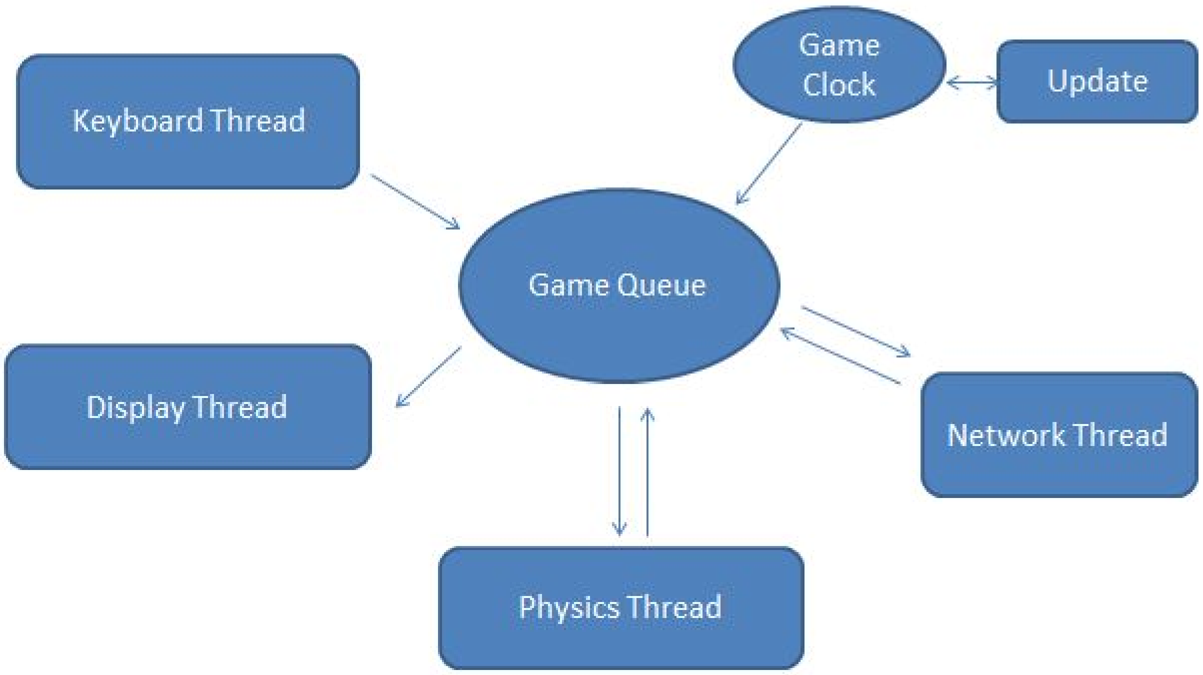
Multiplayer Carrom Game

Satyender Yadav & Shivam Handa

October 21, 2011



The High level Abstraction Structure of the Carrom Game

1 Game Engine Architecture

This is the highest level of Abstraction.

