

# CS3210 Assignment2 Report

---

A0091747W Qing Cheng

## Pseudo-code:

### Training session:

```
Foreach round
    If round == 0:
        Initialize();
    If rank == field:
        Broadcast(ball_location);
        reachable_players = Gather(players);
        Winner = decide_winner(reachable_players);
        Scatter(winner);
        If(winner):
            ball_location = gather(winner);
        Players_info = gather(players);
        Print(players_info);
    If rank == player:
        Ball_location = Broadcast(field);
        Reachability = compute_rechability(ball_location, my_location,
movement);
        If reachability:
            Gather(field, reachability);
            Winner = scatter(field);
            If winner == rank:
                Scatter(field, New_ball_location);
```

```

        Else:// I'M NOT WINNER

            Scatter(winner);

    Else:

        Gather(field, reachability);

        Winner = scatter(field);

        If winner:

            Scatter(winner);

    Gather(field, my_info);

```

## Match:

```

Foreach round:

    If is_new_game || round == num_of_rounds_per_half:

        If round == num_of_rounds_per_half:

            Current_half = 1;

            Initialize();

            Is_new_game = false;

    If rank == field0 || rank == field1:

        Foreach player in its_area:

            Send(player, ball_location);

        If field.has_ball:

            Gather(players, reachability);

            reachable_players = Gather(players);

            Winner = decide_winner(reachable_players);

            Scatter(winner);

        Else:

            Gather(other_field, dummy_value);

            Winner = scatter(other_field);

```

If winner:

New\_location, shooting\_skill = Recv(winner);

If new\_location != dummy\_value:

Shot = determine\_shoot(shooting\_skill);

If shot:

If new\_location == basket\_location:

Add\_score(new\_location);

Is\_new\_game = true;

Else://it's not a goal

If is\_outside\_field(new\_location);

Is\_new\_game = true;

Else://ball didn't get the target

New\_location =

determine\_actual\_location(new\_location);

If is\_outside\_field(new\_location):

Is\_new\_game = true;

Send(another\_field, field\_info);

Else://target position is not on my field

Field\_info = Recv(another\_field)

Update\_my\_info(field\_info);

If rank == field0:

My\_players\_info = Gather(my\_players);

Other\_players\_info = recv(field1);

Print(my\_players\_info + other\_players\_info);

Scatter(is\_new\_game);

Else://field1

```

        My_players_info = Gather(my_players);

        recv(field0,my_players_info);

    else://I am a player

        Ball_location = recv(my_field);

        Reachability = compute_rechability(ball_location,my_location,
my_speed);

        If reachability:

            My_new_position = ball_position;

            send(ball_field, reachability, challenge);

            Winner = scatter(field);

            If winner == rank:

                Distance_to_basket = get_distance(ball_location,
basket_location);

                If distance_to_basket <= my_shoot_threshold:

                    New_ball_location = basket_location;

                    Else://don't shoot to the basket, just towards it for a
distance of my_shoot_threshold

                        New_ball_location =
determine_shoot_location(ball_location, my_shoot_threshold);

                        send(new_ball_location_field, New_ball_location);

                        send(other_field, dummy_value);

            Else://I cannot reach

                Gather(field, reachability, dummy_value);

                My_new_position = Move_towards_ball();

                Winner = scatter(field);

            If my_new_position_field == field0:

```

```

    Gather(field0, my_info);

    Gather(field1, dummy_value);

Else://i'll be in field1

    Gather(field0, dummy_value);

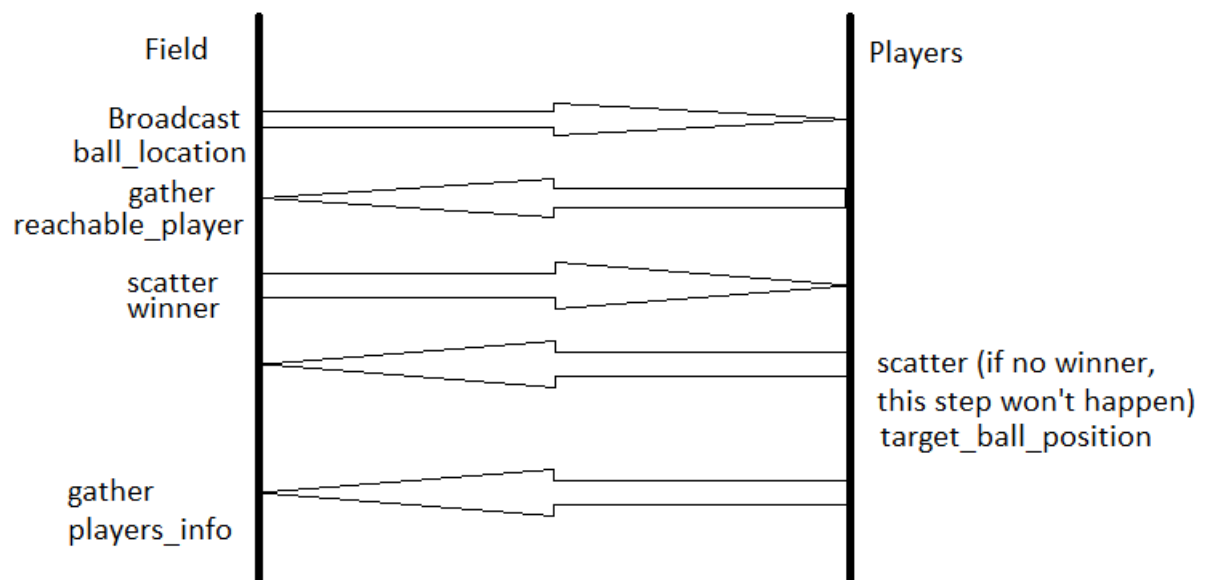
    Gather(field1, my_info);

Is_new_game = scatter(field0);

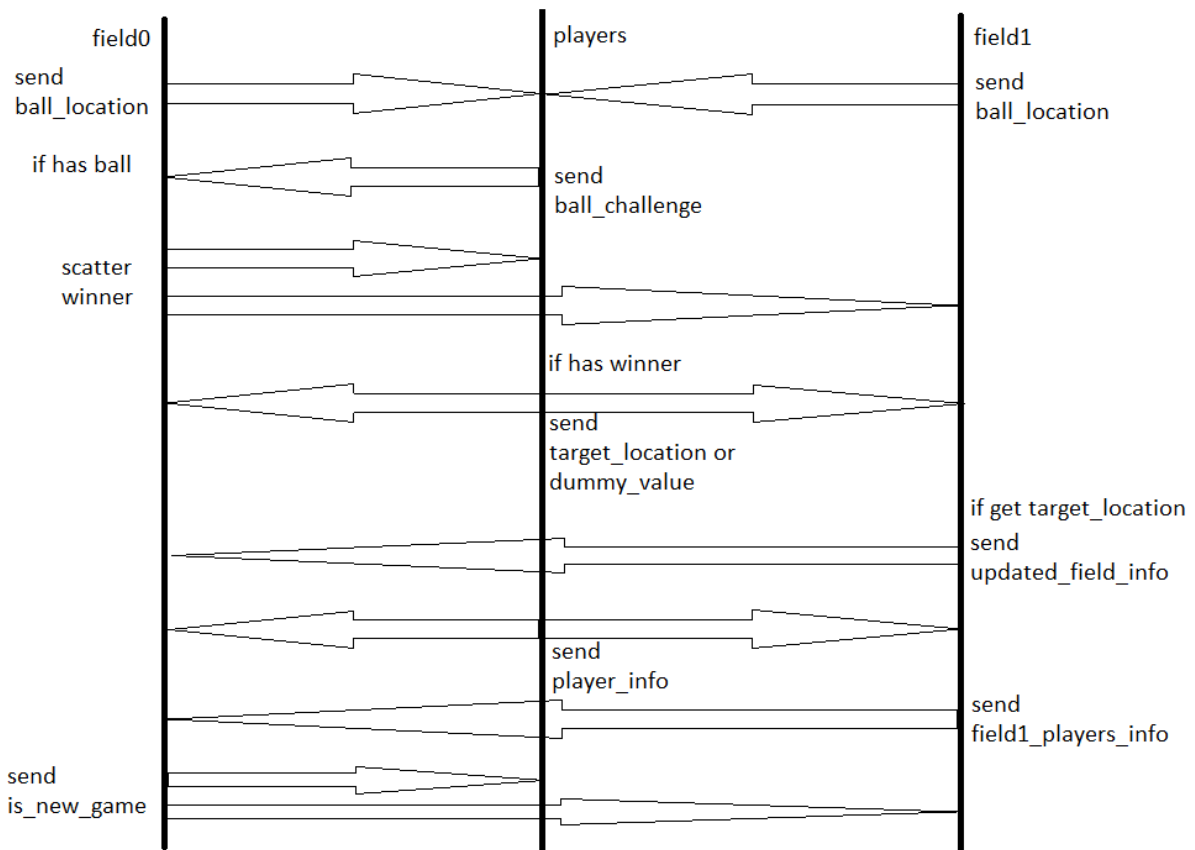
```

## Diagrams:

Training:



Match:



## Design:

I used only collective communication in training session and both collective and point to point communication in match. And I only used blocking communication. My design of match:

The ability distributions are the same among the players in the same team. We can adjust the ability distribution for each team in the constants. If a player can reach a ball, he will go to the ball. If he gets the ball, he will choose to either shoot it to the basket or throw it towards his attack direction, this is determined by the relation between the distance between the ball and the basket and the team's `shoot_threshold`. If the player cannot reach the ball, he will move towards the ball.

In the initialization, ball is put at the center of the field and the players are located randomly on the field that they are defending.

There are three situations where the game is initialized(except for some special attributes like scores, etc):

1. A goal happens
2. The ball is beyond the boundary
3. The match enters the second half

Some findings:

1. The speed has the most influence on the total scores. The smaller the speed each team has, the less scores each team gains.
2. Dribbling skill has the second most influence on the total scores.
3. Shooting skill doesn't seem to matter a lot.

### **Instructions on compiling and running**

Training:

To compile:

```
mpicc training.c
```

To run:

```
mpirun -np 6 a.out
```

Match:

To compile:

```
mpicc match.c -lrt
```

To run:

```
mpirun -np 12 a.out
```

### **Average time per round**

After comment out all the printing, the total running time is 1.54s,  
which gives average time per round = 0.0002655 second