遺伝的アルゴリズムによる適切なロボットの動きの模索プログラム

　　　　　　　　　　　　　　　　　　　　　　　　　　　　　制作者氏名

ファイル構成(.metaファイルとUnityプロジェクトのデフォルトなどは一部省略)

GARobot : Unityプロジェクトファイル

　Assets : アセットファイル、編集したプログラム、データ格納

　　Resources : UnityのPrefab格納

　　　Capsule.prefab : 足一本のプレハブ

　　　Goishi.prefab : 胴体のプレハブ

　　　Robot.prefab : マシン全体のプレハブ

　　Scenes : シーン格納

　　　EvolvingGoishiRobo.unity : 今回用いたプロジェクトシーン

　　Script : スクリプト格納、コンポーネントとして付与　中身は後述

　　　Main.cs : メイン関数、開始時に一度呼び出し　シーン内のカメラに追加

　　　Legspin.cs : モーター制御関数、足に付与する　Robot.prefabの各足に追加

　　　Info.cs : 情報表示用の関数　シーン内のテキストに追加

　　bounding.physicMaterial : 反発係数や摩擦係数のパラメータ

　　floor.mat : 床のマテリアル

　　tape.mat : スタートテープのマテリアル

シーン構成

EvolvingGoishiRobo

　Main Camera : カメラ、Position X=30,Y=3 Rotation Y=90 Main.csを付与

　Directional Light : ライト

　Plane : Plane床、Position X=3000 Rotation Xを変更で坂に Scale X=1000,Z=30 メッシュを付与

　Tape : Cubeスタート位置テープ、Position X=3000 Scale X=7500,Y=0.01

　Canvas

　　Text : テキスト、left top, Pos X=100,Y=-50 Width=180,Height=100

　EventSystem

Robot.prefabのGoishiにはSphere Mesh, Leg1~4にはCapsule Mesh, Capsle Collider, Hinge Joint(use motor, Force=infinity), Legspin.cs を付与

いずれもRigit Bodyを付与、これと床にはboundingを付与

以下UnityのC#スクリプト

##########

Main.cs

##########

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

public class Main : MonoBehaviour

{

// Start is called before the first frame update

public GameObject[] obj;

public GameObject robot;

public static int N =200;//number of machines per generation, lessthan 800

public static int genum=4;//the length of gene

public static int stage = 2;//power stage of machine motor, not include 0

//example: 2 -> -2, -1, 0, 1, 2

public static int elitenum=10;//num of elite, <N, more than 3

public static int remain = 0;//num of continueing elite robot, <=elitenum

public static float mutarate = 0.1f;//mutation rate, !!! apply to each gene !!!

public static float mutarate\_large = 0.8f;//large mutation rate

public static int mutagene\_large = 20;//1 large mutation per n generation

public static int genetypes = 4;//4 legs have...each gene -> 4 , same gene -> 1

public static int[,,] mastercode = new int[N, genetypes, genum];//all codes

public static int[,] bestcode = new int[genetypes, genum];

public int bestscore = 0;

public static int[,,] elitecode = new int[elitenum, genetypes, genum];//codes of elite

public static int limit = 10;//timelimit(sec)

public static int frame = 0;//60fps

public static int generation = 0;//sedai

public static float[] result = new float[N];//score

public static int[] soeji = new int[N];

string str;

public void Start()

{

Application.targetFrameRate = 60;//fps

obj = new GameObject[N];//objects

robot = (GameObject)Resources.Load("Robot");//Prefab of robot

for (int i=0;i<N;i++)

{

for (int j=0;j<genetypes;j++)

{

for (int k=0;k<genum;k++)

{

mastercode[i,j,k]=Random.Range(0, stage\*2+1) -stage;//create N gene random

}

}

}

/\*

if use original code

edit mastercode here

\*/

for (int i = 0; i < N; i++)//create object

{

obj[i] = Instantiate(robot, new Vector3(10 \* i, 2, 0), Quaternion.identity /\*Quaternion.Euler(45, 0, 0)\*/);

for (int j = 0; j < genum; j++)

