

6. kapitola

Simulované žihanie

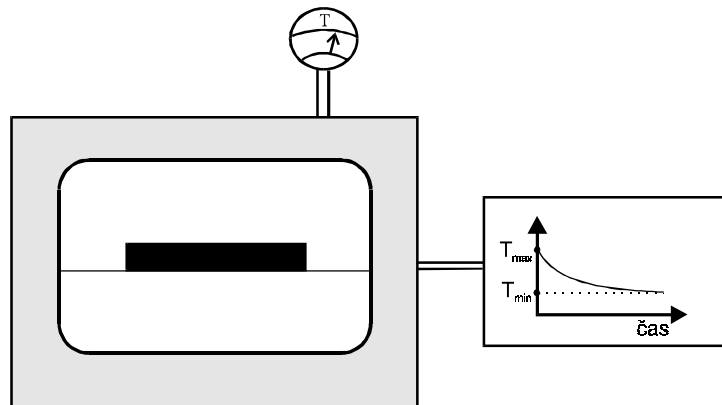
6.1 Úvodné poznámky

Metóda simulovaného žihania [1,2] (Simulated Annealing - SA) patrí medzi tie stochastické optimalizačné algoritmy, ktoré, ako napovedá už ich názov, majú základ vo fyzike (na rozdiel od iných stochastických optimalizačných algoritmov, ktoré majú väčšinou svoj základ v biológii). Algoritmus simulovaného žihania je založený na analógii medzi žiháním tuhých telies a optimalizačným problémom. Počiatkom 80-tych rokov Kirkpatrick, Gelatt a Vecchi [3] (Watson Research Center of the IBM, USA) a nezávisle Černý [4] (pôsobiaci na Katedre teoretickej fyziky MFF UK v Bratislave) dostali geniálny nápad, že problém hľadania globálneho minima môže byť realizovaný podobným spôsobom ako žihanie tuhého telesa. Tento nový prístup k hľadaniu globálneho minima funkcie bol nazvaný *simulované žihanie*.

Vo fyzike žihanie označuje taký proces v ktorom je teleso umiestnené do pece vyhriatej na vysokú teplotu a postupným pomalým znižovaním teploty (pozri obr. 6.1) sa odstraňujú vnútorné defekty telesa. Pri vysokej teplote častice telesa (atómy alebo molekuly) sú náhodne usporiadané v priestore, takže teleso je roztopené. Potom sa teplota postupne znižuje, všetky častice telesa majú možnosť dostať sa do rovnovážnej polohy, čiže energia telesa sa znižuje. Budeme predpokladať, že proces ochladzovania je dostatočne pomalý, potom pre každú teplotu T žihané teleso je v tepelnej rovnováhe, ktorá je popísaná Boltzmannovským rozdelením pravdepodobnosti toho, že pri teplote T je teleso v stave i s energiou E_i

$$w_T(E_i) = \frac{1}{Q(T)} \exp\left(-\frac{E_i}{kT}\right) \quad (6.1a)$$

kde k je Boltzmannova konštanta a $Q(T)$ je normalizačný faktor nazývaný *partičná funkcia*



Obrázok 6.1. Znázornenie fyzikálnej realizácie žihania. Teleso sa vloží do pece (Tavý blok), ktorá je vyhriata na vysokú teplotu T_{\max} . Teplota sa programovacím zariadením (pravý blok) pomaly znižuje na teplotu T_{\min} . Týmto spôsobom sa odstraňujú štruktúrne defekty vyskytujúce sa v telese

$$Q(T) = \sum_i \exp\left(-\frac{E_i}{kT}\right) \quad (6.1b)$$

kde sumácia obsahuje všetky stavy i telesa. Ak teplota T klesá, potom Boltzmannovská distribúcia uprednostňuje stavy s menšou energiou. V prípade, že teplota sa blíži nule, stav s minimálnou energiou má nenulovú (jednotkovú) pravdepodobnosť. Žiaľ, ako je dobre známe, ak je ochladzovanie telesa príliš rýchle (telesu nie je umožnené získať tepelnú rovnováhu pre každú teplotu T), defekty v telese môžu zamrznúť za vzniku metastabilných štruktúr, ktoré sú vzdialené od mriežkovej štruktúry s najnižšou energiou.

Ako simulovať fyzikálnu evolúciu systému (napr. mnohočasticového kryštálu) smerom k tepelnej rovnováhe pre danú teplotu T ? Metropolis a spol. [5] navrhli Monte Carlo metódu, ktorá simuluje evolúciu systému tak, že generuje postupnosť stavov systému nasledujúcim spôsobom: Nech je daný aktuálny stav systému (určený polohou častíc telesa), potom malá náhodná porucha je generovaná tak, že častice sú "jemne" posunuté. Poznamenajme, že táto porucha musí byť "symetrická", t.j. pravdepodobnosť toho, že malou poruchou stav A sa zmení na stav B musí byť rovnaká ako zmene malou poruchou stavu B na stav A. Ak rozdiel $\Delta E = E_{\text{perturbed}} - E_{\text{current}}$ medzi porušeným stavom a aktuálnym stavom je negatívny ($E_{\text{perturbed}} < E_{\text{current}}$), potom proces pokračuje s novým porušeným stavom. V opačnom prípade, ak $\Delta E \geq 0$, pravdepodobnosť akceptovania porušeného stavu, $Pr(\text{perturbed} \leftarrow \text{current})$, je určená exponenciálou $\exp(-\Delta E / kT)$ (pozri obr. 6.2)

$$Pr(\text{perturbed} \leftarrow \text{current}) = \min(1, \exp(-\Delta E / kT)) \quad (6.2)$$

