ME 519 Design Automation – Winter 2017

Group Homework #2 Due: 2/6/17

Automated Design of Gear Trains

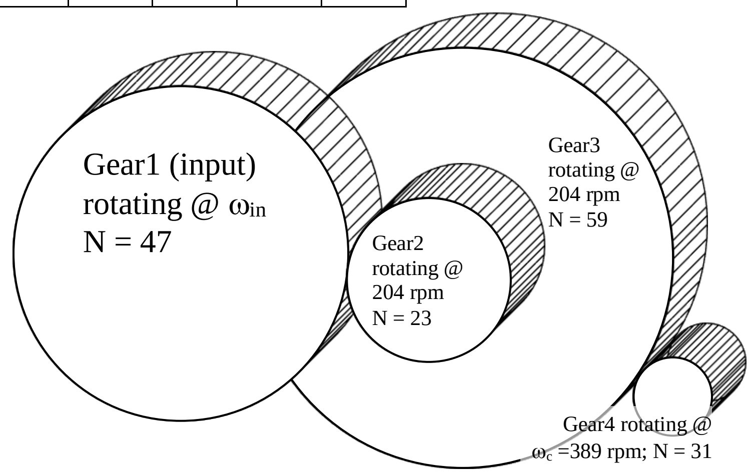
The goal of the algorithm is to find the smallest set of gears that gets can transform the input speed (ω_{in}) to the output speed (ω_{out}) within an error, ε, of 2.5%. Error is calculated as:

$$\varepsilon = \frac{|\omega_{out} - \omega_c|}{|\omega_{out} - \omega_{in}|} * 100\%.$$

The gear trains are to be constructed from a set of gears with the following number of gear teeth:

11	23	31	47	59	71	83	97	109	127

But the problem is a particular gear can only mate with another gear that is less than or equal to 30 teeth away ($|N_1 - N_2| \leq 30$). For example, the 11-tooth gear can mate with the 23 and 31 but not the 47; and the 59-tooth gear can mate with 5 other gears.



Using a search tree, represent and evaluate candidate solutions. For the purpose of this assignment, you can assume no epicyclic gear trains, or worm gear trains are used. Be aware that mating gears rotate in opposite directions, so single idler gears may be used to change direction. In general, most gear trains will be comprised of *transforming-pairs*. As such, the candidate output speed, ω_{in}, can be determined by combining the vector of chosen gears into ratios.

For example, in solving problem (A) below, imagine a candidate to be:

Gear1	Gear2	Gear3	Gear4
input	pair		output
47	23	59	31

$$\omega_C = \left(\frac{47}{23}\right)\left(\frac{59}{31}\right) 100rpm = 389rpm$$

$$\varepsilon = \frac{|117 - 1096|}{|117 - 100|} * 100\% = 255\%.$$

Not at very good candidate!

A) ω_{in} = 100 rpm ω_{out} = -117 rpm	B) ω_{in} = 100 rpm ω_{out} = 77 rpm	C) ω_{in} = 100 rpm ω_{out} = 377 rpm
D) ω_{in} = 100 rpm ω_{out} = -20 rpm	E) ω_{in} = 100 rpm ω_{out} = -2345 rpm	F) ω_{in} = 100 rpm ω_{out} = 2 rpm

1. Representation: Show how you represent the candidates, and transitions for this problem.
2. Evaluation: Show how you evaluate the candidates and check for the goal.
3. Write code for and run DFS to level 10 for the 6 problems above. Show results. What is the average branching factor?
4. Write code for and run BFS. Show and compare results to 3.
5. Write a transition function, g, such that the number of teeth are minimized. Run a uniform cost search on the 6 problems above. Show and compare results to 4.
6. Define a heuristic for this problem. Is it admissible? Write code for and run A* on the six problems.

7. Re-run A with the constraint of 30 teeth-difference relaxed. Compare results.