Day 22 Notes

Zach Neveu

June 12, 2019

1 Agenda

- Genetic Algorithms
- Intro to Network Flow
- Project #5 Introduction

2 Genetic Algorithms (GAs)

- Example: Maximize $f(x) = x^3 - 60x^2 + 900x + 100$ for $x = [0:1:31]$. Answer is x=10, f(x) = 4100.
- Represent solution as a 5-bit binary string
- Start by generating initial population of 5 random bit strings
- Randomly select solutions to mate in proportion to fitness
- Child population will have higher average value for $f(x)$!

Variations of Algorithm

- Selection: instead of eliminating all parents, children could replace random solutions
- Crossover: generate two offspring
- Mating: are both parents in proportion to fitness? Choose one random?
- Selection: lifetime parameter (with randomness) based on fitness?

Algorithm Features Results on Population

- Mating solutions from different bumps of bimodal distribution causes poor solutions
- **speciation:** don't allow mating between very different solutions
- Multipoint crossover possible. Could benefit! Risk is could change too fast.
- Mutation: Could try picking two points, and flipping all bits between points. How to pick points?

Application of GA to TSP

- Biggest problem: finding correct representation of problem and crossover to get valid children
- Trick: select crossover points, and use them to define a series of swaps. Assumes completely connected.
- Trick: if subset of nodes found in same set of locations in two parents, we can exchange them

GA Properties

- Flexible. No gradient, not many features needed to use.
- Slow. No domain knowledge, lots of randomness,
- High quality if given time

3 Project Details

- Implement local search for graph coloring and knapsack
- Two methods for each: steepest descent (very basic), one of the other techniques (tabu search, sim annealing, gas, an algorithm of our own selection (only for one of the two))
- Report compares these results with all previous results

4 Network Flow

- Subset of ILP
- In P
- Can be used to represent many problems we've discussed.
- Good solvers exist

Problem Details

- **Flow Network**: a directed graph $G(V,E)$ where each edge has a non-negative capacity $c(u,v) \geq 0$. For pairs of nodes u, v not in the graph, $c(u,v) = 0$. 2 special nodes: **source**, & **sink**
- **Flow**: a real-valued function mapping a pair of vertices to a scalar
- Rules
  1. For all u, v $f(u,v) \leq c(u,v)$ (capacity constraint)
  2. For all u, v, $f(u,v) = -f(v,u)$ (skew symmetry)

3. For all u, $\sum_v f(u,v) = 0$ (out = in except for source/sink)

- **Maximum Flow Problem**: given a flow network, a source and a sink, find a flow of maximum value.
- Any multi-source, multi-sink problem can just have a single sink, single source, then have infinite capacity links to original sources/sinks. This shows that solving the one-source/one-sink problem solves the multiple case.

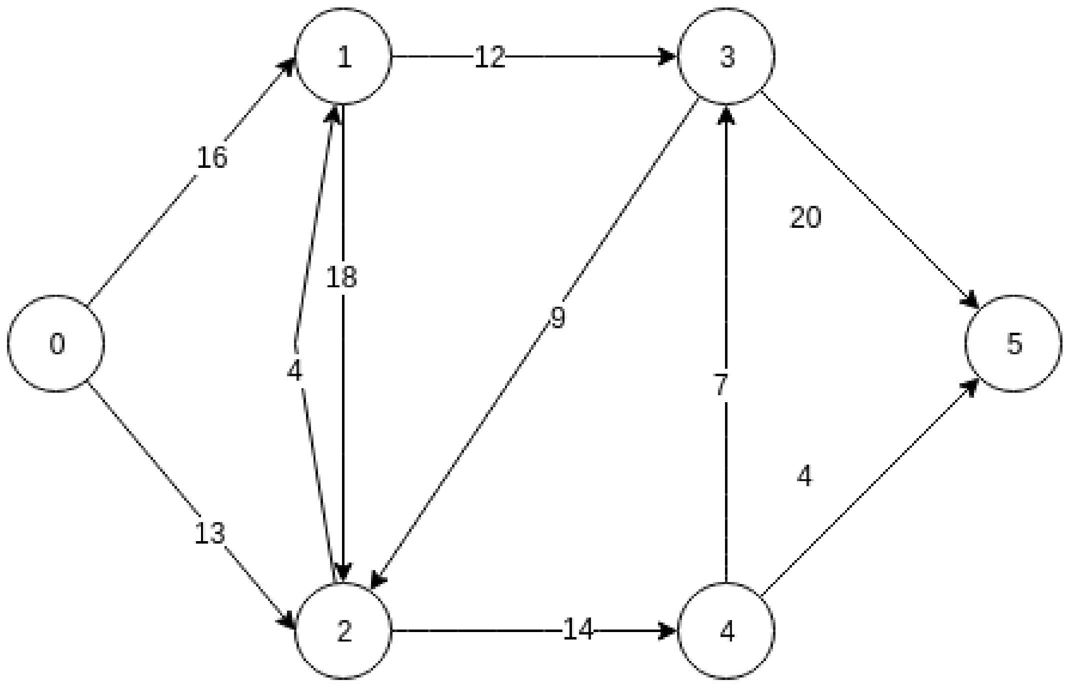

Figure 1: Example Flow Graph