CS540 Introduction to Artificial Intelligence
Lecture 17

Young Wu
Based on lecture slides by Jerry Zhu, Yingyu Liang, and Charles Dyer

July 27, 2020

Course Evaluation

Admin

- M11 is course evaluation on AEFIS: it will be opened on August 10 to August 21. Submissions of M11 before August 10 will receive 0.
- All M8 to M12 and P1 to P6 will be due on August 20 midnight.
- TA's Review Session on August 10. Final exams on August 17 (Version A) and August 18 (Version B), same format as the midterm.

5:30 - 8:30

5 → 9 pick 3 hours

Coordination Game

Admin

- You are not allowed to discuss anything about this question in the public chat. There will be around 5 new questions on the final exam. I will post n of them before the exam (probably next Tuesday):
- A: $n = 0$.
- B: $n = 1$ if more than 50 percent of you choose B.
- C: $n = 2$ if more than 75 percent of you choose C.
- D: $n = 3$ if more than 98 percent of you choose D.
- E: $n = 0$.
- I will repeat this question a second time. If you fail to coordinate both times, I will not post any of the new questions.

Coordination Game Repeat

Admin

Q3

- You are not allowed to discuss anything about this question in the public chat. There will be around 5 new questions on the final exam. I will post n of them before the exam (probably next Tuesday):
- A: $n = 0$.
- B: $n = 1$ if more than 50 percent of you choose B.
- C: $n = 2$ if more than 75 percent of you choose C.
- D: $n = 3$ if more than 98 percent of you choose D.
- E: $n = 0$.

Local Search

Motivation

- Local search is about searching through a state space by iteratively improving the cost to find an optimal or near-optimal state.
- The successor states are called the neighbors (sometimes move set).
- The assumption is that similar (nearby) solutions have similar costs.

Hill Climbing (Valley Finding)

Description

- Start at a random state.
- Move to the best neighbor state (one of the successors).
- Stop when all neighbors are worse than the current state.
- The idea is similar to gradient descent.

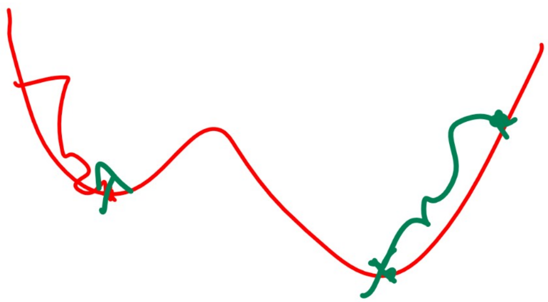
Boolean Satisfiability Example 1

Quiz

- Assume all variables A, B, C, D, E are set to True. How many of the following clauses are satisfied?
  - $A \vee \neg B \vee C$
  - $\neg A \vee C \vee D$
  - $B \vee D \vee \neg E$
  - $\neg C \vee \neg D \vee \neg E$
  - $\neg A \vee \neg C \vee E$

Boolean Satisfiability Example 2

Quiz



Compare cost with multiple initial states.

- Assume all variables A, B, C, D, E are set to True. Which one of the variables should changed to False to maximize the number of clauses satisfied?
- $A \vee \neg B \vee C$
- $\neg A \vee C \vee D$
- $B \vee D \vee \neg E$
- $\neg C \vee \neg D \vee \neg E$
- $\neg A \vee \neg C \vee E$

C, or D.

Boolean Satisfiability Example 3

Quiz

- Assume all variables A, B, C, D, E are set to True. Which one of the variables should changed to False to maximize the number of clauses satisfied?
- $\neg A \vee \neg B \vee \neg C$
- $\neg A \vee \neg B \vee \neg D$
- $\neg A \vee \neg C \vee \neg D$
- $\neg A \vee \neg B \vee \neg D$
- $\neg B \vee \neg C \vee \neg D$

Hill Climbing

Algorithm

- Input: state space S and cost function f.
- Output: $s^\star \in S$ that minimizes $f(s)$.
- Start at a random state s_0.
- At iteration t, find the neighbor that minimizes f.

$$s_{t+1} = \arg \min_{s \in s'(s_t)} f(s)$$

- Stop when none of the neighbors have a lower cost.

$$\text{stop if } f(s_{t+1}) \leqslant f(s_t)$$

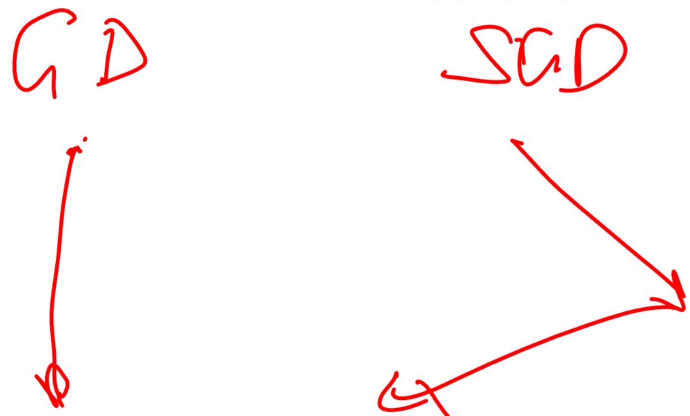
Random Restarts

Discussion

- A simple modification is picking random initial states multiple times and finding the best among the local minima.

