cc30.03
exploring robotics
spring 2007
lecture # B.1

topics:

(1) intelligent robots

(2) gearing and variable speed

course web page:

- `http://agents.sci.brooklyn.cuny.edu/cc30.03`

(1) intelligent robots

- autonomy
- problem solving
- modeling
  - knowledge
  - representation
- control architectures
  - deliberative control
  - reactive control
  - hybrid control

(1) autonomy

- to be truly autonomous, it is not enough for a system simply to establish direct numerical relations between sensor inputs and effector outputs
- a system must be able to accomplish *goals*
- a system must be able to *solve problems*
- $\Rightarrow$ need to represent problem space
  - which contains goals
  - and intermediate states
- there is always a trade-off between *generality* and *efficiency*
  - more specialized $\Rightarrow$ more efficient
  - more generalized $\Rightarrow$ less efficient

(1) modeling the robot's environment

- modeling
  - the way in which *domain knowledge* is embedded into a control system
  - information about the environment stored internally: *internal representation*
  - e.g., maze: robot stores a *map* of the maze "in its head"
- knowledge
  - information in a context
  - organized so it can be readily applied
  - understanding, awareness or familiarity acquired through learning or experience
  - physical structures which have correlations with aspects of the environment and thus have a predictive power for the system

(1) knowledge representation

- must have a relationship to the environment
- must enable predictive power (look-ahead), but if inaccurate, it can deceive the system
- difficult because:
  - sensors provide signals, not symbols
  - symbols are often defined with other symbols (circular, recursive)
  - requires interaction with the world, which is noisy
- other factors
  - speed of sensors
  - response time of effectors

(1) control architecture

- a control architecture provides a set of principles for organizing a control system
- provides structure
- provides constraints
- refers to software control level, not hardware!
- implemented in a programming language
- don't confuse "programming language" with "robot architecture"
- architecture guides how programs are structured

(1) classes of robot control architectures

- *deliberative*
  - look-ahead; think, plan, then act
- *reactive*
  - don't think, don't look ahead, just react!
- *hybrid*
  - think but still act quickly
- *behavior-based*
  - distribute thinking over acting

(1) deliberative control

- classical control architecture (first to be tried)
- first used in AI to reason about actions in non-physical domains (like chess)
- natural to use this in robotics at first
- example: Shakey (1960's, SRI)
  - state-of-the-art machine vision used to process visual information
  - used classical planner (STRIPS)
- planner-based architecture
  1. sensing (S)
  2. planning (P)
  3. acting (A)
- requirements
  - lots of time to think
  - lots of memory
  - (but the environment changes while the controller thinks)

(1) reactive control

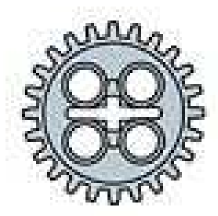
- operate on a short time scale
- does not look ahead
- based on a tight loop connecting the robot's sensors with its effectors
- purely reactive controllers do not use any internal representation; they merely react to the current sensory information
- collection of rules that map situations to actions
  - simplest form: divide the perceptual world into a set of mutually exclusive situations recognize which situation we are in and react to it
  - (but this is hard to do!)
- example: subsumption architecture (Brooks, 1986)
  - hierarchical, layered model

(1) hybrid control

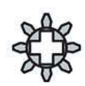
- use the best of both worlds (deliberative and reactive)
- combine open-loop and closed-loop execution
- combine different time scales and representations
- typically consists of three layers:
    1. reactive layer
    2. planner (deliberative layer)
    3. integration layer to combine them
    4. (but this is hard to do!)

(2) gearing and variable speed

- what are gears?
  - gears use small teeth that mesh with the teeth of another gear to transmit motion
  - most gears look just like a wheel with tiny teeth along the outside
  - gears are used in everything from automobiles to small toys
  - we will use spur gears (see pictures). Spur gears are generally identified by the number of teeth they have.
  - common sizes for LEGO spur gears are 8, 16, 24, and 40 tooth.



24-tooth gear



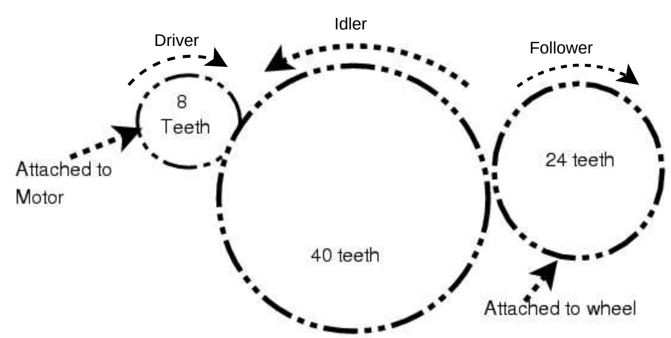
8-tooth gear

(2) function of gears

- what can gears do?
  - change planes of rotation
  - transfer motion
  - increase/decrease Speed (i.e.; gearing up and gearing down)
  - increase/decrease power Torque
  - change direction. idler gear only changes direction, does not affect the gear ratio (speed)

(2) drivers and followers

- driver gear
  - driver is the gear which is being turned by a power source (i.e.; attached to motor).
- follower gear
  - follower is turned by the driver (i.e.; attached to wheel).
- idler gear
  - idler gear is placed directly between two others (i.e.; between driver and follower).
  - it does not affect the gear ratio (rotational speed). Idler gear only changes direction.
  - in the example below, the driver gear is rotating clockwise, idler gear is rotating coutnerclockwise and follower gear is rotating clockwise.

