Inteligência Computacional - 1

Annealing **Simulado**

$\rightarrow$ Metaheurística de otimização baseada em conceitos da Física Estatística

Física Estatística: Lida com sistemas de muitas partículas (muitos graus de liberdade)

Física Estatística: Conceitos Básicos

• *Ensemble* Estatístico: Conjunto de cópias virtuais do sistema de interesse

Cada elemento do *ensemble* pode assumir um qualquer dos 'estados acessíveis' do sistema (estados microscópicos)

e.g., Gás ideal à temperatura constante, T_0

$\rightarrow$ Estados acessíveis satisfazem $PV \propto T_0$

Há uma infinidade de estados microscópicos consistentes com esta condição macroscópica

• Condição de Equilíbrio

$\rightarrow$ O *ensemble* não varia com o tempo

A probabilidade de encontrar um estado qualquer no *ensemble* é constante

• Condição de Ergodicidade

→ Num tempo suficientemente longo, um elemento qualquer do *ensemble* passa por todos os seus estados acessíveis

A média **temporal** de uma grandeza física, tomada em um elemento qualquer do *ensemble*, é igual à média tomada sobre todo o *ensemble* **num dado instante** de tempo

• Tipos de *Ensemble*

Ensemble Microcanônico

→ Sistema isolado; Não há troca de energia ou partículas

Ensemble Canônico

→ Sistema em equilíbrio térmico; Há troca de energia com um sistema maior (*banho térmico*)

Ensemble Gran-Canônico

→ Há troca de energia e partículas

- *Ensemble* Microcanônico

Sistema isolado $\rightarrow$ A energia se conserva

Distribuição de probabilidade dos estados do sistema à energia E_0 :

$$P_\mu(X) = \begin{cases} K & : \quad E(X) = E_0 \\ 0 & : \quad E(X) \neq E_0 \end{cases}$$

onde X é um estado acessível, e K é uma constante

- *Ensemble* Canônico

Sistema (S) pode trocar energia com um sistema maior (S')

A energia da combinação dos dois sistemas se conserva:

$$E_S + E_{S'} = E_0$$

$\rightarrow$ A probabilidade de encontrar o sistema S no estado X é igual à probabilidade de encontrar o sistema S' na energia

$$E_{S'} = E_0 - E(X)$$

Assim,

$$P_c(X) = C'\Omega'(E_0 - E(X))$$

onde

$\Omega'(E) \equiv$ Número de estados de S' com energia E

$C' \equiv$ Constante de normalização

Como $S' >> S$ e $E(X) << E_0$,

$$\log \Omega'(E_0 - E(X)) = \log \Omega'(E_0) - \beta E(X)$$

onde

$$\beta = \left[\frac{\partial \log \Omega'}{\partial E'}\right]_0$$

Assim,

$$\Omega'(E_0 - E(X)) = \Omega'(E_0) \exp(-\beta E(X))$$

de onde se obtém a distribuição de Gibbs (ou Boltzmann)

$$P_c(X) = C \exp(-\beta E(X)) = \frac{\exp\left(-\beta E(X)\right)}{\Sigma_X \exp\left(-\beta E(X)\right)}$$

A grandeza β caracteriza o *banho térmico*, S', e costuma ser expressa como

$$\beta = \frac{1}{k_B T}$$

onde

$T \equiv$ temperatura absoluta do sistema S'

$k_B \equiv$ constante de Boltzmann

$\rightarrow$ No *ensemble* canônico, a temperatura é constante

• O Algoritmo de Metropolis (1953)

$\rightarrow$ Gera amostras da distribuição de Gibbs

A partir de um estado inicial arbitrário, modificações aleatórias são iterativamente propostas no estado do sistema, e aceitas com probabilidade

$$P = \min[1, \exp\left(\frac{-\Delta E}{k_B T}\right)]$$

onde

$\Delta E \equiv E_{fin} - E_{ini}$

$\equiv$ Variação de energia acarretada pela mudança de estado proposta

Assim,

Se $\Delta E \leq 0$, a mudança é aceita

Se $\Delta E > 0$, a mudança é aceita se $\exp(-\Delta E/k_B T) < R$

onde $R \in [0, 1]$ é um número aleatório

- Após várias iterações, o conjunto dos estados obtidos (*ensemble*) converge para uma configuração de equilíbrio que satisfaz a distribuição de Gibbs, i.e.,

$$N_X \propto \exp(-E(X)/k_B T)$$

onde N_X é o número de vezes que o estado X é encontrado no *ensemble*

Prova:

