların hedef noktaya göç etmesi sağlanır.
11. İstenilen şart kontrol edilir, sağlanmamış ise ikinci adımdan geri dönülür.



Şekil 7.1. Guguk kuşu arama algoritması

Sezgisel bir yöntem olarak tasarladığımız Guguk Kuşu Algoritması önemli bir optimizasyon problemi olan Araç Rotalama Problemine uygulanmıştır. (Şekil 7.1.) İlgili program C# Programlama Dili kullanılarak hazırlanmıştır. Elde edilen sonuçlara göre toplamda 5 iterasyonda istenilen sonuçlara ulaşılabilirdiği görülmüştür. (Resim 7.2) Bu sayede Uzman Sistemimizdeki Mesafe kriteri için çıktılar bu algoritma üzerinden alınacaktır.

GUGUK KUŞU ARAMA ALGORİTMASI

Koordinatlar

Konum	X	Y
1. Nokta	39,9207	32,8541
2. Nokta	40,5759	31,5788
3. Nokta	41,0052	28,9769
4. Nokta	37,8666	32,4833
5. Nokta	40,9833	27,5166
6. Nokta		
7. Nokta		
8. Nokta		
9. Nokta		

Koordinatlar Kaydet

Algoritmayı Çalıştır

Temizle

Toplam İşlem Süresi(msn): 58

SONUÇLAR

```

[0,0] =37
[0,1] =35,321333
[1,0] =39,92077
[1,1] =32,85411
[2,0] =40,575977
[2,1] =31,578809
[3,0] =41,00527
[3,1] =28,97696

```

Resim 7.1. Guguk kuşu algoritması programı

GUGUK KUŞU ARAMA ALGORİTMASI

Koordinatlar

Konum	X	Y
1. Nokta	39,9207	32,8541
2. Nokta	40,5759	31,5788
3. Nokta	41,0052	28,9769
4. Nokta	37,8666	32,4833
5. Nokta	40,9833	27,5166
6. Nokta		
7. Nokta		
8. Nokta		
9. Nokta		

Koordinatlar Kaydet

Algoritmayı Çalıştır

Temizle

Toplam İşlem Süresi(msn): 58

SONUÇLAR

```

[8,1] =0
[9,0] =0
[9,1] =0
Koordinatlar Kaydedilmiştir.
[16,1] =0
[68,2] =0
[280,3] =0
[1132,4] =0

```

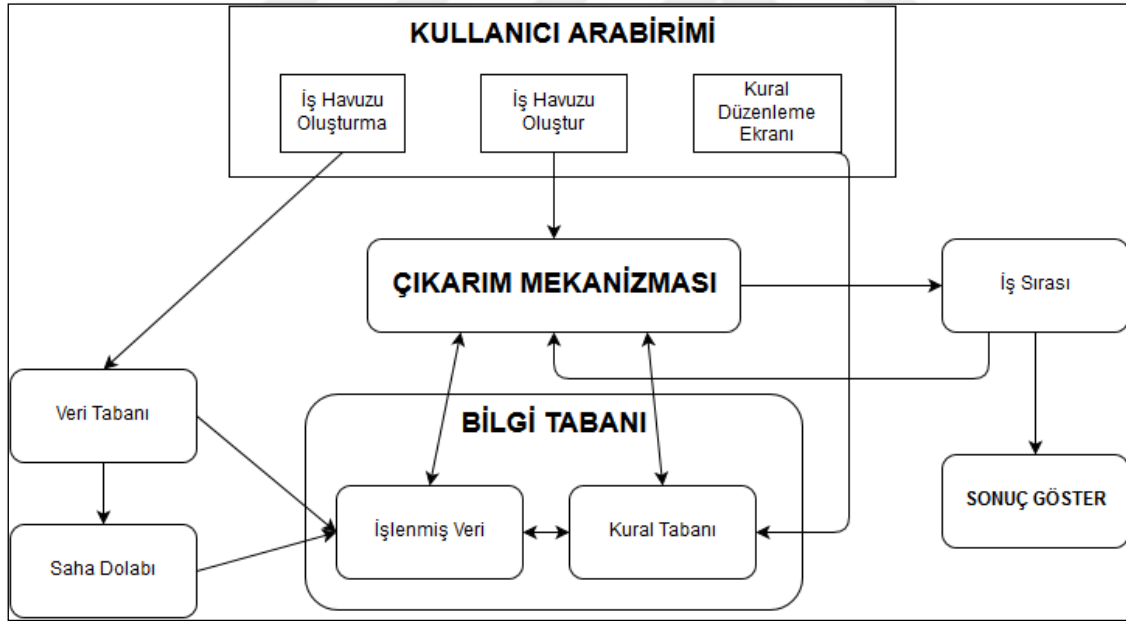
Toplam 5 iterasyonda işlem tamamlanmıştır.
En Kısa Mesafe: 1058,59323786593 km'dir.
En Kısa Mesafe Rotası:
0
4
1
2
3
5

