



**T.C.
FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ**

**Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü**

Lisans Bitirme Projesi I

**MS. PACMAN OYUNU İÇİN AKILI
ALGORİTMALAR KULLANILARAK YAZILIM
KONTROLÖRÜ GELİŞTİRİLMESİ**

**Hasan GÜLBABA
1621221012**

**Enes Behlül YENİDÜNYA
1521221048**

Bitirme Projesi Danışmanı: Prof. Dr. Ayşe Şima UYAR

İstanbul, Kasım 2019



**T.C.
FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ**

**Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü**

Lisans Bitirme Projesi I

**MS. PACMAN OYUNU İÇİN AKILI
ALGORİTMALAR KULLANILARAK YAZILIM
KONTROLÖRÜ GELİŞTİRİLMESİ**

Hasan GÜLBABA
1621221012

Enes Behlül YENİDÜNYA
1521221048

Bitirme Projesi Danışmanı: Prof. Dr. Ayşe Şima UYAR

<u>Jüri Üyeleri:</u>	<u>İmza:</u>
Dr. Öğr. Üyesi	
Dr. Öğr. Üyesi	
Dr. Öğr. Üyesi	

İstanbul, Kasım 2019

İÇİNDEKİLER

İÇİNDEKİLER	ii
1. GİRİŞ	2
1.1 Konunun Tanımı.....	2
1.2 Ms. Pacman	2
1.3 Ms. Pac-Man Yarışması	4
2. TEMEL KAVRAMLAR	6
2.1 Projenin Amacı.....	6
2.2 İş Paketleri ve Gantt Diagramı	Error! Bookmark not defined.
3. ARAŞTIRMA VE TEORİK BİLGİLER	8
3.1 Ms. Pac-Man İçin Genel Algoritma Taraması.....	8
3.1.1 Monte Carlo Search Tree(MCST).....	8
4. TASARIM, GERÇEKLEŞTİRME VE TEST	10
4.1 Ms. Pac-Man Yarışması İçin Sağlanan Sınıflar.....	10
KAYNAKÇA.....	13

Şekil Listesi

Şekil 1.2 Ms. Pac-Man Oyunu	3
Şekil 2.3 Gantt Diagramı	5
Şekil 3.1.1 Monte Carlo Arama Ağacı Algoritması.....	7

Tablo Listesi

Tablo 2.2 İş Paketleri	4
------------------------------	---

**MS. PACMAN OYUNU İÇİN AKILI
ALGORİTMALAR KULLANILARAK YAZILIM
KONTROLÖRÜ GELİŞTİRİLMESİ**

1. GİRİŞ

1.1 Konunun Tanımı

Tarihin başlangıcından beri canlılar o anki amaçlarını gerçekleştirebilmek için çeşitli algoritmalar kullanmışlardır. Teknolojinin gelişmesiyle de birçok araç gereç kullanarak bu amaçlarını daha kolay gerçekleştirebilir hale gelmişlerdir. Artan karmaşık problemlerimizi çözdürdüğümüz bilgisayarlar hayatımızın önemli bir parçası haline gelmiştir. Biyolojik ve doğal zekanın modellenmesiyle büyük başarılar elde edilmiştir ve gelişmeler akıllı sistemler olarak adlandırılmıştır.

Akıllı algoritmaların farklı kullanım alanları vardır, bu alanlardan birisi de bilgisayar oyunlarıdır. Son yıllarda üretilen oyunların grafik, animasyon ve ses özelliklerindeki ilerlemelere rağmen birçok bilgisayar oyununun temel işleyiş mantığı birbirine benzemektedir. Ancak akıllı algoritmalar sayesinde bu tekdüzelikten kurtulabilir ve oyun içinde kullanıcıya ya da bilgisayara karşı kendi kendine kararlar alabilecek yazılım kontrolörü tasarlanabilir.