1.1 Members

1.1.1 Threads

Physics Thread
Network Thread
Display Thread
Keyboard Thread
Update

1.1.2 Shared Variables

Game Queue
Game Clock

1.2 Working

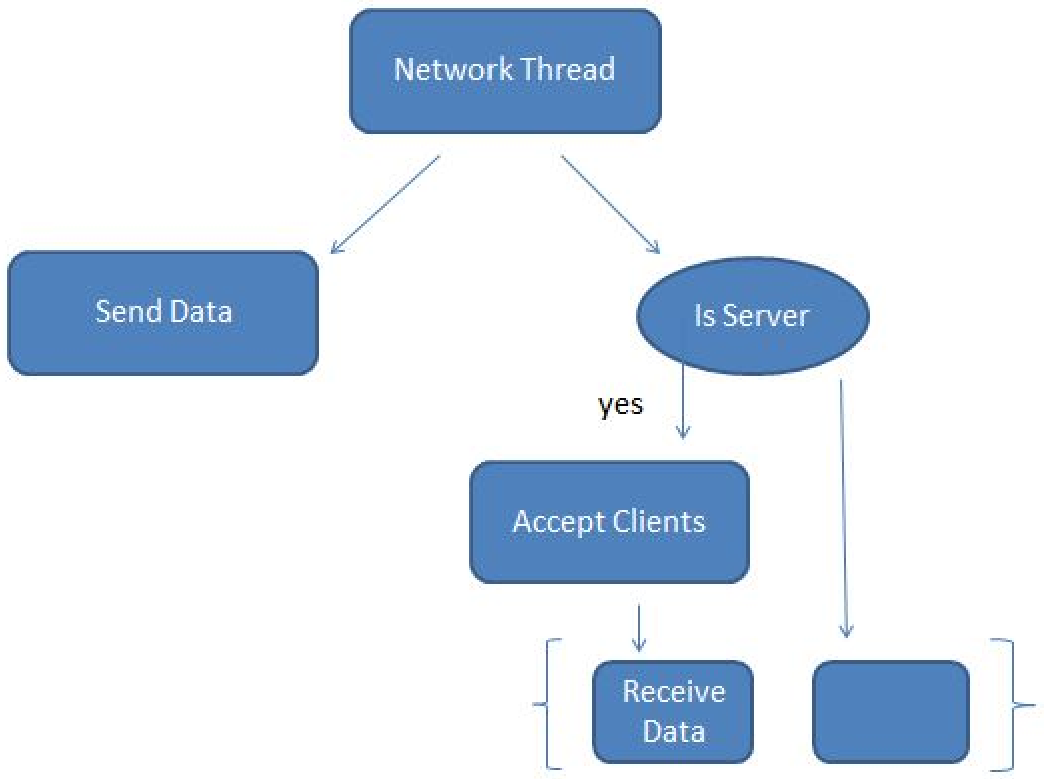
The State of the Game is defined by the Game Queue,Previous History and Game Clock. Threads can only communicate to the Game Queue thus for them other threads dont exist. Each thread can communicate to the Game Queue through a structure called an **Event**. The Event is the discriptor of the current change in state.The Synchronization of these threads is a Function of the game clock. At a given time the Game Queue only contains Events that will have to be executed in the given gameclock. The Update thread will start its work only when all the threads will let him do its work. It will also call the Postredraw Function of the will make the Display refresh the Screen. In Every Clock Cycle, the Physics thread will call queue in the series of position changes per Clock Cycle.

1.3 Structure : Events

label: (int) This is the discriptor of the events.
Time: (int) This is the way the Threads know which event should be scheduled to be executed and when.
Ball1: (int) This is a general int field used for int communication. In Physics Threads it gives the Ball number.
Ball2: (int) This is a general int field used for int communication. In Physics Threads used for Ball number(collision),wall number or pocketno.
Buffer1: (char []) This is used to transport data , in Physics threads it is used to communicate change state of balls.
Buffer2: (char []) This is used to transport data , in Physics threads it is used to communicate change state of balls.

1.4 Problems

Q1. How does Network Threads changes the State of the game?
Ans. The network threads are in constant lookup on all network channels for new Data.When the recieve data it converts into an event then enters it into a queue.
Q2. How do gameclock ensure that the Physics thread has done its job ?
Ans. The maintain a global Variable which the Physics declare done when it has done all its job for the given cycle.