Resim 7.2: Guguk kuşu algoritması sonuç

Rota Takip Uzman Sistemi yapay zekâ algoritması tarafından sıraya dizilen iş havuzunun bazı kısıtlara göre (müşteri memnuniyeti, önceliklendirme, maliyet vb.) tekrar sıralanmasını amaçlamaktadır. Kriterler uzun yıllardır sevk ve idare işlemini gerçekleştiren uzmanlar tarafından belirlenerek hazırlanmıştır. Yapılan Uzman Sistemde iş havuzunun

oluşmasını sağlayacak bir bilgi girişi ekranı oluşturulmuştur. (Resim 7.4) İstenilen bilgilere göre bilgi girişi yapılan işlerin sistem için tasarlanan veri tabanına aktarılmış, oluşturulan veri tabanı havuzundan çıkartılan anlamlı bilgileri bilgi tabanına kaydedilmiştir. Belirlenen kriterlere göre kural oluşturulması sağlanacak kural tabanı ekranı (Resim 7.6) sayesinde yeni kural belirlenebilmekte ya da var olan kurallar güncellenebilmektedir. Rota oluştur butonu sayesinde anlamlı veriler ile oluşan iş havuzundaki işler, kural tabanında oluşturulan kurallara göre sıraya sokulmaktadır. Her kural için bir etki puanı belirlenmiş, tüm kuralların çalışması sonucunda oluşan en yüksek etki puanına göre iş havuzu tekrar sıralanarak gidilecek ilk iş belirlenmiştir. Belirtilen yöntemin çalışma diyagramı Şekil 7.2.'de gösterilmiştir.

Uzman sistem uygulamamız Kural Tabanı, İş Havuzu, Rota Oluştur olmak üzere 3 ekrandan oluşmuştur. Veri tabanı, Bilgi Tabanı, Kural tabanı, Saha Dolabı ve İş Sırası olmak üzere 5 adet veri tablosu üzerinden çalışmaktadır.



Şekil 7.2. Rota Takip Uzman Sistem Çalışma Yapısı

7.1. Rota Takip Kullanıcı Arabirimi

Rota Takip Uzman sisteminde Kullanıcı arayüzü Kural Tabanı, Saha Dolabı, İş Havuzu Oluştur, Rota Oluştur, Bilgi Tabanı ve İş Sırası butonlarından oluşmaktadır. (Resim 7.3) Belirtilen program Visual Studio ortamında C# programlama dili kullanılarak hazırlanmıştır.



Resim 7.3. Rota takip uzman sistem ana menü

İş Havuzu oluştur butonu ile bilgi girişi yapılabilecek ekran erişilerek iş ile ilgili bilgiler girilmektedir. (Resim 7.4). İş kriterleri ile ilgili bilgiler kullanıcı marifiyeti ile girilerek veri tabanına, buradan da anlamlı veriler haline gelmesini sağlayacak güncellemeler ile işlenmiş veri tabanına aktarılmaktadır.

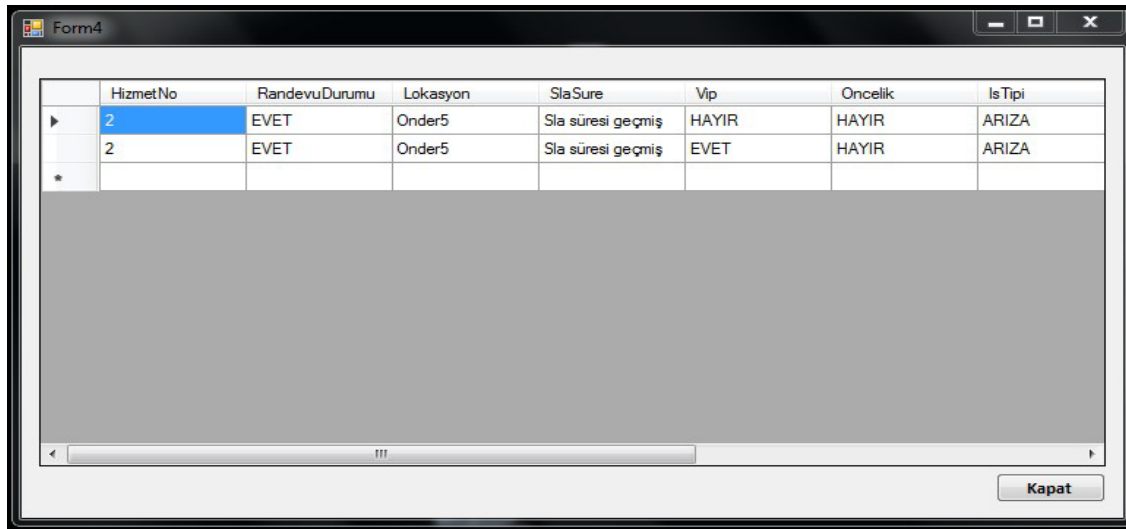
HizmetNo	BildirimZamanı	RandevuZamanı	Vip	Öncelik	İşTipi	Lokasyon
1	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder
2	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder1
3	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder2
4	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder3
5	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder4
6	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder5
7	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder6
8	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder7
1	11.11.2016	01.01.1900	HAYIR	HAYIR	ARIZA	Onder

Resim 7.4. Rota takip uzman sistem bilgi giriş ekranı

7.2. Rota Takip Bilgi Tabanı

Bilgi tabanı bir veya birden çok uzmanın belirli bir konuda yetkinliklerini ifade eden bilgilerin bir araya getirilmesiyle oluşturulur [76]. Bilgi tabanı; kural tabanı ve veri tabanından oluşturulmuştur. Veri tabanı; problemi anlatan doğrular ve herhangi bir ana kadar elde edilen işlenmiş bilgiden oluşur [74]. Uygulamada 2 adet bilgi veri tabanı bulunmaktadır. Bunlar bilgi iş havuzunun oluşması için gereken ham veriler ile bu verilerden elde edilen işlenmiş veri tabanıdır. Bilgi tabanını aşağıdaki kriterler oluşturmaktadır. (Resim 7.5)

