agent fielder() {

agent run(dest, speed, init_pos) {

derivate pos = init_pos + $\int_t^t$ speed · (dest − init_pos)

LIVE = $\frac{d}{dt}$(time_to_intercept) ≤ 1 ∧ pos ≠ dest

}

once ball_pos, ball.v

state speed, pos

derivate expected_intercept_point = intercept(pos, ball_pos, ball.v, speed)

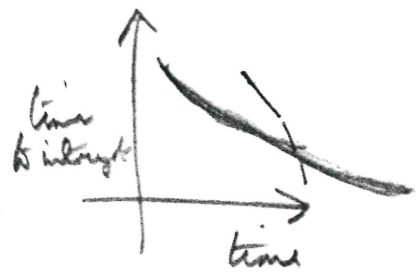
interceptable = is_in_field(expected_intercept_pt)

time_to_intercept = $\frac{\text{dist}(pos, exp\_int\_pt)}{speed}$

}

$$\frac{\text{time\_to\_intercept}' - \text{time\_to\_intercept}}{\text{time} - \text{time}'} > 1$$

means situation is getting worse




---

agent fielder() {

once ball pos, ball.v, pos, speed // knows where ball is / speed / how fast can run / can run where it

state speed, pos

derivate exp_int_pt = intercept(pos, ball_pos, ball.v, speed)

interceptable = is_in_field(exp_int_pt)

time_to_intercept = $\frac{\text{dist}(pos, exp\_int\_pt)}{speed}$

agent run() {

state $t_0$ = time, init_pos = pos, dest = exp_int_pt

derivate pos = init_pos + $\int_{t_0}^{t}$ speed · (dest − init_pos)

LIVE = interceptable ∧ $\frac{d}{dt}$(time_to_intercept) ≤ 1 ∧ pos ≠ ball

More pts.

fielder picks up ball when

ball.pos == pos

but in this context matters that ball.pos is exactly what it should be (authentic value); if inaccurate inside at this point then fielder ~~points~~ can't find up ball. In practice this is about right: fielder may misjudge ball.pos, so do get fluffed interceptions.

---

Fielder has border as a role. (not v.v)

[only certain fielders can play the role]