Hardware/Software

Team 8
Hardware Strategy (Initial)

- **2-Motor Differential Drive**
  - 4 Equal-sized wheels
  - One motor controls left, one controls right
  - 4-Wheel Drive
  - 3:1 Gear Reduction

- **Dynamic Sensing**
  - Sweep Sensors across robot’s field of view
  - Buckets and Lights
    - Use Sensors mounted on a pole attached to a Servo for high sweep
  - Rocks and Walls
    - Use ET Sensor mounted on 2nd (upside-down) Servo for low sweep
    - Or use Switches to create bumpers
Hardware Strategy (Final)

- **2-Motor Differential Drive**
  - 4 Equal-sized wheels
  - One motor controls left, one controls right
- **2-Wheel Drive**
- **5:1 Gear Reduction**

**Dynamic Sensing**
- Mount sensors statically on robot chassis
- **Buckets and Lights**
  - Use Sensors mounted on a pole and attached to a Servo for high sweep robot chassis
- **Rocks and Walls**
  - Use ET Sensor mounted on 2nd (upside down) Servo for low sweep
  - Use Switches to create bumpers
Software Design

- Reactive Paradigm Approach - Subsumption and behavior fusion

- Four Behaviors

- Level - 3: ESCAPE OBSTACLES
- Level - 2: AVOID OBSTACLES
- Level - 1: FOLLOW LIGHT
- Level - 0: FOLLOW LIGHT

SENSORY INPUT ➔ MOTOR COMMANDS
Software Strategy

1) Escape Obstacles Behavior
   - To escape large obstacles (buckets)
   - Takes input from range finders
   - Threshold value: 45
   - LEFT_RANGE_FINDER > 45 – Obstacle on Left
   - RIGHT_RANGE_FINDER > 45 – Obstacle on Right
   - IF BOTH RANGE FINDERS > 45 – Obstacles either on both sides or exactly ahead of the robot

2) Avoid Obstacles Behavior
   - To avoid rocks in the arena
   - Takes input from bumpers/switches
   - Similar to Escape obstacles behavior in strategy
3) Follow Light Behavior
   - To search for and go to light source and follow it until it touches that.
   - Takes input from the four light sensors installed in the front and the back.
   - Goes towards the strongest sensor.

4) Wander Behavior (rarely used)
   - Tolerance of 17%.
   - No input.
   - This is the lowest behavior, so will be active when no other behavior is active.
   - When no light is found, this makes the robot to move.
Improvements needed

- Behavior fusion has to improve
- Infinite loop of any two behaviors has to be avoided in a better manner
- Utilization of range finders must be more efficient.
Questions ?

Team 8