Safe Robot Follow-the-Leader in the Plane

 $(\circ \circ)$

AUSTIN DAVIS DAVID WISE

15-424 FOUNDATIONS OF CYBER-PHYSICAL SYSTEMS DECEMBER 10, 2014

Scenario

- Leader robot and follower robot
- Traveling in the 2D plane, with potential obstacles
- Various constraints on direction, velocity, and acceleration

Safety Properties

DISTANCE

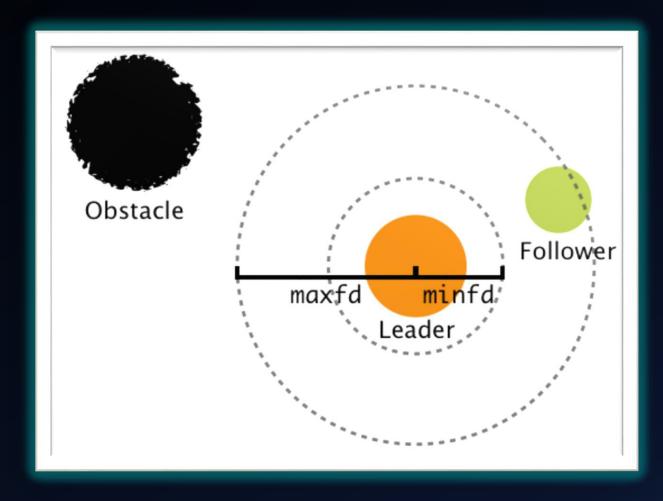
- Minimum following distance
- Robots should not collide with each other

CLOSENESS

- Maximum following distance
- Robots should stay close together



Following Distances





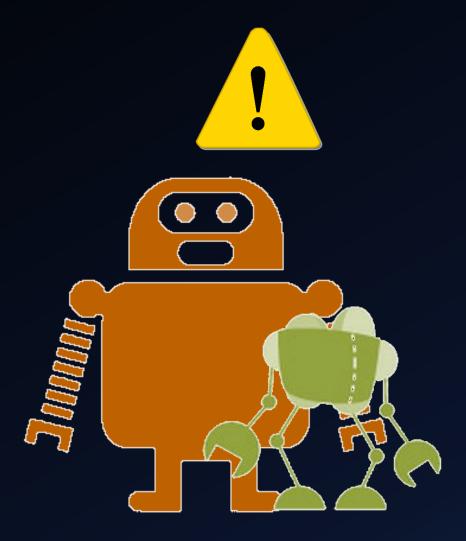
Model Restrictions

- Leader has a maximum velocity
- Leader has a minimum obstacle clearance
- Leader makes wide turns
- Follower must be able to observe leader's state at regular intervals
- Maximum system update interval