Assumindo $E(X) > E(Y)$,

$\rightarrow N_X P_{XY}$ é o número de mudanças de estado $X \rightarrow Y$

$\rightarrow N_Y P_{YX} \exp\{[E(Y) - E(X)]/k_B T\}$ é o número de mudanças de estado $Y \rightarrow X$

No algoritmo de Metropolis, $P_{XY} = P_{YX}$, i.e., todas as mudanças de estado são eqüiprováveis

$\rightarrow$ Número líquido de transições $Y \rightarrow X$:

$$N_{Y \rightarrow X} = P_{XY}(N_Y \exp\{[E(Y) - E(X)]/k_B T\} - N_X)$$

No equilíbrio, $N_{Y \rightarrow X} = 0$

Assim,

$$\frac{N_X}{N_Y} = \frac{\exp[-E(X)/k_B T]}{\exp[-E(Y)/k_B T]}$$

Algoritmo de Metropolis

Dado o sistema à temperatura T

Escolha um estado arbitrário s

do

Faça uma mudança aleatória de estado $r \leftarrow R(s)$

$\Delta E \leftarrow E(r) - E(s)$

; Aceite estados de energia mais baixa

if $\Delta E \leq 0$ **then** $s \leftarrow r$

; Aceite estados de energia mais alta com probabilidade P

else

$P \leftarrow \exp(-\Delta E/T)$

$x \leftarrow$ número randômico em $[0, 1]$

if $x < P$ **then** $s \leftarrow r$

if Não há decréscimo significativo em E por várias iterações

then Stop

Annealing **Simulado**

$\rightarrow$ Paralelo entre a obtenção das configurações de mínimo custo em um problema de otimização e a obtenção dos estados de mínima energia de um sistema físico

$\rightarrow$ Função Custo $\equiv$ Função Energia

Distribuição de Gibbs: $P(X) \propto \exp[-E(X)/k_B T]$

$\rightarrow$ T alta: Todos os estados são eqüiprováveis

$\rightarrow$ T baixa: Apenas os estados de mínimo custo (estados *fundamentais*) têm probabilidade apreciável

Annealing Físico:

Sólido aquecido além do seu ponto de fusão, e resfriado lentamente

$\rightarrow$ Se o resfriamento é suficientemente lento, obtém-se uma estrutura cristalina livre de imperfeições (estado de baixa energia)

Annealing Simulado:

Algoritmo de Metropolis empregado numa seqüência de temperaturas decrescentes, para gerar soluções de um problema de otimização

O processo começa com um valor T elevado, e, a cada T, geram-se soluções até que o equilíbrio àquela temperatura seja alcançado

A temperatura é então rebaixada, e o processo prossegue até o 'congelamento' (i.e., não se obtém mais uma diminuição de custo)

A seqüência de temperaturas empregada, juntamente com o número de iterações a cada temperatura, constitui uma *prescrição de annealing*, que deve ser definida empiricamente

• Exemplo de Aplicação: *Problema do Caixeiro-Viajante*

$\rightarrow$ Planejar a rota de menor custo entre N cidades, dado o custo de viagem entre quaisquer duas delas (*Problema NP-Completo*)

Função Custo: $E(X) = \Sigma_{i=1}^{N-1} d(X_i, X_{i+1})$, onde

$X = \{X_1, .., X_N\} \equiv$ Permutação das N cidades

$d(X_i, X_{i+1}) \equiv$ Distância entre as cidades X_i e X_{i+1} na rota

Partindo de uma rota inicial arbitrária e de uma temperatura elevada, visitamos as cidades em seqüência e propomos reversões locais de percurso: $\{X_{i-1}, X_i, X_{i+1}\} \rightarrow \{X_{i+1}, X_i, X_{i-1}\}$

As modificações de percurso são aceitas de acordo com o algoritmo de *Annealing* Simulado