Structure of the Network Thread

2 Networking

The Networking abstraction

2.1 Members

2.1.1 Threads

Send Data
Accept Clients

Recieve Data - multiple such threads , one for each clients

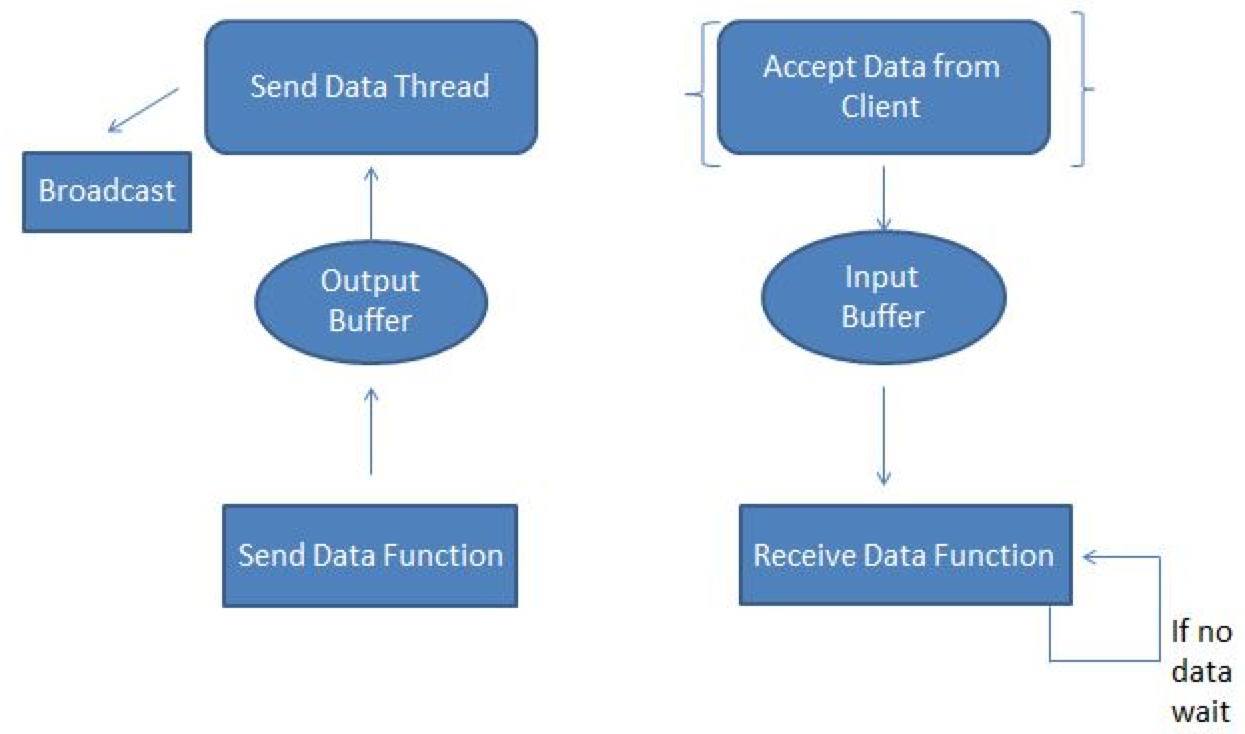
2.1.2 Shared Variable

Input Buffer
Output Buffer
newInput?
newOutput?

2.2 Working

Networking is fully abstracted in the sense of Data being sent. The server is a multiple Client strategy, all clients are attended in Parallel Threads and therefore information is not missed. The accept client threads will connect to a client , assign it a thread and an id within the sever database, then create another socket to communicate to the given client . and thus freeing the previos thread. The Send Data thread uses the id to communicate to the given clients, when the Broadcast mode is set on ,the clients will all recieve data by forking process and after data is sent by destroying processes.

2.3 Communication by outside world



The Functioning of the Network Thread

2.4 Problems

Q1. What does the Program do when there are a multiple clients trying to send data?
Ans. The Thread who gets access to the inputBuffer first will write the Data on it. The other Thread will have to wait till somefetches the data and tells teh program that the data has been recieved.
Q2.how do I control the disconnect of a client?
Ans. It is the function of the user i.e. the programing controlling the server to tell when someone disconnected. The network thread will not stall on a disconnected user.

3 Physics Thread

3.1 Members

3.1.1 Threads

Simulate Physics

3.1.2 Shared Variable

Physics Queue

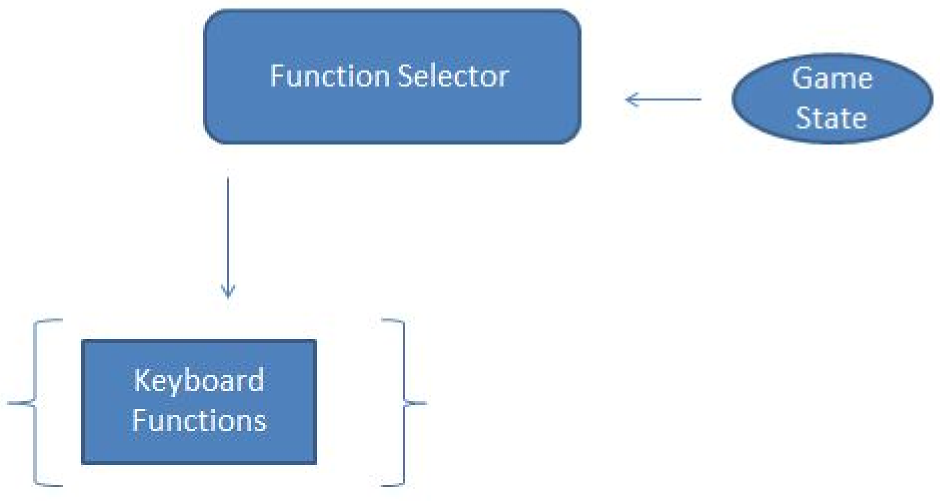
3.2 Working

The Thread will calculate all the data and insert in the Physics Queue. The outside world communicates only to this queue. It will dequeue all the eventsof ,that it uses. The thread also backsup the all its data. The outside can only communicate with the threads.

3.3 Problems

Q1. How do you control the fact that the data will flood the physics Queue?
Ans. When the Physics queue is Full the Physics thread will pause until there is someone who dequeues the events.
Q2. How do you check the fact all balls have Stoped?
Ans. It maintains a function checking that if all balls are moving or not.

