**Sneaker Seeker Game**

**Game rules**

the player should split to 2 groups: **Sneakers** & **Seekers**.

The goal is to scan the board in time and detect as many Sneakers as possible.

parameters

* The board size is:
* Seekers can communicate up to meters. May be fully connected graph.
* Seekers can see up to meters ahead. In a shape of degree FOV.
* Seekers and Sneaker move at the same velocity speed respectively.
  + Seeker's speed can be raised, for limited time , up to 🡪

Assumptions

* Board is 2D.
* Seekers **may** have enough time to spread around the board before the Sneakers will get in.
* Sneakers come from the same **general** direction.
* Sneakers **arrival time** to board can be estimated.
* Sneaker shouldn't change their direction and speed along the Board.

**Seekers Strategies**

Phase 1 – "Spread Out"

1. **In case of "enough time" before the Sneakers arrives and known direction:**

Spread along the edges of the board with the center being the nominal value of the estimated arrival vector. And the spreading along the edges is a function of the statistical certainty of the arrival vector. Can be modeled as normal distribution of "potential energy".

1. **In case of "not enough time" before the Sneakers arrives and "known direction":**

Line up as and move towards to incoming estimated arrival vector.

1. **In case of "enough time" before the Sneakers arrives and "unknown direction":**

Spread evenly across the inner area of the board.

1. **In case of " not enough time" before the Sneakers arrives and "unknown direction":**

Move as a rectangle structure, into the evenly spread location, inside the Board, while seeking for the Sneakers.

Phase 2 – "Seek & Detect"

1. TBD – some alternatives will be checked and the best algorithm will be implemented.
2. RL algorithm. **Optional.**After completion of the work plan (1-4). Such algorithm will be considered upon availability, interest.

**Work Plan**

1. Phase one, implementing **case A** (**enough time & known direction**) is the trivial solution. We will try to solve the more general case, in which it is impossible to cover the edges, due to the board size.
2. If the Seekers can't cover the edges fully, and some of the Sneakers sneak in, we should go into Phase 2 - "Seek & Detect" Algorithm.
3. Will be checked on some random scenarios for statistical analysis. Amount of Sneakers detected Vs time.
4. Scenario Visualization in Python that will demonstrate the entire "Seek & Detect" process.