Akıllı algoritma tekniklerinin uygulandığı en ünlü oyunlardan bir tanesi de Ms. Pac-man'dir. Ms. Pac-man bu kontrolörler sayesinde kullanıcı müdahalesi olmadan pilleri yiyerek ve hayaletlerden kaçarak hayatta kalabilir ve yüksek puanlara ulaşabilir.

1.2 Ms. Pacman

Ms. Pac-man, General Computer Corporation tarafından geliştirilmiş olup 1982 yılında Midway tarafından yayınlanmış bir oyundur. 1980 yılında Namco tarafından ilk defa geliştirilen Pacman'in değiştirilmiş halidir. KAYNAK EKLE



Şekil 1.2 Ms. Pac-Man Oyunu [1]

Oyuncu Ms. Pac-Man'i sonu olmayan bir labirent içinde yönlendirir. Labirent pillerle doludur ve 4 adet farklı renklerde hayalet içerir: Blinky, Pinky, Inky ve Sue. Ekranın sol tarafından sağ tarafına doğru bir geçit vardır. Köşelerde birer tane olmak üzere toplam 4 adet güç pili bulunmaktadır. Oyunun amacı pilleri ve rengi maviye dönmüş hayaletleri yiyerek toplanılabildiği kadar puan toplamaktır.

Oyuncu (Ms. Pac-Man) pilleri yiyerek ve hayaletlerden kaçınarak puan kazanır ve herhangi bir hayaletle teması halinde can kaybeder. Bir güç pili yemesi halinde, önceden tanımlanmış süre bitene kadar hayaletler yenilebilir duruma geçer, renkleri maviye dönüşür ve yenilmeleri halinde fazladan puan kazanır. Tüm piller yendiği takdirde yeni bir seviyeye geçilmiş olur, seviyeler arttıkça hız artar ve hayaletleri yiyebilme süresi azalır. Her 10 bin puanda bir yeni bir can kazanır, oyun başlangıcındaysa 3 adet cana sahiptir. Bir hap yemek Ms. Pac-Man'e 10 puan kazanır ve hayalet yemek 200 puan kazanır, ancak hayalet yemeye devam edilmesi durumunda bu puan sürekli ikiye katlanarak kazanılır. 4 hayaletin de aynı anda yenilmesi durumunda toplamda 3000 puan kazanılmış olur. KAYNAKÇA EKLE(wikipedia)

Pac-man oyunu ile ilgili düzenlenen ilk yarışmalarda görüntü işleme metodları kullanılmaktaydı. KAYNAKÇA EKLE Pac-man ile ilgili birçok alanda araştırmalar yapılmış ve hem pac-man hem de hayalet kontrolörleri için çeşitli çalışmalar yapılmıştır. Oyun için yapılan bu çalışmalar:

- Gallegher ve Ledwich pac-man oyununun basit bir versiyonu ile yaptıkları bir çalışma yalnızca bir hayalet ve bir güç pili içeriyordu.KAYNAKÇA
- Lucas[4], oyuna basit bir Evolution Algoritması kullanarak çeşitli yöntemler denemiş ve bu yöntemleri Sinirsel Ağ modellerini eğitmek amacıyla kullanmıştır.
- Robles ve Lucas, basit bir Tree-Search algoritması kullanarak oyun üzerinde çalışmışlardır.KAYNAKÇA
- Burrow ve Lucas Ms. Pacman oyununu oynamayı öğrenebilen iki farklı yaklaşım sunmuşlardır. İlk metot klasik makine öğrenmesi yöntemleri ile, diğer metot ise temporal difference learning tir. KAYNAKÇA
- Nguyen ve Thawonmas CEC 2011 Ms. Pac-Man vs Hayalet Takım Yarışması'na katılan ajanlarını sundular ve daha sonra kazandılar. Bu kontrolör için kullanılan ajanlar, Blinky için tamamen kural tabanlı bir yaklaşım kullanırken Pinky, Sue ve Inky'yi MCTS ile kontrol etmekte. KAYNAKÇA

1.3 Ms. Pac-Man Yarışması

Bu projemizde <http://www.pacmanvghosts.co.uk/> sitesinden sağlanan platformu kullanarak, Ms. Pac-Man ve hayaletler için kontrolörler geliştireceğiz.

