



**Уральский  
федеральный  
университет**

РАЗРАБОТКА И ОБУЧЕНИЕ АГЕНТНО-  
ОРИЕНТИРОВАННОЙ МОДЕЛИ С  
ИСПОЛЬЗОВАНИЕМ ИГРОВОГО  
ДВИЖКА

---

Выпускная квалификационная работа

Руководитель: Аксенов Константин Александрович

Студент: Шакмаев Евгений Владимирович



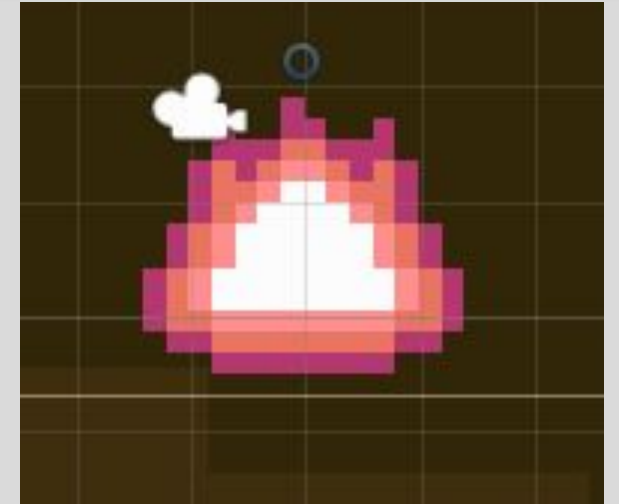
- Задача: Провести агентное моделирование в игровой среде
- Реализация: C#, Python
- Платформа: Unity, Anaconda



# Игровая среда



# Типы агентов



**Mule Follow (Script)**

Script	MuleFollow
Speed	20
Save Distance	10
Stopping Distance	20
Grabe Distance	10
Is Active	<input checked="" type="checkbox"/>

**Player Controller (Script)**

Script	PlayerController
Speed	20
Death Effect	Dead
Redemption Effect	Eye
Lerped Color	[Red bar]
Fixed Color	[Empty bar]

**The Determiner (Script)**

Brain	PlayerBrain (PlayerBrain)
Agent Cameras	<input type="button" value="Add Camera"/> <input type="button" value="Remove Camera"/>
Agent RenderTextures	<input type="button" value="Add RenderTextures"/> <input type="button" value="Remove RenderTextures"/>
Max Step	5000
Reset On Done	<input checked="" type="checkbox"/>
On Demand Decisions	<input type="checkbox"/>
Decision Interval	3
Script	TheDeterminer
Movement Speed	5
Rotation Speed	300
Rotation	0
Scene Controller	SceneController (SceneControlle)

# Машинное обучение

```
public override void AgentAction(float[] vectorAction, string textAction)
{
    AddReward(-1 / agentParameters.maxStep); //ревард при инициализации
    isCarryingDead = carriedDead != null;
    if (carriedDead != null)
        carriedDead.transform.position = transform.position;
    MovementControl(vectorAction);
}
```

```
if (other.CompareTag("Dead") || other.CompareTag("Eye"))
{
    carriedDead = other.gameObject;
    carriedDead.tag += "_carried";
    other.GetComponent<BoxCollider2D>().enabled = false;
    //ревард получения объекта
    AddReward(0.7f);
}
```

```
private void OnCollisionEnter2D(Collision2D other)
{
    //ударяется об стену
    SetReward(-1f); //понижает вес данного действия
    Done();
}
```

```
private void OnTriggerStay2D(Collider2D other)
{
    if (sceneController.AllDeadAreasFilled())
    {
        //ревард с добавление в бокс
        SetReward(2f);
        Done();
    }
}
```

```
if (
    (other.CompareTag("DeadArea") && carriedDead.CompareTag("Dead_carried"))
    || (other.CompareTag("EyeArea") && carriedDead.CompareTag("Eye_carried"))
)
{
    //ревард корректной отправки
    AddReward(1f);
    other.GetComponent<ObjectAreaController>().AddDead(carriedDead);
}
else
{
    //удаление отправки
    AddReward(-0.5f);
    Done();
}
```

# Сравнение агентов

Агент	Критерии				
	Интеракци я	Сложность	Состояния	Управление	Память
HeroDeath	2	2	-	декартовое	2,6 МБ
Medic	2	2	+	декартовое	2,6 МБ
Mule	4	4	-	декартовое	5,8 МБ
Enemy	3	3	+	декартовое	3,4 МБ
Player	6	неограниченная	+	декартовое	10,1 МБ
Agent	4	неограниченная	-	сферическое	12,1 МБ



Спасибо за внимание

---