4 Keyboard Thread



The input Layout

4.1 Members

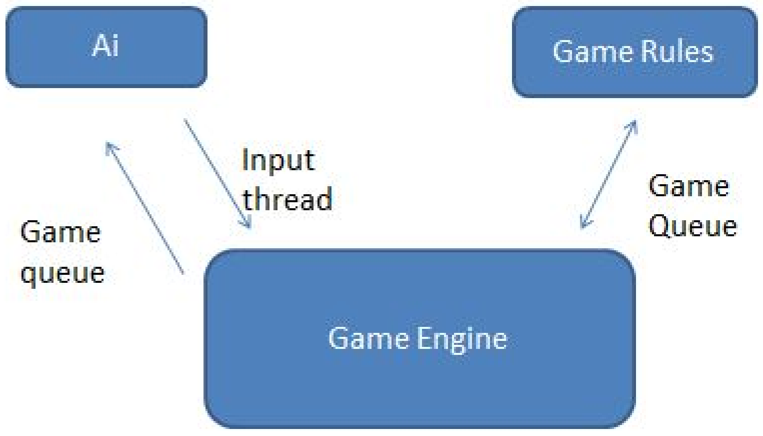
4.1.1 Threads

Select Function

4.2 Working

The Select Function work is will take the game state and using function pointers will do what is required.

5 The Game

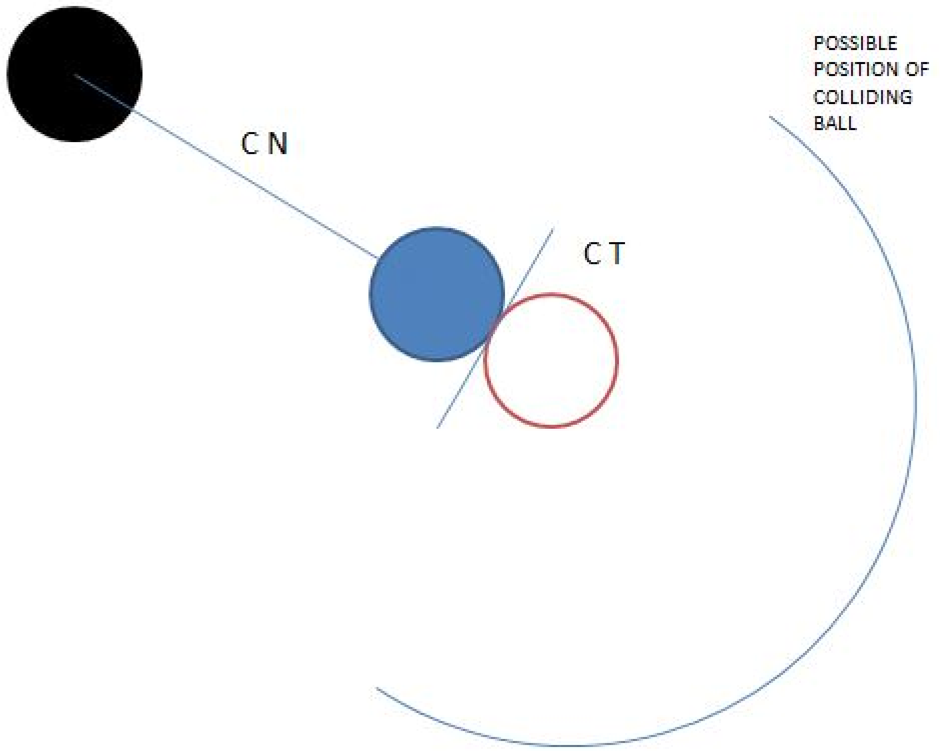


The Game

5.1 Working

The game will work with the interaction of game queue with rules , the rules can themselves create new game Events.
The AI thread is activated when its someones turn ,it takes in the input as game Queue and gives False events througn Keyboard Thread as Input.