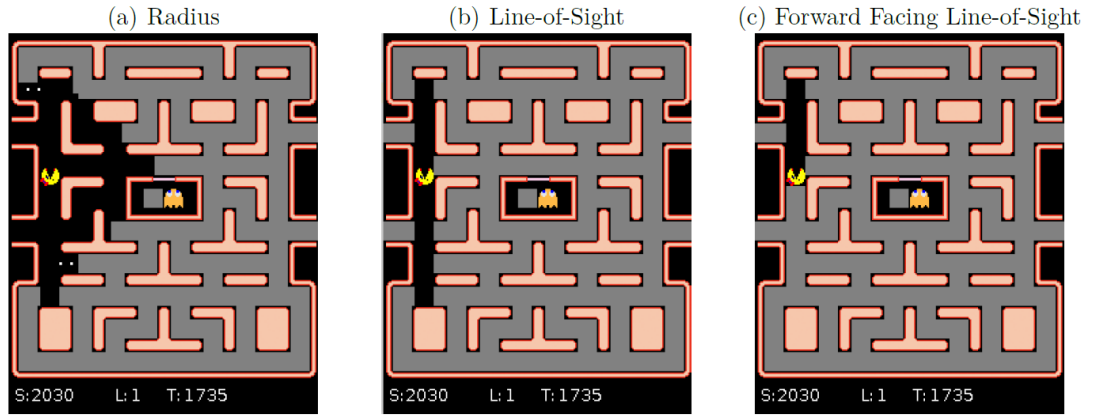
Yarışmada yalnızca Java dili kullanılmalı, performans arttırım amacıyla thread kullanılmamalıdır. Ayrıca Ms. Pac-Man'ın ve 4 adet hayaletin bir sonraki hamlesi için 40ms süreleri vardır. Bu sürenin tamamını Ms. Pac-Man tek başına kullanabilecekken, hayaletler kendi aralarında paylaşırlar.

Kullandığımız platformun bir özelliği de partially observable(PO) olmasıdır. PO, kontrolörün oyunu yalnızca Ms. Pac-Man'ın görüş alanı ile kısıtlı olarak yönetmesi durumudur. Bu durumda kontrolöre giden bilgiler yalnızca o anda oyuncunun

görebildiklerinden ibarettir. Örneğin hayaletlerden herhangi birinin konum bilgisi, oyuncu eğer o hayaleti görmekteyse kontrolöre iletilir, oyuncunun görüş alanı dışındaki bir pilin yenilip yenilmediği bilgisine erişilemez.

1.3.1 Partial Observability

Ms. Pac-Man motoru, her biri görüş limiti uygulanan üç PO türünü desteklemektedir. Bunlar şekil 1.3.1’de görüldüğü üzere Radius, Line-of-Sight(LOS) ve Forward Facing Line-of-Sight(FF-LOS)’dır.[5]



Şekil 1.3.1 PO görüş limiti türleri

- a) Yarıçap temelli PO: Görüş, Manhattan mesafesi veya Öklid mesafesi kullanılarak Bayan Pac-Man çevresindeki bir yarıçapla sınırlandırılabilir.[6]
- b) Görüş Çizgisi(LOS): Alternatif olarak, vizyon görüş çizgisiyle sınırlandırılabilir, yani Ms. Pac-Man koridorlardaki düz çizgilerde sınırlı bir mesafe görebilir ve duvarlar görüşünü engeller. Köşelerden hiçbir şey görülemez.[6]
- c) FF-LOS: Bu, MS. Pac-Man'in sadece seyahat ettiği yönde gözlemleyebileceği LOS üzerinde ek bir kısıtlamadır.

