18-660: Numerical Methods for Engineering Design and Optimization

Xin Li
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
Overview

- Stochastic Optimization
  - Simulated annealing
Local Optimization

- All optimization algorithms in early lectures assumes “local convexity” for cost function and constraint set
  - Gradient method
  - Newton method
  - Conjugate gradient method
  - Interior point method

- Global convergence cannot be guaranteed if the actual cost function or constraint set is non-convex
Filter Design Example

- Design a band-stop filter to remove power supply noise

Input Signal → Filter 1 → Filter 2 → Output Signal

Required frequency response

\[ |H(f)| \]

- 60 ± 5 Hz
- 120 ± 5 Hz

Required frequency response vs. f
Design a band-stop filter to remove power supply noise

Filter Design Example

uityd
Input
Signal

|H(f)|

Filter 1

f

Filter 2

OR

Filter 1

f

Filter 2

f

Output
Signal
Design a band-stop filter to remove power supply noise

Feasible set is not continuous in this example!
Stochastic Optimization

- Stochastic optimization is another useful technique for nonlinear programming
  - Randomized algorithm (not deterministic)
  - Better convergence than local optimization
  - More expensive in computational cost

- Several important algorithms for stochastic optimization
  - Simulated annealing (focus of this lecture)
  - Genetic programming
Simulated Annealing

- **Unconstrained optimization**
  \[ \min_X f(X) \]

- **Simulated annealing:**
  - Start from an initial point
  - Repeatedly consider various new solution points
  - Accept or reject some of these solution candidates
  - Converge to the optimal solution
Simulated Annealing

- Unconstrained optimization
  \[ \min_X f(X) \]

- Simulated annealing was introduced by Metropolis in 1953

- It is based on “similarities” and “analogies” with the way that alloys manage to find a nearly global minimum energy level when they are cooled slowly
Simulated Annealing

- Local optimization vs. simulated annealing

- Local optimization
  - Start from an initial point
  - Repeatedly consider various new solution points
  - Reduce cost function at each iteration
  - Converge to optimal solution

- Simulated annealing
  - Start from an initial point
  - Repeatedly consider various new solution points
  - Accept/reject new solution using probability at each iteration
  - Converge to optimal solution
Simulated Annealing

- Local optimization

Local optimization attempts to reduce cost function at each iteration
Simulated Annealing

- Simulated annealing

Simulated annealing accept/reject new solution candidate based on probability

\[ f(x) \]

Always

Probably

Likely

Always
Simulated Annealing

- Step 1: start from an initial point \( X = X_0 \) & \( K = 0 \)
- Step 2: evaluate cost function \( F = f(X_K) \)
- Step 3: randomly move from \( X_K \) to a new solution \( X_{K+1} \)
- Step 4: if \( f(X_{K+1}) < F \), then
  - Accept new solution
  - \( X = X_{K+1} \) & \( F = f(X_{K+1}) \)
- End if
- Step 5: if \( f(X_{K+1}) \geq F \), then
  - Accept new solution with certain probability
  - \( X = X_{K+1} \) & \( F = f(X_{K+1}) \) iff \( \text{rand}(1) < \epsilon \)
- End if
- Step 6: \( K = K + 1 \) & go to Step 2
Simulated Annealing

- Accept/reject new solution with the probability $\varepsilon$
  - If $f(X_{K+1}) \geq F$, then
    - Accept new solution with certain probability
    - $X = X_{K+1}$ & $F = f(X_{K+1})$ iff $\text{rand}(1) < \varepsilon$
  - End if

- Option 1
  - Constant probability, i.e., $\varepsilon = 0.1$

- Option 2 (better than Option 1)
  - Dynamically varying probability, i.e., decreasing over time
Simulated Annealing

- Accept/reject new solution with the probability $\varepsilon$
  - If $f(X_{K+1}) \geq F$, then
    - Accept new solution with certain probability
    - $X = X_{K+1} \& F = f(X_{K+1})$ iff $\text{rand}(1) < \varepsilon$
  - End if

- Use Boltzmann distribution to determine the probability $\varepsilon$

\[ \varepsilon = \exp\left[ -\frac{f(X_{K+1}) - F}{T_{K+1}} \right] \]

- $T_{K+1}$ is a “temperature” parameter that gradually decreases
- E.g., $T_{K+1} = \alpha \cdot T_K$ where $\alpha < 1$
Simulated Annealing

- **Accept/reject new solution with the probability $\varepsilon$**
  - If $f(X_{K+1}) \geq F$, then
    - Accept new solution with certain probability
    - $X = X_{K+1} \& F = f(X_{K+1})$ iff
      \[
      \text{rand}(1) \leq \exp\left( - \frac{f(X_{K+1}) - F}{T_{K+1}} \right)
      \]
  - End if

- **High temperature**
  - Attempt to accept all new solutions even if $f(X_{K+1}) - F$ is large

- **Low temperature**
  - Only accept the new solutions where $f(X_{K+1}) - F$ is small
Simulated Annealing

- Simulated annealing is particularly developed for unconstrained optimization

- Constrained optimization can be converted to unconstrained optimization using barrier method

\[
\begin{align*}
\min_x & \quad f(X) \\
\text{S.T.} & \quad g(X) \leq 0
\end{align*}
\]

\[
\min_x \quad f(X) - \frac{1}{t} \cdot \log[-g(X)]
\]
Simulated Annealing

- **Simulated annealing does not guarantee global optimum**
  - However, it tries to avoid a large number of local minima
  - Therefore, it often yields a better solution than local optimization

- **Simulated annealing is not deterministic**
  - Whether accept or reject a new solution is random
  - You can get different answers from multiple runs

- **Simulated annealing is more expensive than local optimization**
  - It is the price you must pay to achieve a better optimal solution
Simulated Annealing

- Simulated annealing has been used to solve many practical engineering problems

- A large number of implementation issues must be considered for practical circuit optimization problems
  - How to define optimization variable X (continuous vs. discrete)?
  - How to randomly move to a new solution?
  - Etc.
Example: Travelling Salesman Problem (TSP)

- N cities are located on a 2-D map
- One must visit each city once and then return to start city
- Find the optimal route with minimum length
  - If all cities are visited in the order of \( R = \{C_1, C_2, ..., C_N\} \), we have

\[
f(R) = \|C_1 - C_2\|_2 + \|C_2 - C_3\|_2 + \cdots + \|C_N - C_1\|_2
\]

Distance between \( C_1 \) and \( C_2 \)
Distance between \( C_N \) and \( C_1 \)
Example: Travelling Salesman Problem (TSP)

- Step 1: start from random route $R$, initial temperature $T$ & $K = 1$
- Step 2: evaluate cost function $F = f(R)$
- Step 3: define new route $R_K$ by randomly swapping two cities
- Step 4: if $f(R_K) < F$, then
  - Accept new route
  - $R = R_K$ & $F = f(R_K)$
- End if
- Step 5: if $f(R_K) \geq F$, then
  - Accept new solution with certain probability
  - $R = R_K$ & $F = f(R_K)$ iff $\text{rand}(1) < \exp\{[F - f(R_K)]/T\}$
- End if
- Step 6: $T = \alpha T$ ($\alpha < 1$), $K = K + 1$, and go to Step 3
Example: Travelling Salesman Problem (TSP)

- TSP route optimized by simulated annealing

Initial route

Optimized route
Summary

- Stochastic optimization
  - Simulated annealing