{

obj[i].transform.Find("Leg1").GetComponent<Legspin>().code[j] = mastercode[i, 0 % genetypes, j];

obj[i].transform.Find("Leg2").GetComponent<Legspin>().code[j] = mastercode[i, 1 % genetypes, j];

obj[i].transform.Find("Leg3").GetComponent<Legspin>().code[j] = mastercode[i, 2 % genetypes, j];

obj[i].transform.Find("Leg4").GetComponent<Legspin>().code[j] = mastercode[i, 3 % genetypes, j];

obj[i].transform.Find("Leg1").GetComponent<Legspin>().oya = obj[i];

obj[i].transform.Find("Leg2").GetComponent<Legspin>().oya = obj[i];

obj[i].transform.Find("Leg3").GetComponent<Legspin>().oya = obj[i];

obj[i].transform.Find("Leg4").GetComponent<Legspin>().oya = obj[i];

obj[i].transform.Find("Leg1").GetComponent<Legspin>().machinenum = i;

obj[i].transform.Find("Leg2").GetComponent<Legspin>().machinenum = i;

obj[i].transform.Find("Leg3").GetComponent<Legspin>().machinenum = i;

obj[i].transform.Find("Leg4").GetComponent<Legspin>().machinenum = i;

}

}

}

// Update is called once per frame

void Update()

{

if (frame==(limit\*60))//time up

{

Evolve(frame);

frame = 0;

}

else//count up

{

frame++;

}

}

void Evolve(int checkfirst)//function to connect the next generation

{

/\*

Get score & Select gene

\*/

for (int i = 0; i < N; i++)

{

result[i] = obj[i].transform.Find("Goishi").transform.position.z;//get result

}

for (int i = 0; i < N; i++)

{

soeji[i] = i;//for sort

}

System.Array.Sort(result, soeji);//sort

System.Array.Reverse(result);

System.Array.Reverse(soeji);

Debug.Log(generation+"gen:"+soeji[0].ToString("D3") + ": " + result[0]);//display the best score of the generation

if (result[0] > bestscore)//update the best score of the simuration

{

for (int j = 0; j < genetypes; j++)

{

for (int k=0;k<genum;k++)

{

bestcode[j, k] = mastercode[soeji[0],j,k];

}

}

}

if (generation%50==0 && generation>0)//option for output

{

for (int j = 0; j < genetypes; j++)

{

str = j+": ";

for (int k = 0; k < genum; k++)

{

//str +=System.String.Format("{0,2}",mastercode[soeji[0],j,k]);

str += ", " + bestcode[j,k];

}

Debug.Log(str);

}

}

for (int i=0;i<elitenum;i++)//record elite

{

for (int j=0;j<genetypes;j++)

{

for (int k=0;k<genum;k++)

{

elitecode[i, j, k] = mastercode[soeji[i],j,k];

}

}

}

/\*

Crossing & Create children

\*/

for (int i=0;i<remain;i++)//remain & continue

{

for (int j=0;j<genetypes;j++)

{

for (int k=0;k<genum;k++)

{

mastercode[i, j, k] = elitecode[i, j, k];

}

}

}

for (int i = remain; i < N; i++)

{

for (int j = 0; j < genetypes; j++)

{

for (int k = 0; k < genum; k++)

{

if (((generation+1)%mutagene\_large==0 && Random.Range(0.0f, 1.0f) < mutarate\_large) || ((generation + 1) % mutagene\_large != 0 && Random.Range(0.0f, 1.0f) < mutarate))

{//if next generation have large mutation, mutarate->mutarate\_large

mastercode[i, j, k] = Random.Range(0, stage\*2+1) - stage;

}

else

{

mastercode[i, j, k] = elitecode[System.Math.Min(Random.Range(0, elitenum-1), Random.Range(0, elitenum-1)), j, k];

}

}

}

}

for (int i = 0; i < N; i++)//destroy object

{

Destroy(obj[i]);

}

for (int i = 0; i < N; i++)

{

obj[i] = Instantiate(robot, new Vector3(10 \* i, 2, 0), Quaternion.identity /\*Quaternion.Euler(45, 0, 0)\*/);

for (int j = 0; j < genum; j++)