PROS Collision

avoidance



PROS

Obstacle avoidance



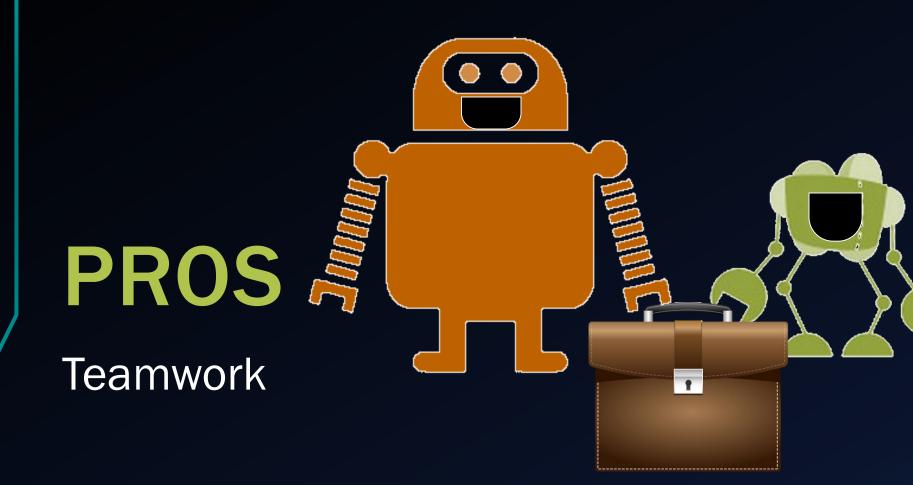


No shepherding

Safe Robot Follow-the-Leader – Austin Davis & David Wise

MANAN

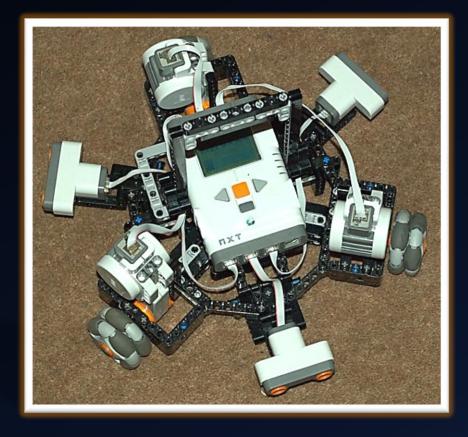




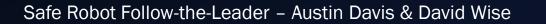
Robots traveling in a group (escort)



Robots with different sensors or capabilities



Path-following robot pairs



 \circ

Robots tethered together

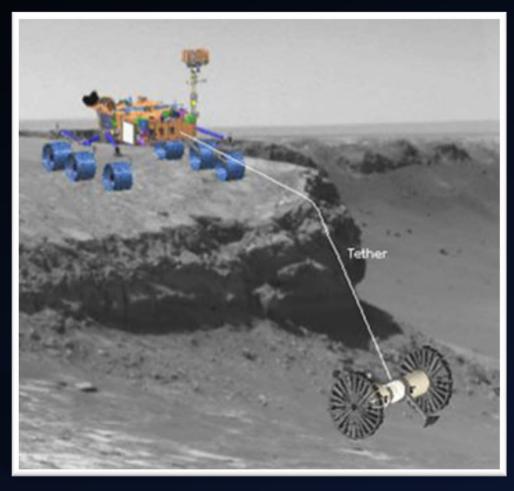
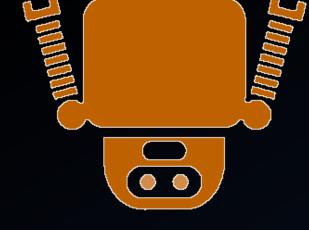


Image by NASA/Jet Propulsion Laboratory



Robots pairs with wireless communication

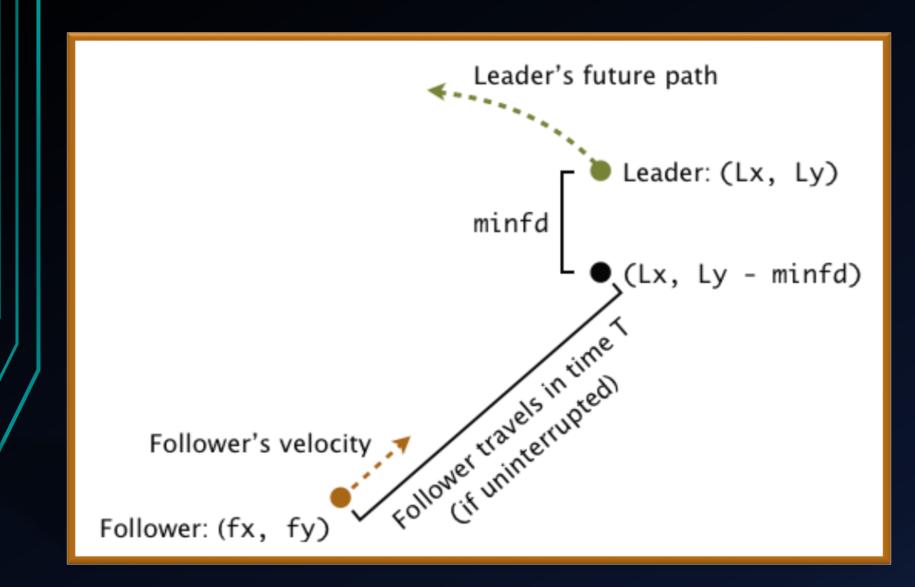




Search and rescue robot pairs (e.g. scout and rescuer)