1.3.2 Oyun İçi Mesajlaşma

Oyunun PO olmasından dolayı hayaletler birbirleri arasında iletişime ihtiyaç duymaktadır. Ms. Pac-Man'ın görüldüğü son konum, güç piline olan mesafesi gibi önemli bilgiler hayaletler arası iletilir.

Mesaj tipi	Tanım
Pacman seen	Pacman in görüldüğünü diğer hayaletlere iletme mesajı
I am	Mesajı gönderenin nerede olduğunu bildiren mesaj
I am heading	Mesajı gönderenin nereye gittiğini bildiren mesaj

2. TEMEL KAVRAMLAR

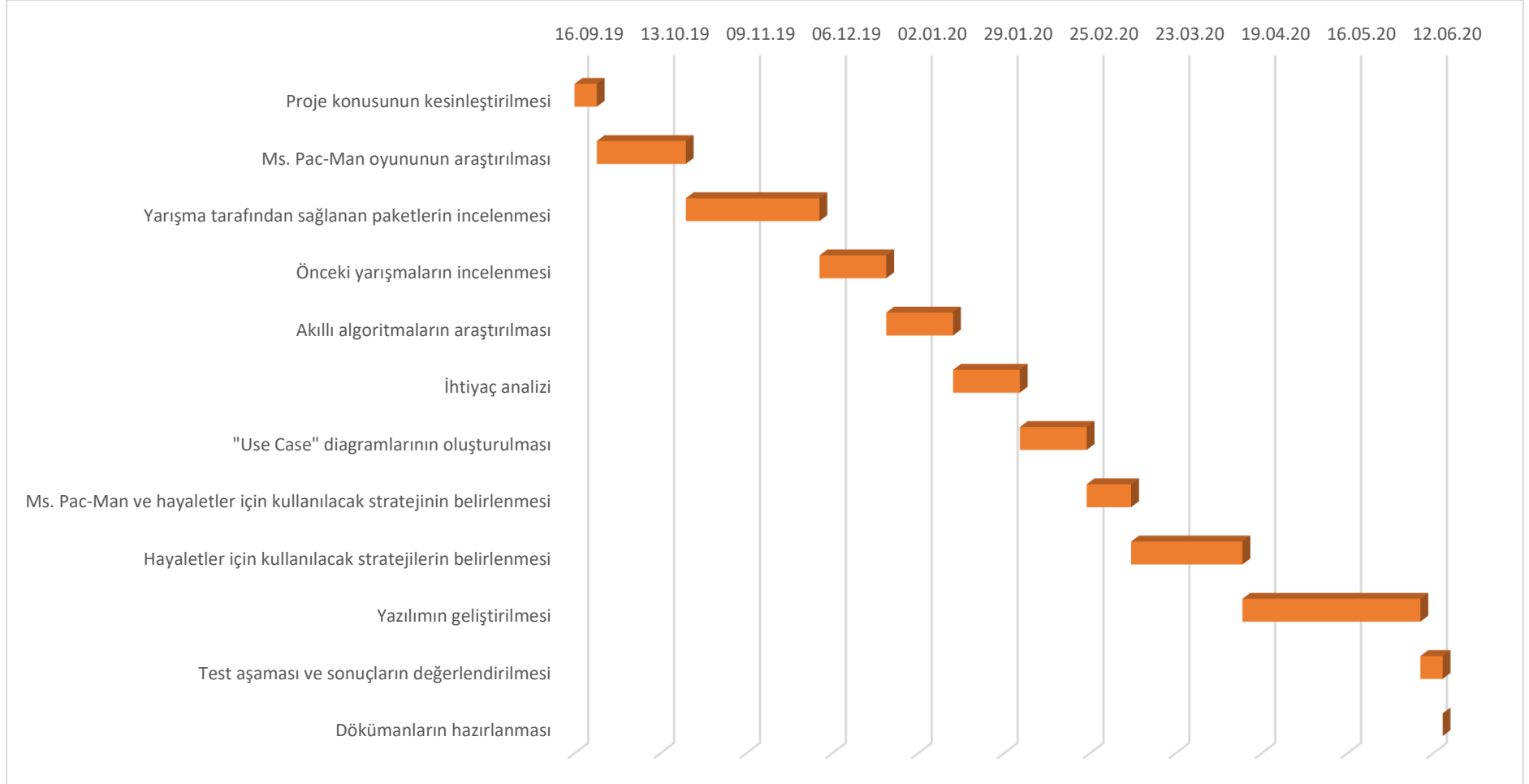
2.1 Projenin Amacı

Yapılan yarışmalarda amaç, Ms. Pac-man için en yüksek puanı toplaması, hayaletler içinse Ms. Pac-Man'ın en az puanla oyunu bitirmesidir.

2.3 İş Paketleri ve Gantt Diagramı

Görev	Başlangıç Tarihi	Süre (Gün)
Proje konusunun kesinleştirilmesi	16.09.19	7
Ms. Pac-Man oyununun araştırılması	23.09.19	21
Yarışma tarafından sağlanan paketlerin incelenmesi	14.10.19	42
Önceki yarışmaların incelenmesi	25.11.19	21
Akıllı algoritmaların araştırılması	16.12.19	21
İhtiyaç analizi	06.01.20	21
"Use Case" diagramlarının oluşturulması	27.01.20	21
Ms. Pac-Man ve hayaletler için kullanılacak stratejinin belirlenmesi	17.02.20	14
Hayaletler için kullanılacak stratejilerin belirlenmesi	02.03.20	35
Yazılımın geliştirilmesi	06.04.20	56
Test aşaması ve sonuçların değerlendirilmesi	01.06.20	14
Dökümanların hazırlanması	15.06.20	14

Tablo 2.2 İş Paketleri[1]



Şekil 2.2 Gantt Diagramı[2]

3. ARAŞTIRMA VE TEORİK BİLGİLER

3.1 Ms. Pac-Man İçin Genel Algoritma Taraması

Daha önce Ms Pacman için yapılan çözümlere bakıldığında, Kural Tabanlı Sistemler, Monte Carlo Search Tree, Evrimsel Algoritmalar, Yapay Sinir Ağları ve Takviyeli Öğrenme gibi yöntemlerin kullanıldığını görüyoruz. [1]