- Hizmet No
- Randevu Durumu
- Lokasyon
- Sla Suresi
- Vip
- Öncelik
- İş Tipi
- Mesafe
- Koordinatx
- Koordinaty
- İş Süresi



	HizmetNo	RandevuDurumu	Lokasyon	SlaSure	Vip	Oncelik	Is Tipi
▶	2	EVET	Onder5	Sla süresi geçmiş	HAYIR	HAYIR	ARIZA
	2	EVET	Onder5	Sla süresi geçmiş	EVET	HAYIR	ARIZA
*							

Resim 7.5. Rota takip uzman sistem bilgi tabanı görüntüsü

Bilgi tabanında bulunan Mesafe, Koordinatx, Koordinaty kriterleri uygulamada ayrı bir yerde bulunan veri tabanında saklanmakta ve gidilecek işlerin lokasyonunu tayin etmektedir. Saha dolabı veri tabanında işlerde kullanılacak olan lokasyonları koordinat bilgileri ve ana merkeze olan uzaklığını içermektedir. Bilgi girişi yapılırken işle ilgili lokasyon seçimi sonrasında lokasyonun merkeze olan uzaklığı Mesafe kriterine, Müşterinin bulunduğu lokasyonun koordinat bilgileri ise Koordinat kriterlerine eklenmiştir.

Diğer kriterler bilgi giriş ekranında girilen diğer bilgiler sayesinde oluşturulmuştur. Örneğin; Bir işin VIP olması giriş anında belirlenen kritere göre işlenmektedir.

Kural Tabanı; Uzman sistem, belirli halleder çalışan ve “eğer-o halde” gibi bir mantıkla tanımlanan kuralları depolamak için bir veri tabanına sahip olmalıdır [76]. Uygulamamızda oluşturulan kural tabanı uzman kişilerce belirlenen, işlenmiş veri tabanındaki işlerin sıralanması için değerlendirebilecek niteliklere göre hazırlanmıştır (Resim 7.6). Kurallar incelenmesi istenen kriterler için girilen bilgiler ile oluşturulmaktadır. Kural tabanında kontrol ettiğimiz kriterler aşağıdaki gibidir.

- Randevu Durumu
- Sla Suresi
- Vip
- Öncelik
- İş Tipi
- Mesafe
- İş Süresi
- Etki Puanı

Her kural için bir miktar etki puanı belirlenerek işlerin sıraya girmesi amaçlanmıştır. Bu kapsamda hizmet veren işletmeler müşteri memnuniyet, maliyet ve işin verimli yapılabilmesi için kuralların etki puanlarını odak noktalarına anlık olarak değiştirebilmektedir. Kurallar çalışmanın yapıldığı saha koşulları, işin etkinliğinin önemi ve işletmeciye göre değişebilmekte ve bunların oluşturduğu etki puanında güncellenebilmektedir. Bu sayede oluşturulan kuralların günün koşullarına göre güncelliğinin sağlanabilmesi, uzun yıllar bu alanda sevk ve idare yapmış uzmanların yaklaşım tarzlarını uygulamaya yansıtabilmeleri sağlanmıştır. Kuralların saklandığı veri

tabanı MySQL ortamında sağlanmıştır. (Resim 7.7) Uygulamanın değerlendirilmesi için iki farklı kural tabanı tayin edilmiş, sırasıyla Nitelik Kural Tabanı ve Hassas Kural tabanı olarak adlandırılmıştır.

Nitelik Kural Tabanı; Uzmanlar tarafından belirlenen Kural tabanı kriterlerinin hepsine diğerleri ile kıyaslandığında uzmanlar tarafından bir etki puanı verilmiş, bu sayede 22 farklı kural meydana gelmiştir. (Resim 7.8).

Hassas Kural Tabanı; Kural tabanı için belirlenen kriterlerin birbirleri ile olan ilişkileri üzerine oluşturulan 40 adet kuralın Uzmanlar tarafından yorumlanarak bir etki puanı verilmesiyle oluşturulmuştur (Resim 7.9).

The screenshot shows a software window titled 'Form5'. It contains a form with several dropdown menus and text input fields. The form fields are: 'Randevu Durumu:', 'Mesafe', 'SLA Süre:', 'İş Süresi:', 'Vip Durumu:', 'Etki Puanı:', 'Öncelik Durumu:', and 'İş Tipi:'. To the right of the form are buttons: 'Kaydet', 'Güncelle', 'Sil', 'Temizle', and 'Kapat'. Below the form is a table with the following columns: 'Kural_ID', 'RandevuDuru', 'SlaSure', 'Vip', 'Oncelik', 'IsTipi', 'Mesafe', and 'IsSure'. The table contains 14 rows of data, with the first row (Kural_ID: 8) highlighted in blue. The 'Mesafe' column contains values: 'Aynı Lokasyon', '100 metreye Kadar', '200 metreye Kadar', '400 metreye Kadar', '800 metreye Kadar', '1 kmye Kadar', and 'Uzak Lokasyon'.

