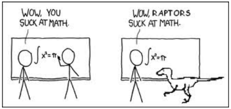
Surviving from the Raptors II:

Toward Interactive Optimization and MFG?

Fayadhoi Ibrahima

Stanford University

CME 334: Advanced Optimization



November 29, 2012

Getting ready for the presentation

Q: What's the best way to talk to a Velociraptor?
A: Long distance!

Setting

We live in a cruel (and pretty strange) world. You (and possibly some of your friends) are thrown in an arena where some velociraptors are unleashed towards you. The setting is simple:

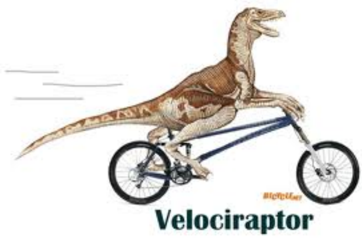
- The arena is rectangular (2D view from the top);
- You (preys) and the raptors are represented by points in the plane;
- You and the raptors respectively have only one constant speed in magnitude;
- The raptors are always running towards the (nearest) prey

Aim

Survive from the raptors and make it to the top!

- The arena has a way out, which corresponds to the upper border
- Find a path that is safe

Of course the task is not simple because you are limited in speed and you want to run as less as possible (running to a constant fast speed demands energy!)



Notations

- the arena is the rectangle $\Omega = [-60; 60] \times [-30; 30]$
- the safe zone is the upper boundary Γ of the rectangle (segment $[-60; 60]$)
- n raptors, m preys
- Position of the i^{th} prey (resp. raptor): $\vec{r}_P^{(i)}$ (resp. $\vec{r}_R^{(i)}$)
- Velocity of the i^{th} prey (resp. raptor): $\vec{v}_P^{(i)}$ (resp. $\vec{v}_R^{(i)}$)
- Constant speed (magnitude) of the i^{th} prey (resp. raptor): $v_P^{(i)}$ (resp. $v_R^{(i)}$)

Get an optimization problem

- Time is discretized into time steps of size $\Delta t = 0.05$
  - $t^{k+1} = t^k + \Delta t$
- the rule for updating the position of raptor i is, if prey j_i is the nearest one to the raptor

$$\vec{r}_R^{(i)}(t^{k+1}) = \vec{r}_R^{(i)}(t^k + \Delta t) = \vec{r}_R^{(i)}(t^k) + v_R^{(i)} \Delta t \frac{\vec{r}_P^{(j_i)}(t^k) - \vec{r}_R^{(i)}(t^k)}{\left\| \vec{r}_P^{(j_i)}(t^k) - \vec{r}_R^{(i)}(t^k) \right\|} \quad (1)$$

Get an optimization problem

- rule for updating the position of prey i is

$$\vec{r}_R^{(i)}(t^{k+1}) = \vec{r}_R^{(i)}(t^k + \Delta t) = \vec{r}_R^{(i)}(t^k) + \Delta t \vec{v}_P^{(i)}(t^k) \tag{2}$$

- **challenge: how to define $\vec{v}_P^{(i)}$?**
- **first observation:**

$$\vec{v}_P^{(i)}(t^k) = v_P^{(i)} \frac{\vec{v}_P^{(i)}(t^k)}{\left\| \vec{v}_P^{(i)}(t^k) \right\|} \tag{3}$$

Get simple bounds constraints

- Prey must stay in the arena: $\vec{l} \leq \vec{r}_P^{(i)}(t) \leq \vec{u}$
  - $\vec{l} = [-60, -30]^T$ and $\vec{u} = [60, 30]^T$
- which gives for the update: $\vec{l} \leq \vec{r}_P^{(i)}(t^k) + \Delta t \vec{v} \leq \vec{u}$
- and gives the simple bounds:

$$\boxed{\vec{L}_k \leq \vec{v} \leq \vec{U}_k} \tag{4}$$

  - $\vec{L}_k = \dfrac{\vec{l} - \vec{r}_P^{(i)}(t^k)}{\Delta t}$ and $\vec{U}_k = \dfrac{\vec{u} - \vec{r}_P^{(i)}(t^k)}{\Delta t}$

Get the objective function

- 2 "objectives":
- run towards the safe zone
  - minimize the distance from the prey(s) to the safe zone
- stay away from the raptors as "cleverly" as possible
  - maximize the distance between the prey(s) and the raptors

First objective

- simple to state
- best choice?

$$F_1(\vec{v}) = \text{dist}(\vec{r}_P^{(i)}(t^k) + \vec{v}\Delta t, \Gamma) \quad (5)$$

Second objective

- choices to be made
- adopt a specific model (optimality?)
- my model: Moderately Weighted Objective (MWO)

$$F_2(\vec{v}) = \sum_{i=1}^{n} \frac{w_i^\theta}{\left\| \vec{r}_P^{(i)}(t^k) + \vec{v}\Delta t - \vec{r}_R^{(i)}(t^k) \right\|} \quad (6)$$

with:

$$w_i^\theta = 2^{\frac{\theta \vec{v}_R^{(i)}}{\max_{1\leq j\leq n} \vec{v}_R^{(j)}}} \quad (7)$$

and θ is a parameter to adapt!