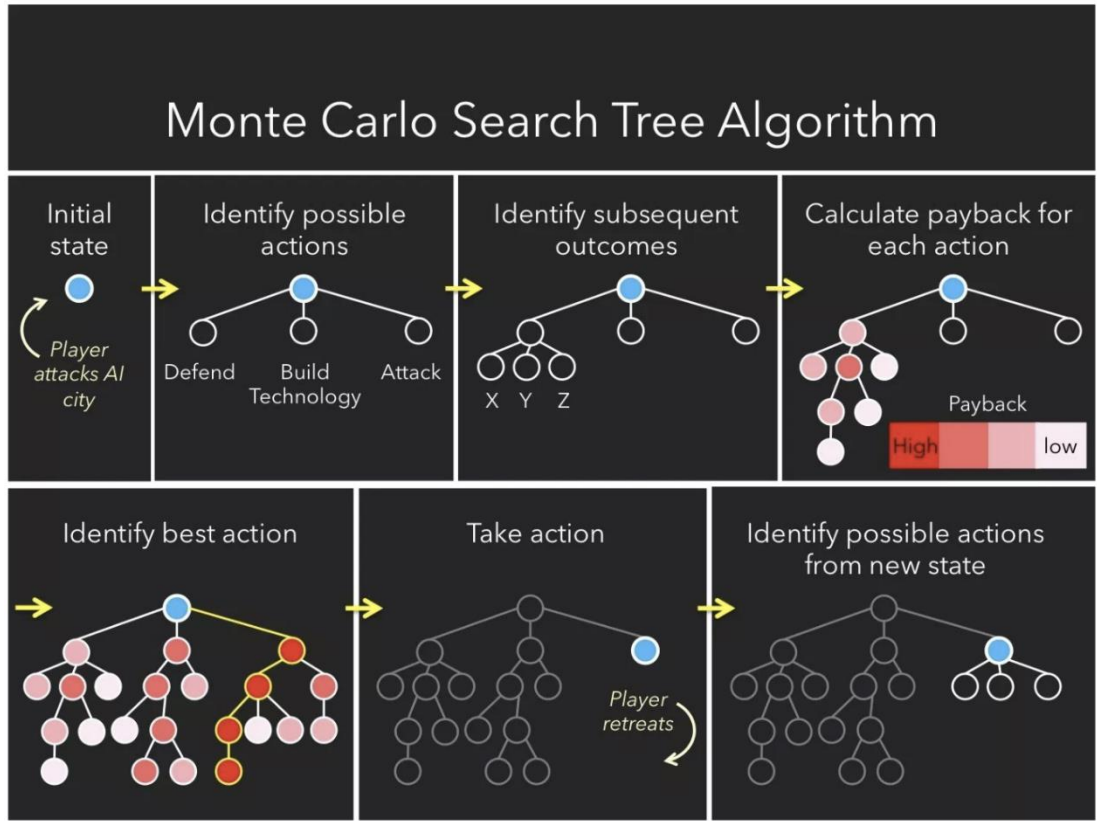
Bunlar arasında Monte Carlo Search Tree yüksek puanlara ulaşan bir yöntem olup dikkat çekmektedir.

3.1.1 Monte Carlo Search Tree(MCST)

Son yıllarda üretilen oyunların grafik, animasyon ve ses özelliklerindeki ilerlemelere rağmen birçok bilgisayar oyununun temel işleyiş mantığının birbirine benzemesi araştırmacıları MCST gibi farklı oyunlara uygulanabilen algoritmalara yönlendirmektedir.

Monte Carlo Ağaç Araması, bir oyunun anlık durumuna bakarak, bir sonraki duruma götürecek en iyi hamleyi bulmak için belirli sayıda rastgele tahminlerde bulunur. [2]

MCST'nin çalışma şekli, ilk önce bir botun halihazırda sahip olabileceği tüm olası hareketleri görselleştirmektir. Ardından, bu olası hamlelerin her biri için, bir oyuncunun yanıt verebileceği tüm hamleleri analiz eder, daha sonra da tepki olarak yapabileceği tüm olası tepki hamlelerini vb. dikkate alır. [3]



Şekil 3.1.1 Monte Carlo Arama Ağacı Algoritması [3]

Şekil 3.1.1, MCST kullanan bir bilgisayarın, bir insan bileşenine karşı harekete geçmeden önce geçtiği süreci vurgulamaktadır. Öncelikle sahip olduğu tüm seçeneklere bakıyor, yukarıdaki örnekte bu seçenekler ya savunmak, teknoloji geliştirmek ya da saldırı yapmak. Daha sonra, bundan sonraki her potansiyel hareket için başarı olasılığını öngören bir ağaç oluşturur. Yukarıda, başarı olasılığının en yüksek olduğu seçeneğin “saldırı” olduğunu görüyoruz (çünkü koyu kırmızı, daha yüksek bir ödül olasılığına eşittir), bu nedenle bilgisayar saldırımayı seçer. Oyuncu bir sonraki hamlesini yaptığında, bilgisayar tekrar ağaç yapım işlemini tekrarlar.

4. TASARIM, GERÇEKLEŞTİRME VE TEST

4.1 Ms. Pac-Man Yarışması İçin Sağlanan Sınıflar

Ms. Pac-Man oyunu, Game, Executor, MyPacman (Ms. Pac-Man) ve MASController (Hayaletler) ana sınıflarının ortaklaşa çağrılmasıyla çalışır. Main sınıfı içinde Executor nesnesi oluşturulur ve ihtiyaç duyulan nesneler ve metodlar bu sınıf vasıtasıyla çağrılır.