Kural_ID	RandevuDuru	SlaSure	Vip	Oncelik	IsTipi	Mesafe	IsSure
8						Aynı Lokasyon	
9						100 metreye Kadar	
10						200 metreye Kadar	
11						400 metreye Kadar	
12						800 metreye Kadar	
13						1 kmye Kadar	
14						Uzak Lokasyon	
*							

Resim 7.6. Rota takip uzman sistem kural tabanı

Results		Messages							
	Kural_ID	RandevuDurumu	Slasüre	Vip	Oncelik	IsTipi	Mesafe	IsSure	EtkiPuani
1	81	EVET					Aynı Lokasyon		7
2	82					ARIZA	Aynı Lokasyon		8
3	83					SİPARİŞ	Aynı Lokasyon		7
4	84						Aynı Lokasyon	15 dk.	8
5	85						Aynı Lokasyon	30 dk.	7
6	86				EVET				2
7	87			EVET					4
8	88				EVET	ARIZA			4
9	89				EVET	SİPARİŞ			3
10	90				EVET			15 dk.	4
11	91				EVET			30 dk.	3
12	92			EVET		ARIZA			6
13	93			EVET		SİPARİŞ			5
14	94			EVET				15 dk.	6
15	95			EVET				30 dk.	5
16	74						Aynı Lokasyon		6
17	75						100 metreye ...		4
18	76						200 metreye ...		3
19	77						400 metreye ...		2
20	78						800 metreye ...		1
21	79						1 km'ye K...		0
22	80						Uzak Lokas...		-1
23	96		Slasü...			SİPARİŞ			-1
24	97		2 saat...			ARIZA			2
25	98		2 saat...			SİPARİŞ			1

Resim 7.7. Rota takip uzman sistem kural listesi

Kural Sayısı	Kurallar							Etki Puani
	Randevu Durumu	SLA Süre	Vip Durumu	Öncelik Durumu	İş Tipi	İş Süresi	Mesafe	
1	Evet							1
2	Hayır							0
3		SLA süresi geçmiş						-2
4		2 saatten az						0
5		4 saatten az						4
6		8 saatten az						2
7		8 saatten fazla						1
8			Evet					4
9			Hayır					0
10				Evet				2
11				Hayır				0
12					Arıza			2
13					Sipariş			1
14						15 dk.		2
15						30 dk.		1
16							Aynı Lokasyon	6
17							100 metreye kadar	4
18							200 metreye kadar	3
19							400 metreye kadar	2
20							800 metreye kadar	1
21							1 km'ye kadar	0
22							Uzak Lokasyon	-1

Resim 7.8. Nitelik kural tabanı

Kural Sayısı	Kurallar							Etki Puanı
	Randevu Durumu	SLA Süre	Vip Durumu	Öncelik Durumu	İş Tipi	İş Süresi	Mesafe	
1							Aynı Lokasyon	6
2							100 metreye kadar	4
3							200 metreye kadar	3
4							400 metreye kadar	2
5							800 metreye kadar	1
6							1 km'ye kadar	0
7							Uzak Lokasyon	-1
8	EVET						Aynı Lokasyon	7
9					Arıza		Aynı Lokasyon	8
10					Sipariş		Aynı Lokasyon	7
11						15 dk.	Aynı Lokasyon	8
12						30 dk.	Aynı Lokasyon	7
13				Evet				2
14			Evet					4
15				Evet	Arıza			4
16				Evet	Sipariş			3
17				Evet		15 dk.		4
18				Evet		30 dk.		3
19			Evet		Arıza			6
20			Evet		Sipariş			5
21			Evet			15 dk.		6

Resim 7.9. Hassas kural tabanı

İki kural tabanının değerlendirilmesi ile oluşan sonuçlar Uzman kişilerin görüşleri ile karşılaştırıldığında Hassas kural tabanının Nitelik kural tabanına göre daha etkin ve doğru tespitler yaptığı anlaşılmıştır. Resim 7.10'da bulunan karşılaştırma tablosunda uzman görüşüne göre kural tabanlarının doğruluk oranları %40'tır. Yapılan incelemelerde işin uzmanının sıralamayı belirlerken 2 farklı müşterinin aynı lokasyonda olma durumuna dikkat etmediği, Hassas kural tabanının ise bu işlerin aynı noktalardan gelmesi nedeniyle sıralamayı arka arkaya yaptığı tespit edilmiştir. Bu düzeltmenin Uzman kişinin yaptığı sıralamada yapılması ile beraber Hassas kural tabanında başarı oranı %80'e kadar çıkmıştır.

Deney Veri Seti								Kural Tabanları Karşılaştırma		
Hizmet No	Bildirim Zamanı	Randevu Zamanı	VIP	Öncelik	İş Tipi	Lokasyon	İlk Mesafe	Uzman Görüşü	Hassas Kural Tabanı	Nitelik Kural Tabanı
1	3.10.22 5:00	3.11.22 10:00		EVET	ARIZA	Onder7	10	4	6	1
2	3.10.22 5:00	3.11.22 11:00			ARIZA	Onder5	211	8	9	8
3	3.8.22 5:00	3.11.22 12:00			SİPARİŞ	Onder3	333	5	7	6
4	3.8.22 10:00	3.11.22 13:00			SİPARİŞ	Onder2	427	9	8	9
5	3.10.22 10:00				ARIZA	Onder1	811	1	1	2
6	3.8.22 5:00		EVET	EVET	SİPARİŞ	Onder	520	2	2	3
7	3.10.22 10:00		EVET	EVET	ARIZA	Onder4	498	3	3	4
8	3.10.22 10:00			EVET	ARIZA	Onder4	498	6	4	5
9	3.8.22 5:00	3.11.22 16:00	EVET		SİPARİŞ	Onder6	314	7	5	7
10	3.6.22 16:00				ARIZA	Onder1	811	10	10	10
								Başarı Oranı	40%	40%

Resim 7.10. Kural tabanı karşılaştırma

7.3. Rota Takip Çıkarım Mekanizması

Çıkarım mekanizması elde edilen bilginin kontrol edilerek yorumladığı bölümdür. Çıkarım mekanizması, kuralların oluşturduğu veri yığnında anlamlı sonuçların çıkarıldığı ve kullanıcılara bu sonuçların gösterildiği alandır. Çıkarım mekanizmasında kuralları yorumlamak için ileri ve geriye zincirleme metotları kullanılmaktadır [74]. Uygulamamızda çıkarım mekanizması için ileri zincirleme metodu kullanılmıştır.