6 AI

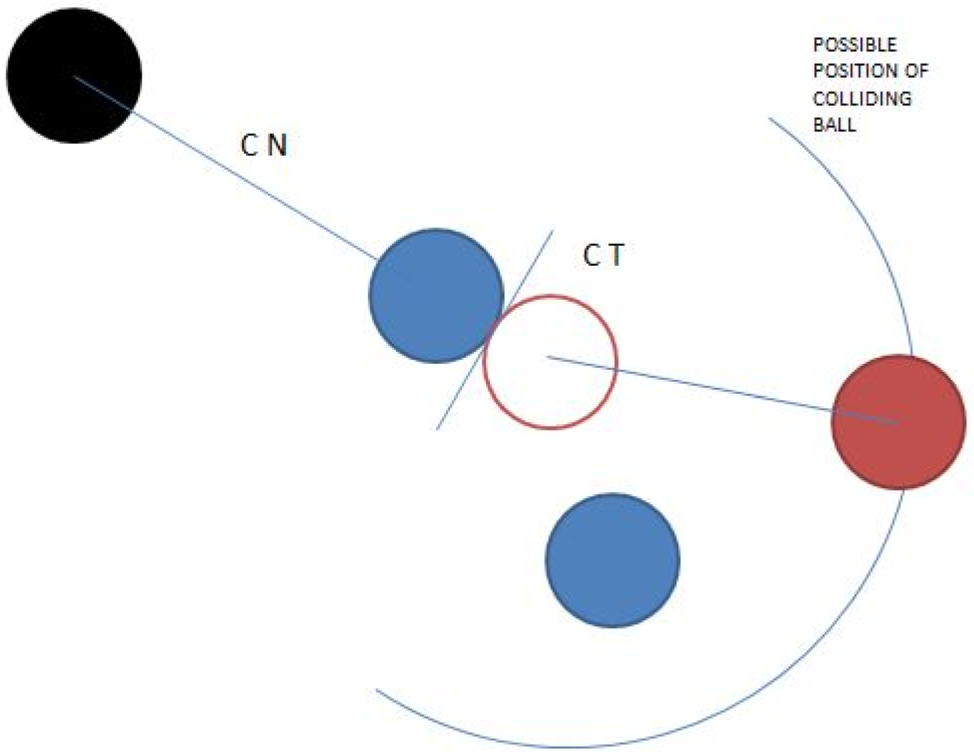


The First Motive is to find a way to find which coin can go in. The coins which can go in .thats easy by tracing path.

6.1 Ghost coins

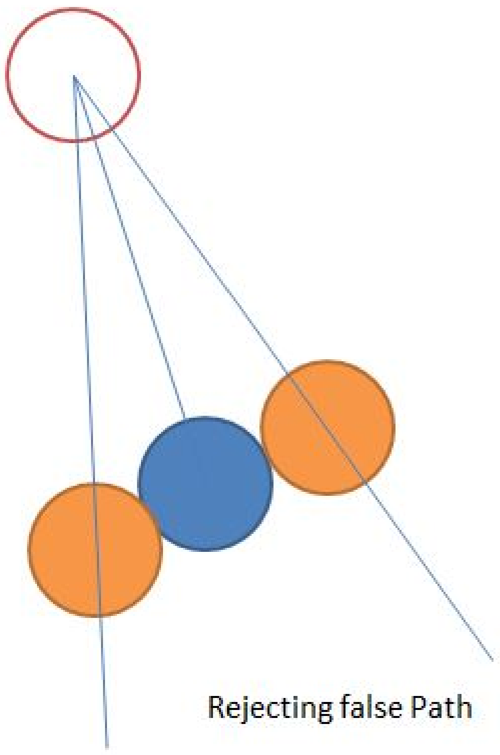
Let me say a coins x is going in, you create a ghost ball , this ball serve as a refrence to collision. now my work is to narrow down the possibilities.

6.1.1 Semicircular Path



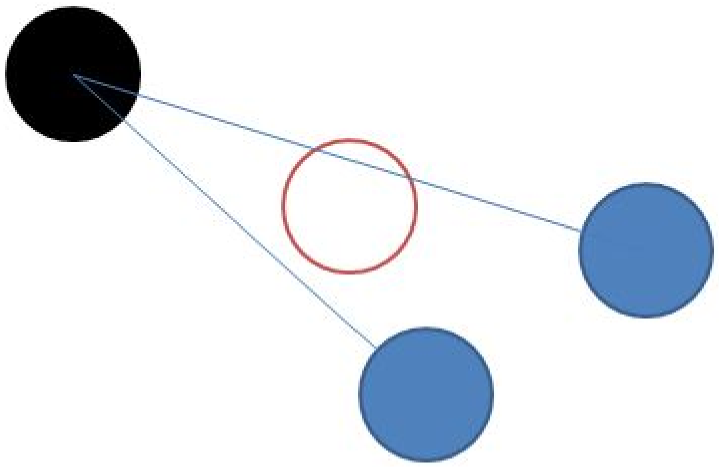
now you know the final ball position and you know the backtrack critical postions to to find what part of the shooting line lies in.

6.1.2 removing illegal paths



you can remove illegal areas by joining the ball with other balls and then remove possible areas

6.1.3 expanding this method to blocking



this method can be Easily expanded to blocking .To block a ball you have to place a ghost ball in front of the possible move. Best position can be found by reducing some possible by multiple blocks.