$\rightarrow$ Efeito de escala: Para T alta, os trechos de larga escala se definem; só quando $T \approx 0$ a estrutura de curta escala se estabiliza

Algoritmo de *Annealing* Simulado:

Escolha um estado arbitrário s

Escolha uma temperatura inicial suficientemente elevada T

while $T > 0$ **do**

Faça uma mudança randômica de estado $r \leftarrow R(s)$

$\Delta E \leftarrow E(r) - E(s)$

; Aceite estados de energia mais baixa

if $\Delta E \leq 0$ **then** $s \leftarrow r$

; Aceite estados de energia mais alta com probabilidade P

else

$P \leftarrow \exp(-\Delta E/T)$

$x \leftarrow$ número randômico em $[0, 1]$

if $x < P$ **then** $s \leftarrow r$

if Não há decréscimo significativo em E por várias iterações

then Baixe a temperatura T

Implementação do *Annealing*

$\rightarrow$ Conjunto de Decisões *Genéricas* e *Específicas do Problema*

• Decisões Genéricas: Prescrição de *Annealing*

$\rightarrow$ Temperatura Inicial, Temperatura Final, Taxa de Resfriamento, Condições de Parada

Temperatura Inicial: Deve ser alta o bastante para permitir movimentos livres entre soluções vizinhas

$\rightarrow$ Pode ser escolhida a partir do conhecimento da variação média de custo entre soluções vizinhas

$\rightarrow$ Alternativamente, pode ser obtida por *simulação*, e.g., fixando-se uma *taxa de aceitação mínima* de movimentos

Taxa de Resfriamento: O equilíbrio térmico deve ser aproximado a cada temperatura (em teoria, o número de iterações requerido cresce exponencialmente com o tamanho do problema)

Prescrições Comuns:

a) Resfriamento Geométrico: $T = aT$, $a < 1$

$\rightarrow$ Resfriamento lento $(0.8 < a < 0.99)$

$\rightarrow$ O número de iterações a cada T pode ser variável, e.g., ligado a uma taxa fixa de aceitação de movimentos: alto $T \rightarrow$ poucas iterações

b) $T = T/(1 + \beta T)$, com β pequeno

$\rightarrow$ Resfriamento rápido: uma só iteração por temperatura

c) Prescrição de Hajek: $T = c/[\log(1 + k)]$, $k \equiv$ iteração

$\rightarrow$ Resfriamento muito lento

$\rightarrow$ Para c da ordem da *profundidade* do mínimo local mais profundo, a convergência do algoritmo está garantida se $k \rightarrow \infty$

Temperatura Final: Em teoria, a temperatura final deve ser *zero*. Na prática, é suficiente chegar a uma temperatura *próxima a zero*, devido à precisão limitada da implementação computacional

$\rightarrow$ Especifica-se um número máximo de iterações do algoritmo (garantindo que ele atinja baixas temperaturas)

$\rightarrow$ Alternativamente, identifica-se o *congelamento* do processo, quando a taxa de aceitação de movimentos cai abaixo de um valor predeterminado

Regra Geral: *Os parâmetros mais adequados para uma dada aplicação do algoritmo só podem ser estabelecidos por experimentação*

• Decisões Específicas do Problema

Espaço de Soluções, Estrutura de Vizinhança, Função Custo

→ Do resultado de Hajek: Espaço de soluções com topografia acidentada deve ser evitado; espaço com grandes áreas planas também, já que prejudica a evolução do algoritmo

→ Estrutura de vizinhança deve garantir que qualquer solução seja alcançável a partir de qualquer outra, para garantir convergência

→ Soluções não-plausíveis devem ser preferencialmente penalizadas, ao invés de mantidas fora do espaço de soluções, para garantir condição acima, e também para facilitar o cômputo da função custo

Exemplo: *Problema de Coloração de Grafos*

Grafo: Conjunto de pontos (*nós*), alguns deles conectados entre si por linhas (*arestas*)

Problema de coloração dos nós: Alocar uma cor para cada nó, de forma que nós adjacentes não recebam a mesma cor e que o número mínimo de cores seja empregado

(Em certas aplicações, deseja-se obter uma coloração plausível para um dado número de cores)