Intermediate Model

- Leader travels in circular arcs
- Leader can't move down
- Follower travels in straight lines
- Follower uses a time-trigger, and can instantaneously adjust its velocity when making its control decision
- We were able to prove safety of this model



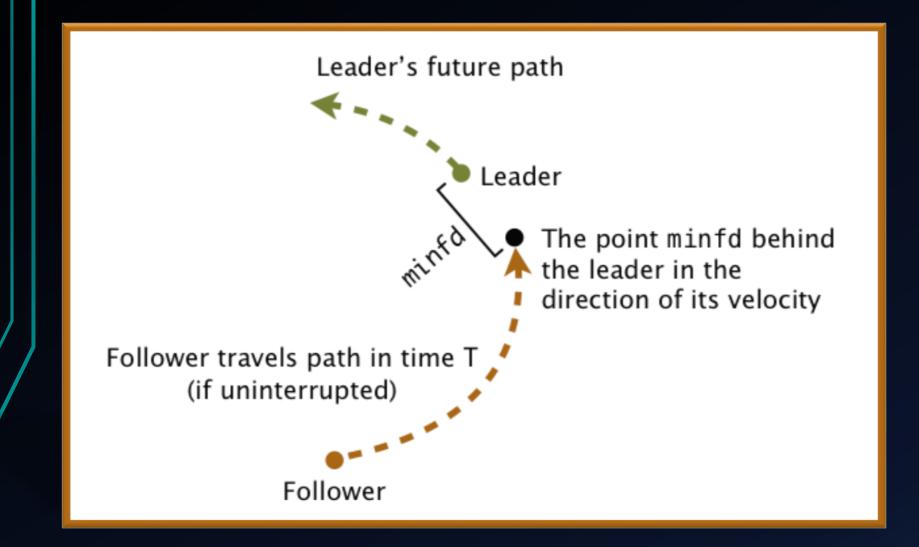
Intermediate Model

| Kg/ | KeYmaera Prover 🛛 🗕 🗖 | × |
|---|--|-------|
| <u>File View Proof Options</u> | | About |
| 🕨 Start 🔄 🚬 Prune Proof 🛛 🕅 Reuse | | |
| Tasks Env. with no model #1 | Inner Node | |
| ☆ intermediate.almost.proof | -> \[| |
| | (d := *; | |
| Proof Search Strategy Rules | $2d \ge 0 \land d^2 = (1x - fx)^2 + (1y - minfd - fy)^2;$ | |
| Proof Hybrid Strategy Goals | ltrackr := * ; | |
| Proof | $?ltrackr \neq 0$; | |
| - I:→r imply right | la := *; | |
| - 2:∧I and left | $2-B \leq la \wedge la \leq A;$ | |
| - I 3:∧I and left | $f_{x} = f_{x} + f_{x$ | |
| - ∉ 4:∧I and left | $f_{VX} := ((1X - 1X) / 1) ,$ $f_{VY} := ((1Y - minfd - fy) / T) ;$ | |
| - I 5:∧I and left | t := 0; | |
| - I of the left | Lx := lx; | |
| - ∉ 7:∧I and left | Ly := ly; | |
| - ∛ 8:∧I and left | d1 := d; | = |
| - ∛ 9:∧I and left | d2 := 0; | |
| -∛ 10:∧I and left | (fx' = fvx, fy' = fvy, | |
| -∛ 11:∧I and left | $lx' = ldx \cdot lv, ly' = ldy \cdot lv,$ | |
| - ∉ 12:[]∧ split box conjunct | $ldx' = (-ldy \cdot ly) / ltrackr,$ | |