8. SONUÇLAR

Uygulama yöntemi; kuralların her iş için oluşturduğu etki puanını hesaplanarak en yüksek puanı alan iş için sıralama belirlenmektedir. Belirlenen iş mevcut havuzdan silindikten sonra kalan işler için kural mekanizması çalışarak tekrardan bir etki puanı hesaplanıp hangi işin sırada olduğu tayin edilmektedir.

Bu aşamalar sırasında Mesafe kriterinin her işlemde bir önceki iş merkez kabul edilerek guguk kuşu algoritmasına göre tekrardan sıralamaya koyulup;

- Aynı Lokasyon (İlk sıradaki iş)
- 100 metreye kadar olan (ikinci sıradaki iş)
- 200 metreye kadar olan (üçüncü sıradaki iş)
- 400 metreye kadar olan (dördüncü sıradaki iş)
- 800 metreye kadar olan (beşinci sıradaki iş)
- 1 km.ye kadar olan (altıncı sıradaki iş)
- Uzak Lokasyon (yedinci ve üzeri sıradaki iş)

Şeklinde işaretlenerek bir etki puanı alması sağlanmaktadır. Yapay zekâ algoritmasıyla yüksek bir hız, daha verimli bir çözümle hesaplanan ve maliyet anlamında sadece en düşük yol mesafesini bulmaya amaçlayan guguk kuşu algoritması üzerinden yapılan sıralama Uzman sisteme yüksek bir oranda etki edecek Mesafe kriterine bir girdi olarak kullanılmıştır. Böylece tüm iş havuzundaki işler sıralamaya koyuldukça tekraren kural tabanı mekanizması kalan işler için hesaplanmış ve iş sıralaması oluşturulmuştur (Resim 8.1).

HizmetNo	Lokasyon	İsSırası
6	Onder	1
9	Onder6	2
7	Onder4	3
8	Onder4	4
1	Onder7	5
2	Onder5	6
3	Onder3	7
4	Onder2	8
5	Onder1	9
10	Onder1	10
*		

Resim 8.1. Rota takip uzman sistem iş sırası listesi

Günümüzde iş planlarının verimli yapılabilmesi, bu planlamada insan gücünün en az şekilde harcanması, oluşan maliyetlerin en aza indirilmesi ve müşteri memnuniyetinin bu koşullarda sağlıklı bir şekilde sağlanabilmesi işletmeler açısından çok önemli ve kritiktir.

Çalışmamda Uzman sistemin ile yapay zekâ arasında hibrit bir sistem oluşturularak sonuçların matematiksel hesaplara göre daha verimli, hızlı ve zeki bir hal alması sağlanmıştır. Bu sayede oluşturulan yapının kendini güncelleyebilmesi, öğrenilen lokasyon bilgisine mesafe kriterinin zamanla daha doğru sonuçlar verebilmesi, uzman sistem yaklaşımı sayesinde sadece doğru mesafe kriteri değil uzman yaklaşımlarının işletmenin isteklerine karşılık verebilmesi sağlanmıştır.

Uzman sistem için kural yapıları farklı 2 kural tabanı denenmiş ve en verimli olduğu düşünülen, diğer kriterlerin birbirleri ile oluşan ilişkilerinin gözetildiği Hassas kural tabanı

kullanılarak daha daha net sıralamaların oluřtuđu gözlemlenmiřtir. Yapay zekâ mesafe belirleme yöntemiyle Hassas kural tabanının senkron çalışabilmesi uygulamanın en önemli çıktılarından birisidir. Sadece maliyet ve verimlilik ön planda tutulduğunda uzmandan daha iyi bir sıralamayı guguk kuřu algoritması ile sağlanabildiđi görölmüş, İşletmeci istekleri ile harmanlaşan kural yapısı ile hem müşteri memnuniyet hemde işletmeci verimliliđi artırılması sağlandığı tespit edilmiştir. İlk olarak kullanılan Nitelik kural tabanı-yapay zekâ hibrit sisteminin başarımlarının bir hayli düşük olması nedeniyle tekrardan oluřturulan Hassas kural tabanı arasındaki başarı farkı yaklaşık 2 kattır. Hibrit bir yapıda uzman sistem kural yapısının dikkatli bir şekilde belirlenmesi gerektiđi denenen kural yapısında oluřan farklılık ve başarımların farkından dolayı net bir şekilde anlaşılmıştır. Oluřturulan bu yapı sayesinde müşteri memnuniyeti ve verimliliğin yüksek seviyede olduđu maliyetin ise o denli azalacağı görölmüřtür. Bu yapıda kurulan hibrit uzman sistem bu alanda tasarlanacak diđer uzman sistemlere yardımcı olabilecektir.