First Choice Hill Climbing

Discussion



- If there are too many neighbors, randomly generate neighbors until a better neighbor is found.
- This method is called first choice hill climbing.

Simulated Annealing

Description

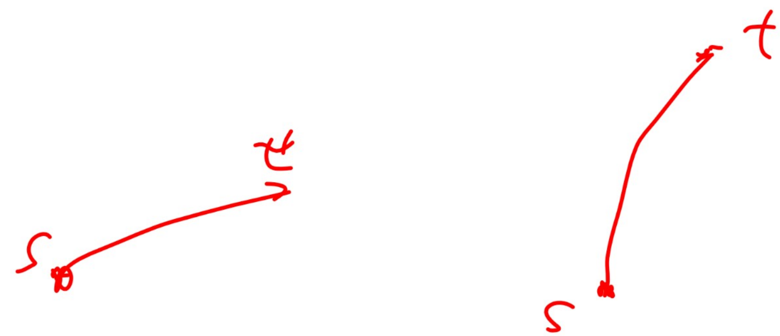
- Each time, a random neighbor is generated.
- If the neighbor has a lower cost, move to the neighbor.
- If the neighbor has a higher cost, move to the neighbor with a small probability.
- Stop until bored.
- It is a version of Metropolis-Hastings Algorithm.

Acceptance Probability

Definition

- The probability of moving to a state with a higher cost should be small.

1. Constant: $p = 0.1$
2. Decreases with time: $p = \frac{1}{t}$
3. Decreases with time and as the energy difference increases:
$$p = \exp\left(-\frac{|f(s') - f(s)|}{\text{Temp}\ (t)}\right)$$



- The algorithm corresponding to the third idea is called simulated annealing. Temp should be a decreasing in time (iteration number).

Temperature

Definition

- Temp represents temperature which decreases over time. For example, the temperature can change arithmetically or geometrically.

$$\text{Temp }(t+1) = \max\{\text{ Temp }(t) - 1, 1\}, \text{ Temp }(0) = \text{ large}$$
$$\text{Temp }(t+1) = 0.9 \text{ Temp }(t), \text{ Temp }(0) = \text{ large}$$

- High temperature: almost always accept any s'. random.
- Low temperature: first choice hill climbing.

→ only move to lower cost states

Simulated Annealing Example 1

Quiz

- Suppose we are minimizing and $f(s) = 6$, $f(t) = 5$, $T = 4$. What is the probability we move from s to t in the next step? What is the probability we move from t to s in the next step?

cost cost

t is better than s

prob = 1

$$\text{prob} = \exp\left(-\frac{|6-5|}{4}\right) = e^{-\frac{1}{4}}$$

Simulated Annealing Example 2

Quiz

Q5

- Suppose we are minimizing and $f(s) = 0, f(t) = \log(5)$, $T = 1$. What is the probability we move from s to t.
- A: 0
- B: $\frac{1}{5}$
- C: $\frac{4}{5}$
- D: 1
- E: 5

$e^{\log a} = a$

$e^{-\frac{|f(s) - f(t)|}{T}}$

$s \to t$ if s is better

$e^{-\frac{\log 5}{1}} = \left(e^{\log 5}\right)^{-1} = \frac{1}{5}$

Simulated Annealing

Algorithm

- Input: state space S, temperature function Temp, and cost function f.
- Output: $s^{\star} \in S$ that minimizes $f(s)$.
- Start at a random state s_0.
- At iteration t, generate a random neighbor s', and update the state according to the following rule.

$$s_{t+1} = \begin{cases} s' & \text{if } f(s') < f(s_t) \\ s' & \text{with probability } \exp\left(-\dfrac{|f(s') - f(s_t)|}{\text{Temp}(t)}\right) \\ s_t & \text{otherwise} \end{cases}$$

Simulated Annealing Performance

Discussion

- Use hill-climbing first.
- Neighborhood design is the most important.
- In theory, with infinitely slow cooling rate, SA finds global minimum with probability 1.

Genetic Algorithm

Description

- Start with a fixed population of initial states.
- Find the successors by:

1. Cross over.
2. Mutation.

Reproduction Probability

Definition

- Each state in the population has probability of reproduction proportional to the fitness. Fitness is the opposite of the cost: higher cost means lower fitness. Use F to denote the fitness function, for example, $F(s) = \dfrac{1}{f(s)}$ is a valid fitness function.