Executor sınıfı aracılığıyla bir adet Game nesnesi, hayaletler ve pacman için kontrolörler oluşturulur. 40 ms aralıklarla sürekli olarak, oyun, pacman ve hayaletler güncellenir.

Game sınıfı içinde, oyunun oynandığı Maze(labirent) nesnesi, piller, pillerin yenildiği bilgisi, skor, oyunun bitip bitmediği gibi bilgiler tutulur.

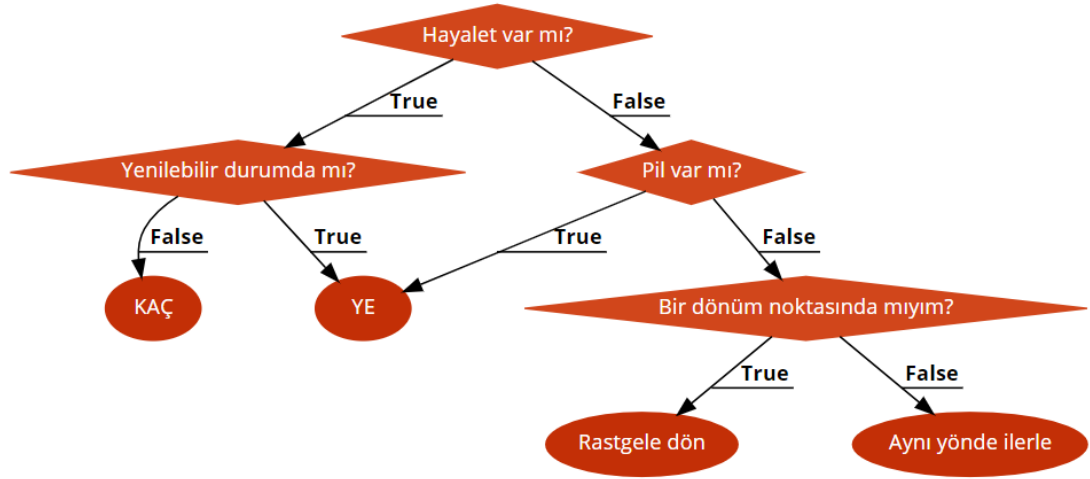
MyPacman sınıfı içinde, Ms. Pac-Man'ın hareket stratejisinin belirlenir. Başlangıçta bu sınıf pacman'i rastgele hareket ettirecek basit kodlardan oluşur ve yarışmacıların bu sınıfta yaptığı değişikliklerle daha yüksek puan alması beklenir.

POGhost sınıfı temel olarak hayaletlerin hangi durumlarda nasıl davranacağını belirlediği sınıftır. Hayaletlerin durumlara karşı verdiği tepkiler ve pac-man'ın durumuna karşı aldığı pozisyonlar için fonksiyonlar yer alır, "getApproximateNextMoveAwayFromTarget()" metodu ile pacman'den kaçma işlemini gerçekleştirir. Bu işlemi gerçekleştirirken pac-manın bulunduğu konuma bakar, pac-man powerPill'e yakınsa veya powerPill'i yediyse bu hayaletimiz için tehlike demektir ve kaçmaya başlaması komutu verilir.

Hayaletin en önemli işlevi pac-man'i kovalama ve onu yemektir. Bu işlemler yine bu sınıftaki metodlar ile gerçekleşir. "getApproximateNextMoveTowardsTarget()" metodu ile pac-man yenilebilir olduğu sürece (herhangi bir powerPill almamış ise) hayaletler pac-man'i kovalama görevini gerçekleştirir, eğer pac-man bir powerPill'e belli bir birimden daha fazla yakınsa hayaletler pac-man'i kovalamayı bırakır ve random bir yol seçerler.