Bundan sonra yapılacak çalışmalarda diđer kriterlere de yapay zekâ üzerinde bir girdi verilmesi, oluřturulan sıralamaların veri tabanında tutularak ilerleyen süreçlerde veri madenciliđi çalışmalarıyla işlenip anlamlı verilere ulaşılması, ulařılan anlamlı veriler ile yeni kuralların sistem tarafından otomatik olarak yazılabilmesi, anlık olaylara bu kurallar sayesinde müdahale edilebilmesi, yapay zekâ kullanılarak veri tabanında tutulan eski kayıtların incelenerek servis hizmeti ile ilgili noktasal öngörü çözümlerinin üretilmesi, bakım faaliyetlerine buralardan girdi çıkartılabilmesi sağlanabilir.



KAYNAKLAR

1. Ai, T. J., Kachitvichyanukul, V., (2009). "A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery", *Computers & Operations Research*, 36 (5), 1693-1702.
2. Alfredo Tang Montané, F., Galvão, R. D., (2006). "A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service", *Computers & Operations Research*, 33 (3), 595-619.
3. Avci, M., Topaloglu, S., (2015). "An adaptive local search algorithm for vehicle routing problem with simultaneous and mixed pickups and deliveries", *Computers & Industrial Engineering*, 83, 15-29
4. Bianchessi, N., Righini, G., (2007). "Heuristic algorithms for the vehicle routing problem with simultaneous pick-up and delivery", *Computers & Operations Research*, 34 (2), 578-594.
5. Çatay, B. (2010). "A new saving-based ant algorithm for the Vehicle Routing Problem with Simultaneous Pickup and Delivery", *Expert Systems with Applications*, 37 (10), 6809-6817.
6. Clarke, G., Wright, J. W., (1964). "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points", *Operations Research*, 12 (4), 568-581.
7. Crispim, J., Brandao, J., (2005). "Metaheuristics applied to mixed and simultaneous extensions of vehicle routing problems with backhauls", *J Oper Res Soc*, 56 (11), 1296-1302.
8. Dantzig, G. B., Ramser, J. H., (1959). "The Truck Dispatching Problem", *Management Science*, 6 (1), 80-91.
9. Dell'Amico, M., Righini, G., Salani, M., (2006). "A branch-and-price approach to the vehicle routing problem with simultaneous distribution and collection", *Transportation Science*, 40 (2), 235-247.
10. Dethloff, J. (2001). "Vehicle routing and reverse logistics: The vehicle routing problem with simultaneous delivery and pick-up", *OR Spektrum*, 23 (1), 79- 96.
11. Gajpal, Y., Abad, P., (2009). "An ant colony system (ACS) for vehicle routing problem with simultaneous delivery and pickup", *Computers & Operations Research*, 36 (12), 3215-3223.
12. Goksal, F. P., Karaoglan, I., Altiparmak, F., (2013). "A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery", *Computers & Industrial Engineering*, 65 (1), 39-53.
13. Jun, Y., Kim, B.-I., (2012). "New best solutions to VRPSPD benchmark problems by a perturbation based algorithm", *Expert Systems with Applications*, 39 (5), 5641-5648.

14. Li, J., Pardalos, P. M., Sun, H., Pei, J., Zhang, Y., (2015). "Iterated local search embedded adaptive neighborhood selection approach for the multi-depot vehicle routing problem with simultaneous deliveries and pickups", *Expert Systems with Applications*, 42 (7), 3551-3561.
15. Nagy, G., Salhi, S., (2005). "Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries", *European Journal of Operational Research*, 162 (1), 126-141.
16. Polat, O., Kalayci, C. B., Kulak, O., Günther, H.-O., (2015a). "A perturbation based variable neighborhood search heuristic for solving the Vehicle Routing Problem with Simultaneous Pickup and Delivery with Time Limit", *European Journal of Operational Research*, 242 (2), 369-382.
17. Ropke, S., Pisinger, D., (2006). "A unified heuristic for a large class of Vehicle Routing Problems with Backhauls", *European Journal of Operational Research*, 171 (3), 750-775.
18. Salhi, S., Nagy, G., (1999). "A Cluster Insertion Heuristic for Single and Multiple Depot Vehicle Routing Problems with Backhauling", *The Journal of the Operational Research Society*, 50 (10), 1034-1042.
19. Subramanian, A., Drummond, L. M. A., Bentes, C., Ochi, L. S., Farias, R., (2010). "A parallel heuristic for the Vehicle Routing Problem with Simultaneous Pickup and Delivery", *Computers & Operations Research*, 37 (11), 1899-1911.
20. Subramanian, A., Uchoa, E., Pessoa, A., Ochi, L., (2013). "Branch-cut-and-price for the vehicle routing problem with simultaneous pickup and delivery", *Optim Lett*, 7 (7), 1569-1581.
21. Subramanian, A., Uchoa, E., Pessoa, A. A., Ochi, L. S., (2011). "Branch-and-cut with lazy separation for the vehicle routing problem with simultaneous pickup and delivery", *Operations Research Letters*, 39 (5), 338-341.
22. Tovey, C. A. (2002). "Tutorial on Computational Complexity", *Interfaces*, 32 (3), 30-61.
23. Wassan, N., Wassan, A. H., Nagy, G., (2008). "A reactive tabu search algorithm for the vehicle routing problem with simultaneous pickups and deliveries", *J Comb Optim*, 15 (4), 368-386.
24. Zachariadis, E. E., Kiranoudis, C. T., (2011). "A local search metaheuristic algorithm for the vehicle routing problem with simultaneous pick-ups and deliveries", *Expert Systems with Applications*, 38 (3), 2717-2726.
25. Zachariadis, E. E., Tarantilis, C. D., Kiranoudis, C. T., (2009). "A hybrid metaheuristic algorithm for the vehicle routing problem with simultaneous delivery and pick-up service", *Expert Systems with Applications*, 36 (2, Part 1), 1070-1081.
26. Zachariadis, E. E., Tarantilis, C. D., Kiranoudis, C. T., (2010). "An adaptive memory methodology for the vehicle routing problem with simultaneous pick-ups and deliveries", *European Journal of Operational Research*, 202 (2), 401-411.

