

# Pacman Final Project

Xiaodi

November 22, 2010

## 1 Game rules updated

In the new world of pacmen. We have the following changes:

**Digestion** : When eating a pacdot, the pacman stops for a short time  $T_{digest}$  to digest the dot. Therefore, if your pacman needs to go to somewhere as quick as possible (such as being chased by a super-pacman, or trying to eat the fruit), you may consider a route without pacdots, instead of being slowed down by pacdots.

**Fruit** : A fruit converts a pacman into a super-pacman for a short period of time. The growth of the fruit take a constant time  $T_{fruit}$ . The fruit enters the maze from one portal, and exits at the corresponding portal. It always take the shortest path – in other words, the behavior of a fruit is fully predictable. The moving speed of the fruit is 1 (A normal pacman's moves at 2).

**Super-pacman** : Infused with the magic power of the fruit, a super-pacman has many abilities: first, it always travels in full-speed, even when eating pacdots. More importantly, it can stun an opponent pacman for  $T_{stun}$  when they meet. Finally, like many other games, a pacman in this super state has a blinking appearance. The superman state only lasts a short period  $T_{superman}$ .

## 2 Events and simultaneous events

Now we give a formal definition of events and simultaneous events.

Let  $\mathbf{x}_i, \mathbf{x}_j$  be the center points of pacman-i and pacman-j. The center of an object is  $\mathbf{o}$ . We say pacman-i touches the object, if:

$$\|\mathbf{x}_i - \mathbf{o}\|_{\infty} < r_{inner}, \quad (2.1)$$

where  $\|\cdot\|_{\infty}$  takes the max distance of x- and y- coordinates.

Given that pacman-i touches an object, we can also define a simultaneous event if:

$$\|\mathbf{x}_j - \mathbf{o}\|_{\infty} < r_{outer} \quad (2.2)$$

If two pacmen eats the same pacdot *simultaneously*, they split the score. If two pacmen eats the same fruit *simultaneously*, the fruit is nullified, where no one becomes a superman.

### 3 Data structure

Since the world of pacman is becoming continuous, we will give the x- y-coordinate of an object, instead of their nearest column/row.

The initialize function is very similar to the milestone project:

```
team = teamAI(map, selfPos, rivalPos, mapHeight, mapWidth)
```

During the game, the `think` function is called in **every frame**. This means that you're observing the world in a continuous way. For example, this design gives your pacman the ability to "run away" from your opponent immediately after him eating the fruit. However, you have to design your algorithm more carefully to save your grace time.

Once the think function is called, it receives the following input:

```
team.think(map, selfStat, rivalStat, fruitPos, graceTime)
```

`selfStat` contains the location of your own position, as well as the remaining timers:  $t_{superman}$ ,  $t_{digest}$ ,  $t_{stun}$ , and your current score. `rivalStat` contains the same information for your rival. `fruitPos` is the position of the fruit. If the fruit is not available, it will be positioned at  $(-16, -16)$ .

Each team has to write its own classes (exactly the same as milestone). It is not allowed to call any internal variable or functions. The max initialization time is 30 second, with the maximum usage of memory 1GB.

Also, we encourage teams to utilize the `graceTime` variable. In milestone, most of the winning teams have used up 4000ms or more of their grace time.

### 4 Parameters

1.  $T_{digest}$  is a random integer drawn uniformly from  $[0, 10]$ .
2.  $T_{fruit} = 800$
3.  $T_{stun} = 200$
4.  $T_{superman} = 400$
5.  $r_{inner} = 6$
6.  $r_{outer} = 8$

### 5 Scoring

Every two pacman will be matched twice. For each match, winning=3/tie=1/loss=0. The total score for one team is the sum of all 30 match scores (In 15 of the matches you're playing team1, and other 15 matches team2).