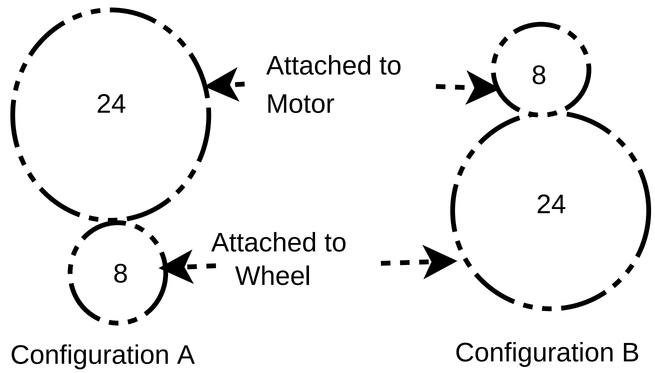


(2) gear ratios and torque

- gear ratios
  - you can calculate the "gear ratio" by using the number of teeth of the "driver gear"(i.e.; attached to motor) divided by the number of teeth of the "follower gear".
  - if **driver gear** has 8-teeth and **follower gear** has 24-teeth, the gear ratio of these two gears is 3:1. Remember, idler gear (i.e.; gear which is between the driver and follower) does not affect the gear ratio (rotational speed). Idler gear only changes direction.
- torque
  - torque is a measure of how much a force acting on an object causes that object to rotate.
  - torque must overcome friction to move the wheels of a vehicle (gravity too if it is on an incline).
  - this force is defined by linear force multiplied by a radius.

(2) controling robot's speed

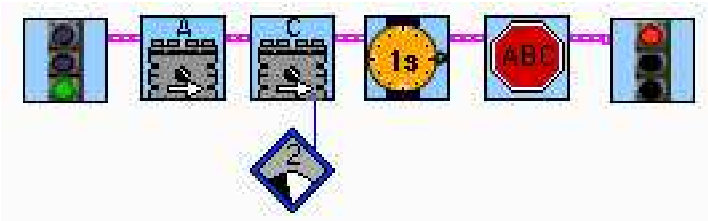
- physical control: gear-ratios and torque



- using gears to make your robot go slower is called **gearing down** where small gear is the driver and large gear is the follower. Gearing down gives you more torque. If your robot is going uphill, you will need more toque. (i.e.; configuration B)
- using gears to make your robot go faster is called **gearing up**. where large gear is the driver and small gear is the follower. Gearing up gives you more speed (i.e.; configuration A).

(2) controling robot's speed

- program contol: power level
  - you can also control the robot's speed from your RoboLab program by adjusting motor's power level.
  - select a power level modifier from the modifier palette and connect to your motor function icon in your program.
  - you can use this program control for turning the robot.



homework assignment (5 points): DARPA Grand Challenege

- read all articles about DARPA grand challenge(posted in our website under Unit B case study) and answer the following questions:
  1. The Blue Team from The University of California at Berkeley chose to use a motorcycle rather than a four (or more) wheeled vehicle which the other teams used.
     1.1. What advantages are there to using a motorcycle?
     1.2. What disadvantages are there in using a motorcycle?
  2. During the 2004 Challenge the Blue Team neglected to turn on the unit responsible for maintaining balance. Consequently, the robot fell over immediately and was unable to right itself. What changes might be made to the system to prevent this sort of blunder?
  3. Team DAD used a Toyota truck with a system of spinning lasers as it's "visual" system. What advantages and or disadvantages does such a system have compared to camera-based systems?
  4. The team from Virginia Tech entered two vehicles. They are nearly identical mechanically but have widely different control strategies. One remembers every object it detects (Rocky) while the other reacts to what it sees but then forgets everything it

saw when it scans its field of view again (Cliff). You might say Rocky used a more deliberative strategy while Cliff used a more reactive strategy. What advantages and or disadvantages would there be to such an approach to the race?

5. The team from Stanford University (which one the 2005 challenge) used redundancy extensively in their entry, Stanley. Most crucial systems are duplicated within Stanley. In addition, their "visual" system consists of radar, stereo video, lasers, more redundancy.

   5.1. What are the advantages of such a design strategy?

   5.2. What are the disadvantages?

- due: unit C.1
- submission format: you should submit hardcopy (typing is strongly recommended or writing clearly) before the beginning of class time on the due date. If you have difficulty to submit the hardcopy, email is acceptable. Assessment is not acceptable after we discuss it in the class.