27. Laporte, G., Nobert, Y., & Taillefer, S. (1988). Solving a family of multi-depot vehicle routing and location-routing problems. *Transportation science*, 22(3), 161-172.
28. Toth, P., & Vigo, D. (Eds.). (2002). The vehicle routing problem. Society for Industrial and Applied Mathematics.
29. Tan, K. C. (2001). "A framework of supply chain management literature." *European Journal of Purchasing & Supply Management*, 7(1), 39-48.
30. Eryavuz, M. (2001). *Araç Rotalama Problemi ve Örnek Bir Uygulama*, Yayınlanmamış Yüksek Lisans Tezi. Gazi Üniversitesi Fen Bilimleri Enstitüsü.
31. Tokaylı, M. A. (2005). *Zaman Pencere Aracı Rotalama Problemi İçin Bir Karar Destek Sistemi*. Yüksek Lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü Ankara. 17-75.
32. Erel, R. (1995). Taşıt rotalaması ve çizelgelemesi: Otobüsle kentlerarası yolcu taşımacılığı için bir model. Doktora Tezi, Yıldız Teknik Üniversitesi, İstanbul, 31-95.
33. Desrosiers, J., Dumas, Y., Solomon, M. M., and Soumis, F. (1995). Time constrained routing and scheduling. *Handbooks in operations research and management science*, 8, 35-139.
34. Bayzan, Ş., Çetin, M. ve Uğur, A. (2002). Araç Rotalama Probleminde Araç Rotalarının Tespitinde En Kısa Yol Yaklaşımı: Denizli Örneği. *Mühendislik Fakültesi Pamukkale Üniversitesi, Kınıklı, Denizli*.
35. Ballou, R.H., (1999). Business Logistics Management: Planning, Organizing and Controlling The Supply Chain, McGraw-Hill.
36. Cordeau, J. F., Gendreau, M., Laporte, G., Potvin, J.Y., Semet, F., (2002). "A Guide To Vehicle Routing Heuristics", *Journal of the Operational Research Society*, 53, 512-522.
37. Laporte, G., Gendreau, M., Potvin, J. Y., Semet, F., (2000). "Classical And Modern Heuristics For The Vehicle Routing Problem", *International Transactions in Operation Research*, 7, 285-300.
38. Özalp, V.M., (1995). *Gezgin Satıcı Problemi*, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 23-48, İstanbul.
39. Demircioğlu M. (2009). "Araç Rotalama Probleminin sezgisel bir yaklaşım ile çözümlenmesi üzerine bir uygulama", Doktora Tezi, Çukurova Üniversitesi, Adana, 22-72.
40. Laporte G. (1992). "The Vehicle Routing Problem: An overview of exact and approximate algorithms", *European Journal of Operational Research*, 59, 345-358.
41. Khan, M.S. ve Siddiqui, A.S., (1998). The Radius Method: A Modified Heuristic For The Vehicle Routing Problem, *Department of Industrial Engineering and Operations Research University of California Berkeley*.
42. Yang, X. S., and Deb, S., 2009, Cuckoo search via Lévy flights, 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), IEEE, 210-214.

43. Quaarab A, Ahiod B, Yang X.S., (1977). "Discrete Cuckoo Search Algorithm for the Traveling Salesman Problem". *Neural Computing & Applications*, 24, 1659-1669.
44. Fix, E., & Hodges, J. L. (1977). Discriminatory analysis, nonparametric discrimination: small sample performance. USAF School of Aviation Medicine Report, 1951; also In Agrawala, AK, Ed: Machine Recognition of patterns.
45. Akhun, M. (1992). Üretim Sistemlerinde Benzetim ve Uzman Sistemler, Endüstri Mühendisliği, İstanbul Teknik Üniveristesi, Benzetimde ilerlemeler-92, İstanbul.
46. James, M. (1984). Artificial Intelligence in BASIC, Butterworth&Co Ltd., Kent, England.
47. Genesereth, M.R. (1987). Logical Foundations of Artificial Intelligence, Los Altos, CA:Morgan Kaufmann Publishers.
48. Garnham, A. (1988). Artificial Intelligence, Routledge&Kagan Paul Ltd., London.
49. Özdemir A., Şeker, U. ve Aslan, E., (1992). Yapay Zekâ ve Uzman Sistemler, *Gazi Üniversitesi Teknik Eğitim Fakültesi Dergisi*, Ankara, 3, 3-4.
50. Erginbilgiç F. (1995). Artificial Intelligence Applications in Credit Analysis: An Overview with Four Examples, *METU Studies in Development*, 22 (2), 135-147.
51. Schildt, H. (1987). Advanced Turbo PROLOG, McGraw-Hill, Berkeley, California.
52. Turban, E. (1990). *Decision support and expert systems: Managerial perspectives*. Macmillan Library Reference.
53. Turban, E. (1992). Expert systems and applied artificial intelligence. *In Expert systems and applied artificial intelligence*, 804-804.
54. Yang, X. S. (2010). Nature-Inspired Metaheuristic Algorithms, Second Edition, *Luniver Press*.
55. Payne, R. B. (1977). The Ecology of Brood Parasitism in Birds, *Annual Review of Ecology and Systematics*, 8, 1–28.
56. Jourdain, F. C. R. (1925). A Study on Parasitism in the Cuckoos, *Proceedings of the Zoological Society of London*, 95, 639-667.
57. Viswanathan, G. M., Afanasyev, V., Buldyrev, S. V., Murphy, E. J., Prince, P. A. Ve Stanley, H. E., (1996). Lévy Flight Search Patterns of Wandering Albatrosses, *Nature*, 413-415.
58. Chechkin, A. V., Metzler, R., Klafter, J. ve Gonchar, V. Y., (2008). Introduction to the Theory of Lévy Flights, *Anomalous Transport: Foundations and Applications*, 129-162.
59. Mantegna, R. N. (1994). Fast, Accurate Algorithm for Numerical Simulation of Lévy Stable Stochastic Processes, *Physical Review E*, 4677-4683,
60. Nabiye V. V., (2003). *Yapay Zekâ*, Seçkin Yayınları, Ankara, 12-184.