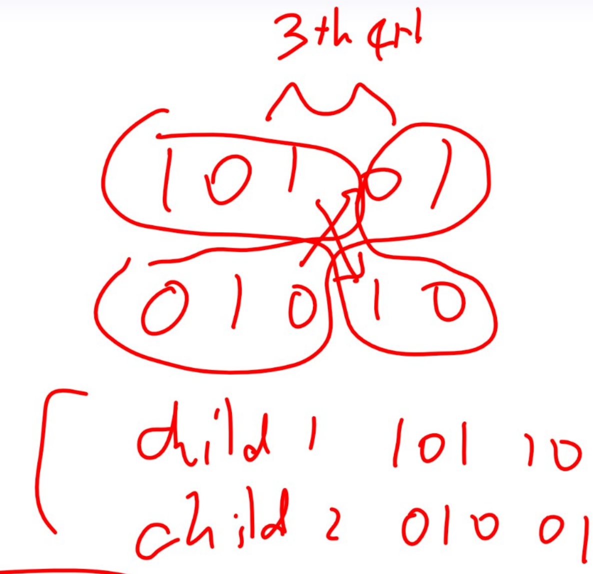
cost −f(s)

$$p_i = \frac{F(s_i)}{\sum_{j=1}^{N} F(s_j)}, i = 1, 2, ..., N$$

- A pair of states are selected according to the reproduction probabilities (using CDF inversion).

Cross Over

Definition



- The states need to be encoded by strings.
- Cross over means swapping substrings.
- For example, the children of 10101 and 01010 could be the same as the parents or one of the following variations.

$$(11010, 00101), (10010, 01101)$$
$$(10110, 01001), (10100, 01011)$$

Mutation

Definition

- The states need to be encoded by strings.
- Mutation means randomly updating substrings. Each character is changed with small probability q, called the mutation rate.
- For example, the mutated state from 000 could stay the same or be one of the following.

$$\text{one of } 001, 010, 100, \text{ with probability } q(1-q)^2$$
$$\text{one of } 011, 101, 110, \text{ with probability } q^2(1-q)$$
$$\text{and } 111, \text{ with probability } q^3$$

Fitness Example 1

Quiz

- Fall 1999 Final Q5
- Which ones (multiple) of the following states have the highest reproduction probability?
- The fitness function is $f(x) = 5x_1 + 3x_2x_3 - x_4 + 2x_5$.
- A: $(1, 1, 0, 1, 1)$
- B: $(0, 1, 1, 0, 1)$
- C: $(1, 1, 0, 0, 0)$
- D: $(1, 0, 1, 1, 1)$
- E: $(1, 0, 0, 0, 0)$

Fitness Example 2

Quiz

- Which one of the following states have the highest reproduction probability? The fitness function is $f(x) = \min\{t \in \{1, 2, 3, 4, 5, 6\} : x_t = 1\}$ with $x_6 = 1$.
- A: $(0, 0, 1, 0, 0)$
- B: $(0, 1, 0, 0, 1)$
- C: $(0, 0, 1, 1, 0)$
- D: $(0, 0, 0, 1, 0)$
- E: $(0, 0, 0, 0, 0)$

Fitness Example 3

Quiz

Q7 (last)

- Which one of the following states have the highest reproduction probability? The fitness function is $f(x) = \max\{t \in \{0, 1, 2, 3, 4, 5\} : x_t = 1\}$ with $x_0 = 1$.
- A: $(0, 0, 1, 0, 0)$ 3
- B: $(0, 1, 0, 0, 1)$ 5 ←
- C: $(0, 0, 1, 1, 0)$ 4
- D: $(0, 0, 0, 1, 0)$ 4
- E: $(0, 0, 0, 0, 0)$ 6

Genetic Algorithm, Part I

Algorithm

- Input: state space S represented by strings s and cost function f or fitness function F.
- Output: $s^\star \in S$ that minimizes $f(s)$.
- Randomly generate N solutions as the initial population.

$$s_1, s_2, ..., s_N$$

- Compute the reproduction probability.

$$p_i = \frac{F(s_i)}{\sum_{j=1}^{N} F(s_j)}, i = 1, 2, ..., N$$

Genetic Algorithm, Part II

Algorithm

- Randomly pick two states according to p_i, say s_a, s_b. Randomly select a cross over point c, swap the strings.

$$s'_a = s_a\left[0 \ldots c\right) s_b\left[c \ldots m\right)$$
$$s'_b = s_b\left[0 \ldots c\right) s_a\left[c \ldots m\right)$$

- Randomly mutate each position of each state s_i with a small probability (mutation rate).

$$s'_i\left[k\right] = \begin{cases} s_i\left[k\right] & \text{with probability } 1 - q \\ \text{random} & \text{with probability } q \end{cases}, k = 1, 2, ..., m$$

- Repeat with population s'.

Genetic Algorithm Performance

Discussion

- Use hill-climbing first.
- State design is the most important.
- In theory, cross over is much more efficient than mutation.