Espaço de Soluções Natural: Conjunto de Colorações Plausíveis

$\rightarrow$ Partições dos nós em subconjuntos tais que não haja arestas entre dois nós quaisquer do mesmo subconjunto

Vizinhança: Pode ser obtida trocando alguns nós entre dois subconjuntos

$\rightarrow$ Problema: Isso deve ser feito mantendo-se a plausibilidade (não se pode introduzir arestas nos subconjuntos)

$\rightarrow$ Possível solução: Usar *cadeias de Kempe*

Cadeias de Kempe: O grafo definido pelos nós de quaisquer dois subconjuntos contém um número de componentes desconectadas; estas são as cadeias de Kempe

A plausibilidade das soluções pode ser mantida trocando-se as cores dos nós numa mesma cadeia de Kempe

$\rightarrow$ Problema: Calcular repetidamente as cadeias de Kempe pode ser custoso

$\rightarrow$ Solução alternativa: Relaxar a definição de plausibilidade, aumentando o espaço de soluções e incluindo uma *penalidade* na função custo, para tornar caras (i.e., improváveis) as soluções implausíveis

Função Custo Natural: Número de subconjuntos em cada partição

→ Problema: Platôs na função custo: O custo não muda, a não ser que algum dos subconjuntos torne-se vazio

(O termo de penalidade introduz certa variação na função custo, mas não serve para guiar o *annealing* na direção das melhores soluções: partições com muitos subconjuntos grandes e alguns subconjuntos quase vazios são melhores que partições uniformes)

→ Possível solução: Termo que encoraja subconjuntos grandes

e.g.,

$$E = -\sum_{i=1}^{k} C_i^2 + 2\sum_{i=1}^{k} C_i E_i$$

onde C_i é o número de nós e E_i é o número de arestas no subconjunto i, e k é o número (desconhecido) de subconjuntos

Propostas de Modificação do *Annealing*

- Probabilidade de Aceitação

Cálculo de $\exp(-\Delta E/T)$ é custoso (1/3 do tempo computacional, em uma aplicação de partição de grafo)

$\rightarrow$ Alternativas:

i) $P(\Delta E) = 1 - \Delta E/T$

ii) Tabela para cálculo de $P(\Delta E)$ em valores fixos de $\Delta E/T$

- Resfriamento

Fase intermediária do *annealing* é a mais importante (fase de alta temperatura deve ser curta)

$\rightarrow$ Possibilidades:

i) Reduzir T após um número fixo de aceitações de movimentos, antes que o número máximo de iterações seja atingido

ii) Iniciar o *annealing* numa temperatura para a qual a taxa de aceitação já seja pequena

iii) *Annealing* a Temperatura Fixa: Executar o algoritmo inicialmente com 1 iteração a cada temperatura; determinar o T para o qual o menor valor de custo é obtido; prosseguir com o *annealing* àquela temperatura fixa

iv) Usar *reaquecimento*:

e.g., a cada aceitação: $T \to T/(1+\alpha T)$

a cada rejeição: $T \to T/(1-\gamma T)$

• Vizinhanças

A vizinhança pode ser ajustada ao longo da execução

e.g., Quando um termo de penalidade é usado para garantir plausibilidade, os movimentos podem ser restritos às variáveis que contribuem para a sua quebra

• Amostragem da Vizinhança

Annealing Padrão: Vizinhança é amostrada aleatoriamente

$\to$ Próximo a uma solução ótima, como a maioria das soluções vizinhas são de maior custo, pode acontecer de um movimento pior ser aceito *antes de o ótimo ser atingido*; assim, o algoritmo pode se afastar do ótimo

$\to$ Possível solução: Amostragem Cíclica - Todas as soluções vizinhas são testadas uma vez, antes que alguma possa ser reconsiderada

(A ordem das visitas no ciclo deve ser randomizada, de forma a manter a aleatoriedade global do algoritmo)

- Função Custo

Pode ser difícil determinar o peso de um termo de penalização (plausibilidade)

→ Possível solução: Pode-se resolver o problema iterativamente, procurando-se soluções plausíveis para valores progressivamente menores da função custo verdadeira (sem penalidade)

e.g., Na coloração de grafos, encontrar soluções plausíveis usando k cores, para valores decrescentes de k