61. İncekara H., (2010), *Tıbbi Tahlil Sonuçlarının Analizinde Web Ara Yüzlü Bulanık Uzman Sistem Tasarımı*, Yüksek Lisans Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, 11-36.
62. Çınar M. (2007). *Prostat Kanseri Riskinin Tesbitinde Sınıflandırıcı Tabanlı Uzman Sistem Kullanımı*, Yüksek Lisans Tezi, Ege Üniversitesi Fen Bilimleri Enstitüsü, İzmir, 18-23.
63. Harmon P., King D., (1985). *Expert Systems*, Wiley, New York.
64. Stirnen H. (1994). Otomasyonun Yazılım Yönü: Uzman Sistemler, *Otomasyon*, 108-116.
65. Allahverdi N.M., Yıldız S., Ünüvar A., (1995). *Endüstride Uzman Sistem Uygulamaları*, Otomasyon'95 Sempozyumu Bildiriler Kitabı, Mart sayısı, 75-86.
66. Bashlykov A. A., Eremecv A. P., (1994). Expert Systems for Support of Decision-Making in Energy, 216, Moscow.
67. Mishkoff H. C., (1986). Understanding Artificial Intelligence.
68. Popov E.V. (1987). Expert Systems: Solution of Non-formal Problems by Dialog with Computer, Nauka, 288, Moscow.
69. Aliev R.A., Abdikeev N.M., Shahnazarov M.M., (1990). Manufacturing Systems with Artificial Intellect, Radio i Svyaz, Moscow.
70. Allahverdi N. M., Huseynov B. A., Mustafayev R. T., (1995). Eski Sovyetlerde Uzman Sistemler, *Otomasyon*, 84-87.
71. Sındır K.O. ve Watt C., (1992). Uzman Sistemlerin Tarımda ve Tarımsal Mekanizasyon Alanlarda Uygulanabilirliği, Tarımsal Mekanizasyon 13. Ulusal Kongresi Bildiri Kitabı, Konya, 692-700.
72. Cebe B. (2012). *2d:4d Parmak Uzunlukları Oranı ile Kişilerin Sayısal-Sözel Dallara Yatkınlıkları Arasındaki İlişkinin Uzman Sistem ile Tespiti*, Yüksek Lisans Tezi, Fırat Üniversitesi Fen Bilimleri Enstitüsü, Elazığ, 28-45.
73. Taşgetiren M.F. (1996). Atelye Tipi Çizelgeleme Problemi İçin bir Uzman-Yapay Sinir Ağı Modeli, *İstanbul Üniversitesi.*, 34-88.
74. N. Allahverdi (2002). “*Uzman Sistemler: Bir Yapay Zekâ Uygulaması*”, Atlas Yayıncılık, 16-20, İstanbul.
75. Demirtaş, F., Zengin, K., Demirtaş, F. ve Zengin, K. (2016). Guguk Kuşu Algoritmasının Gezgin Satıcı Problemine Uygulanması ve Simülasyonu. *EEB 2016 Elektrik-Elektronik ve Bilgisayar Sempozyumu*, 396-399.
76. H. Başak, İ. Şahin, M. Gülen, (2008). “İnsansız hava aracı kazalarının önlenmesi için uzman sisteme dayalı risk yönetim modeli, *Teknoloji*, 11(3), 187-200.



ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : ÖNDEROĞLU, Mehmet Önder
Uyruğu : T.C.

Eğitim Derecesi	Okul/Program	Mezuniyet Yılı
Yüksek Lisans	Gazi Üniversitesi / Bilişim Enstitüsü Bilişim Sistemleri Ana bilim Dalı	Devam ediyor
Lisans	Kırıkkale Üniversitesi / Elektrik-Elektronik Mühendisliği	2010
Lise	Ankara Kocatepe Mimar Kemal Lisesi	2006

İş Deneyimi

Yıl	Yer	Görev
2011-Halen	Türk Telekom Ankara Bölge Müdürlüğü	Erişim Operasyon Müdürü

Yabancı Dil

İngilizce - Orta

Hobiler

Futbol oynamak, kitap okumak, seyahat etmek, film izlemek, masa tenisi oynamak



GAZİLİ OLMAK AYRICALIKTIR...