Objective function and Optimization problems

$$F(\vec{v}) = \text{dist}(\vec{r}_P^{(i)}(t^k) + \vec{v}\Delta t, \Gamma) + \sum_{i=1}^{n} \frac{w_i^{\theta}}{\left\| \vec{r}_P^{(i)}(t^k) + \vec{v}\Delta t - \vec{r}_R^{(i)}(t^k) \right\|} \quad (8)$$

and the optimization problem to be solved at each time step k in order to get the update of the positions of the preys is:

$$\min_{\vec{v} \in \mathbb{R}^2} F(\vec{v}) \; s.t. \; \vec{L}_k \leq \vec{v} \leq \vec{U}_k \quad (9)$$

- The objective function makes sense:
  - the closer to the safe zone, the happier
  - you should fear most the fastest raptors, while avoiding getting too close to any of them

Algorithm

Data: positions and velocities for raptors and preys, arena dimensions, time step
Result: successful path for preys, survival times, corresponding parameter θ

initialization;
while *not converging (at least one dead)* **do**
  update θ;
  if *prey attains safezone or is caught* **then**
    break updates for the specific prey;
  else
    find "optimal" v;
    update position of prey ;
    update position of raptors ;
  end
end

Algorithm 1: How to get a safe path

Adaptive Optimization

Taylor the value of θ. The optimal way, within the model, is to solve the unconstrained optimization problem:

$$\max_{\theta \in \mathbb{R}} \frac{h(\theta)}{1 + T(\theta)} \tag{10}$$

where $h(\theta)$ is the sum of binary functions taking value 0 if the corresponding prey is dead, and 1 if he/she made it to the safe zone, and $T(\theta)$ is the function corresponding to the sum (or the max) of the survival times.

- not done yet: just stop when I get a feasible solution
- but looks like survival time is generally an increasing function of θ
- non-smooth function

Test with one prey

- Problem: Find a θ that works
- One prey, multiple raptors

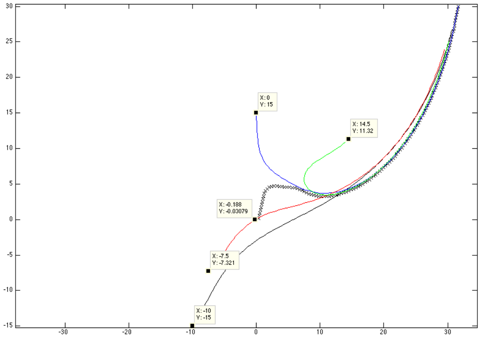


Table: Successful path with one prey($v = 8$), 3 raptors ($v_b = 8.5, v_r = 8, v_g = 7, v_k = 9.5$)

- survival time increasing function of θ (at least compared to the next

Test with multiple preys

- Problem: Find a θ that works
- multiple preys, multiple raptors

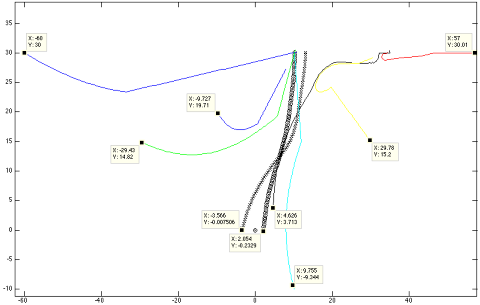


Table: Successful path with 3 preys($v = 8$), 6 raptors ($v_{b1} = 13, v_r = 5, v_g = 12, v_y = 6, v_c = 7, v_{b2} = 4$)

- survival time increasing function of θ (at least compared to the next step)

Observations

- Preys independent but share same strategy (and same θ)
- Stability:
  - positions: with small random perturbations, no real change on (first) θ safe
  - velocities: same phenomena apparently
  - for large enough perturbations (compared to the relative distances), significant change in paths
- Survival time increasing function of θ?
  - makes sense: the more you care about the raptors, the longer you be focusing on them and forget about your objective of getting to the safe zone!
  - but depend on configurations

Learning to get first reasonable parameter guess

- Problem: Find $\theta_{opt} = \theta(\vec{r}_P^{(i)}, \vec{v}_P^{(i)}, \vec{r}_R^{(i)}, \vec{v}_R^{(i)})$
  - i.e. find the best θ (if it exists) in our strategy that insures that every prey has survived and the total amount of efforts is minimized
- Use Machine Learning!
- Separate arena in $k \times k$ (k small) zones
  - continuous possible positions
  - but relative stability in positions
- could use θ from ML as a good first guess

Interactive optimization via MFG

- Assumption: each prey is using the same strategy (same speed and same θ)
- Problem: Find the best strategy
- Aim: keep the independence property but add some "mean field information"
  - i.e. find the best strategy that insures that every prey has survived optimally given the density of other preys
- Use theory of Mean Field Games?
- Issues:
  - is it really helpful?
  - need big number of preys to be applicable
  - heavy math

Applications

- Biology: Preys/predators behaviors in nature
- Finance: portfolio optimization
- US Football
  - Best strategy to make it to the touchdown zone (with minimal effort)
  - (simplified version though)
  - Help coach Shaw & the Stanford team win the Rose Bowl!

Summary

- first pitch in "interactive optimization"
- What was done:
  - define a strategy (objective function)
    - prey sharing the same strategy (collaboration?)
    - anticipation
  - implementation for 1 or multiple preys
  - run tests
- Next steps:
  - try to implement a way of getting θ optimal
  - play more on stability problems
  - different θs for different preys?
  - what would be a better strategy?
  - get the best solution when not feasible
  - turn it more interactive/visual!