| -∛ 13:∧r and right | $ldy' = (ldx \cdot lv) / ltrackr,$ | |
| □ III Case 1 | $lv' = la, d' = (lx - fx) \cdot (ldx \cdot lv - fvx) + (ly - minfd - d)$ | frz) |
| - ∉ 14:[]gen generalization | $d1' = ((fx - Lx) \cdot fvx + (fv - Lv) \cdot fvv) / d1,$ | -y/ |
| ■ ■ Generalisation Holds | d1' = ((1x - Lx) + 1vx + (1y - Ly) + 1vy) / d1, d2' = ((1x - Lx) + 1dx + v + (1y - Ly) + 1dy + 1v) / d2, | |
| - I6:ind loop invariant | $\begin{aligned} u_{2} &= (11x - 1x) + 10x + 1v + (1y - 1y) + 10y + 1v) + u_{2}, \\ t' &= 1, t \leq T, \ ldy \geq 0, \end{aligned}$ | |
| 🗉 🖬 Invariant Initially Vali | $lv \ge 0, lv \cdot T < maxfd - minfd\})*$ | |
| 🛛 🖬 Use Case | $\gamma = 0, \gamma = 1 $ maxia $-$ minimally γ | |
| ■ ■ Body Preserves Invai | \] ($(lx - fx)^2 + (ly - fy)^2$ | |
| -29:Update Simplifica | \geq (minfd) ² | |
| -∛30:∀r all right | | |
| -∛78:∀r all right | $\Lambda \qquad (lx - fx)^2 + (ly - fy)^2$ | |
| -∛79:∀r all right | $\leq (maxfd)^2$ | |
| | | |
| KGY Integrated Deductive Software Design: Ready | | |

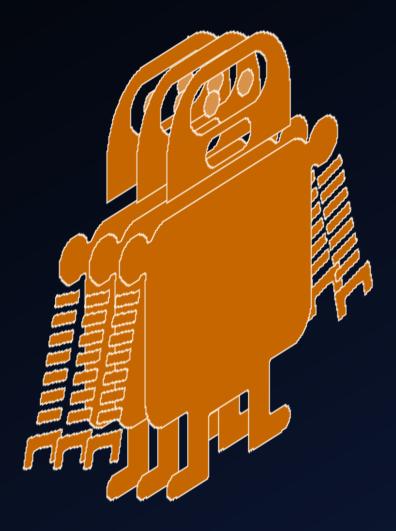


Complete Model

- Follower and leader travel in circular arcs
- Leader cannot turn too sharply
- Safe control decisions are difficult
 - Follower must consider both its ending velocity and ending position

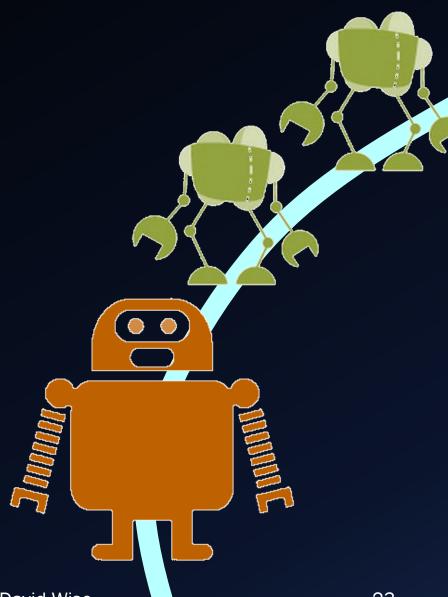


NEXT Extend to 3-D



NEXT

Multiple Followers





UAVs



USES Flocking