{

obj[i].transform.Find("Leg1").GetComponent<Legspin>().code[j] = mastercode[i, 0 % genetypes, j];

obj[i].transform.Find("Leg2").GetComponent<Legspin>().code[j] = mastercode[i, 1 % genetypes, j];

obj[i].transform.Find("Leg3").GetComponent<Legspin>().code[j] = mastercode[i, 2 % genetypes, j];

obj[i].transform.Find("Leg4").GetComponent<Legspin>().code[j] = mastercode[i, 3 % genetypes, j];

obj[i].transform.Find("Leg1").GetComponent<Legspin>().oya = obj[i];

obj[i].transform.Find("Leg2").GetComponent<Legspin>().oya = obj[i];

obj[i].transform.Find("Leg3").GetComponent<Legspin>().oya = obj[i];

obj[i].transform.Find("Leg4").GetComponent<Legspin>().oya = obj[i];

obj[i].transform.Find("Leg1").GetComponent<Legspin>().machinenum = i;

obj[i].transform.Find("Leg2").GetComponent<Legspin>().machinenum = i;

obj[i].transform.Find("Leg3").GetComponent<Legspin>().machinenum = i;

obj[i].transform.Find("Leg4").GetComponent<Legspin>().machinenum = i;

/\*format leg position -> 0 degree --- without this, X axis plus direction -> 0 degree

obj[i].transform.Find("Leg1").GetComponent<Legspin>().deg = 0;

obj[i].transform.Find("Leg2").GetComponent<Legspin>().deg = 90;

obj[i].transform.Find("Leg3").GetComponent<Legspin>().deg = 180;

obj[i].transform.Find("Leg4").GetComponent<Legspin>().deg = 270;

\*/

}

}

generation++;

}

}

##########

Legspin.cs

##########

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Legspin : MonoBehaviour

{

// Start is called before the first frame update

int frame = 0;

float x = 0, z = 0, y = 0, r = 0;

public int[] code = new int[Main.genum];

HingeJoint joint;//hinge joint

JointMotor motor;//1 motor per 1 leg

public int mode = 0;//state: body -> leg direction

public int machinenum = 0;

public float deg = 0;//option

public GameObject oya;

void Start()

{

oya = new GameObject();

joint = GetComponent<HingeJoint>();

motor = joint.motor;

motor.freeSpin = false;//use force to stop the motor

motor.targetVelocity = 0;

//motor.force = 10000000;//power of motor, default is infinity(3.4e+38)

joint.motor = motor;

}

// Update is called once per frame

void Update()

{

joint.motor = motor;

if (frame%6==0)//change motor order 60/n times per second

{

x = transform.position.x-oya.transform.position.x - 10 \* machinenum;

z = transform.position.z - oya.transform.position.z;

y = transform.position.y;

r = Mathf.Atan2(x,z)\*Mathf.Rad2Deg;//body -> leg direction

r += deg-(180/Main.genum);

if (r<0)

{

r += 360;

}

r = r % 360;

mode = (int)Mathf.Floor(r/(360/Main.genum));

motor.targetVelocity = code[mode%Main.genum] \* 720;

//motor.targetVelocity = code[(frame / 20)%30] \* 180;//move by frame

//motor.targetVelocity = (Random.Range(0, 5) - 2) \* 180;

}

/\*

if (frame%180==0)

{

Debug.Log(machinenum+": "+x+" "+z);

}

\*/

frame++;

}

}

##########

Info.cs

##########

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

public class Info : MonoBehaviour

{

// Start is called before the first frame update

public static Text InfoText;

int frame = 0;

string score = "";

string[] str = new string[Main.genetypes];

void Start()

{

InfoText = GetComponent<Text>();

InfoText.text = "success";

}

// Update is called once per frame

void Update()

{

InfoText.text = "第" + Main.generation + "世代 " + (600-frame) + "\n" + "score: " + score + "\n";

for (int j=0; j<Main.genetypes; j++)

{

InfoText.text += str[j]+"\n";

}

if (frame==Main.limit\*60)

{

frame = 0;

}

else

{

if (frame==1)

{

score = Main.result[0].ToString("F3");

for (int j = 0; j < Main.genetypes; j++)

{

str[j] = j + ": ";

for (int k = 0; k < Main.genum; k++)

{

if (Main.elitecode[0, j, k]>=0)

{

str[j] += " ";

}

str[j] += Main.elitecode[0, j, k];

if (k + 1 < Main.genum)

{

str[j] += ",";

}

}

}

}

frame++;

}

}

}