POCommGhost sınıfının temel amacı ise hayaletlerin pac-man'i kovalarken birbirlerine haber verip pac-man'in konumu hakkında diğerlerini de haberdar etmektir. Hayaletlerin birbiri ile gerçekleştirdiği haberleşme işlemi yapay zekanın temellerinden olan sürü teorisi algoritmalarına (swarm-based algorithm) ve karınca teorisi algoritmalarına (ant colony algorithm) benzer olarak örnek gösterilebilir.

4.2 Ms. Pac-Man için kullandığımız yöntem



Şekil 4.2.1 Ms. Pac-Man algoritma akış diagramı

4.3 POCommGhosts algoritması

Algorithm 5 POCommGhosts basic algorithm

```
function GETMOVE()
    currentTick  $\leftarrow$  GETCURRENTTICK()
    if currentTick = 0 || currentTick - tickSeen
 $\geq$  TICKTHRESHOLD then
        lastPacmanIndex  $\leftarrow$  -1
        tickSeen  $\leftarrow$  -1
    end if
    pacman  $\leftarrow$  GETPACMANINDEX()
    messenger  $\leftarrow$  GETMESSENGER()
    if pacman  $\neq$  -1 then
        lastPacmanIndex  $\leftarrow$  pacman
        tickSeen  $\leftarrow$  currentTick
        if messenger  $\neq$  NULL then MESSENGER.ADDMESSAGE(...)
        end if
    end if
    if pacman = -1 AND messenger  $\neq$  NULL then
        for message in MESSENGER.GETMESSAGES(ghost) do
            if MESSAGE.GETTYPE() = PACMANSEEN then
                if MESSAGE.GETTICK() > tickSeen then
                    lastPacmanIndex  $\leftarrow$  MESSAGE.GETDATA()
                    tickSeen  $\leftarrow$  MESSAGE.GETTICK()
                end if
            end if
        end for
    end if
    if pacmanIndex = -1 then pacmanIndex  $\leftarrow$  lastPacmanIndex
end if
    if DOESREQUIREACTION() = False then return NULL end if
if
    pacman  $\leftarrow$  GETPACMANINDEX()
    if pacman  $\neq$  NULL then
        if ISEDIBLE() OR PACMANCLOSETOPPILL() then
            return NEXTMOVEAWAYFROM(pacman)
        end if
        if NEXTFLOAT < 0.9 then
            return NEXTMOVETOWARDS(pacman)
        end if
    else
        return NEXTRANDOMMOVE()
    end if
end function
```

KAYNAKÇA

- [1] Rohlfshagen Philipp, Liu Jialin, Perez-Liebana Diego, Lucas Simon M., “Pac-Man Conquers Academia: Two Decades of Research Using a Classic Arcade Game” (2017) s6.
- [2]<https://devhunteryz.wordpress.com/2018/03/30/monte-carlo-agac-aramasimonte-carlo-tree-search/>
- [3]<https://towardsdatascience.com/artificial-intelligence-in-video-games-3e2566d59c22>
- [4] S. M. Lucas, “Evolving a Neural Network Location Evaluator to Play Ms. Pac-Man,” in Computational Intelligence and Games. IEEE, 2005.
- [5] Price, W., & Schrum, J. (2019). Neuroevolution of Multimodal Ms. Pac-Man Controllers under Partially Observable Conditions. *2019 IEEE Congress on Evolutionary Computation, CEC 2019 - Proceedings*, 466–473. <https://doi.org/10.1109/CEC.2019.8790278>
- [6] Price, W., & Schrum, J. (2019). Neuroevolution of Multimodal Ms. Pac-Man Controllers under Partially Observable Conditions. *2019 IEEE Congress on Evolutionary Computation, CEC 2019 - Proceedings*, 466–473. <https://doi.org/10.1109/CEC.2019.8790278>