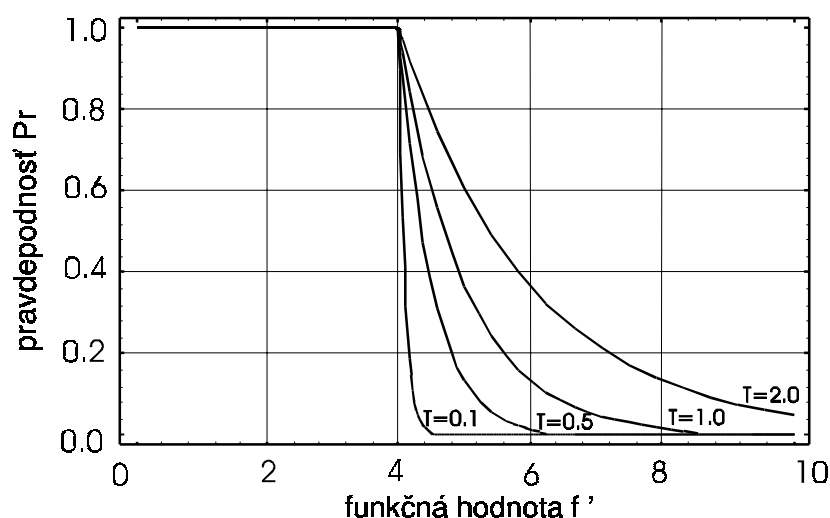
Toto pravidlo akceptovania porušeného stavu sa nazýva *Metropolisovo kritérium*. Podľa tohto kritéria, aplikovaním veľkého počtu porúch a akceptovaním ich do ďalšieho procesu s pravdepodobnosťou (6.2), dostaneme systém v tepelnej rovnováhe, pričom distribúcia pravdepodobnosti rozloženia stavov sa asymptoticky blíži k Boltzmannovej distribúcii (6.1). Táto forma Monte Carlo metódy sa nazýva v štatistickej fyzike *Metropolisov algoritmus* [5]. K tomu, aby sme formalizovali Metropolisov algoritmus (pozri Algoritmus 6.1) zavedieme nasledujúci formalizmus (ktorý je vhodný pre aplikáciu simulovaného žihania na minimalizáciu funkcií): Stav systému nech je určený *stavovou premenou* \mathbf{x} (vo všeobecnosti vektor obsahujúci mnoho nezávislých reálnych premenných) a energiou $f(\mathbf{x})$. Proces poruchy stavu \mathbf{x} na iný stav \mathbf{x}' je formálne reprezentovaný stochastickým operátorom, $\mathbf{x}' = O_{\text{pertur}}(\mathbf{x})$. Stochastický charakter tohto operátora spočíva v náhodnej zmene stavu \mathbf{x} na stav \mathbf{x}' .

Ako už bolo poznamenané, Metropolisov algoritmus [5] produkuje distribúciu pravdepodobnosti stavov, ktorá sa asymptoticky blíži Boltzmannovskej distribúcii (6.1)

$$w_T(x) = \frac{1}{Q(T)} \exp\left(-\frac{f(x)}{T}\right) \quad (6.3a)$$

$$Q(T) = \sum_x \exp\left(-\frac{f(x)}{T}\right) \quad (6.3b)$$

kde sumácia obsahuje všetky stavy \mathbf{x} . Pre jednoduchosť, Boltzmannova konštanta k bola teraz vynechaná (pozri rovnice (6.1a-b), formálne, táto konštanta bola presunutá do teploty T).



Obrázok 6.1. Priebeh Metropolisovho kritéria $Pr = \min(1, \exp(-(f'-f)/T))$ pre rôzne teploty T , kde f je fixná funkčná hodnota ($f=4$) a f' je nezávislá premenná braná z intervalu $[0,10]$. Pre klesajúce hodnoty teploty T a pre $f' > f$, pravdepodobnosť $Pr \rightarrow 0$ ak $T \rightarrow 0$.

Metropolisov algoritmus môže byť použitý pre počítačovú simuláciu žihania. V tomto prístupe je simulované žihanie chápané ako postupnosť Metropolisových algoritmov realizovaných pre postupnosť vhodne sa znižujúcich teplôt, pričom výstupný stav \mathbf{x}_{out} z posledného Metropolisovho algoritmu slúži ako vstupný stav \mathbf{x}_{in} pre nasledujúci Metropolisov algoritmus. Táto procedúra je inicializovaná tak, že teplota T sa položí rovná maximálnej teplote T_{max} a Metropolisov algoritmus sa aplikuje tak dlho (k_{max} krát, kde k_{max} je parameter Metropolisovho algoritmu), až je dosiahnutá tepelná rovnováha. Potom sa teplota zníži (napr. pravidlom $T := \alpha \cdot T$, kde $0 << \alpha < 1$) a znovu sa aplikuje Metropolisov algoritmus, pričom sa predpokladá, že sa dosiahne tepelná rovnováha. Algoritmus je ukončený vtedy, keď teplota T dosiahla minimálnu hodnotu T_{min} , pre ktorú už žiadny stav s vyššou funkcionálnou hodnotou nie je akceptovaný (skoro nulová pravdepodobnosť akceptovania). "Zamrznutý" stav \mathbf{x}_{out} sa potom chápe ako výsledné riešenie, pozri

```

procedure Metropolis_algorithm(input:  $\mathbf{x}_{ini}$ ,  $k_{max}$ ,  $T$ ; output:  $\mathbf{x}_{out}$ ) ;
begin  $k := 0$ ;  $\mathbf{x} := \mathbf{x}_{ini}$ ;
      while  $k < k_{max}$  do
        begin  $k := k + 1$ ;
           $\mathbf{x}' := O_{pertur}(\mathbf{x})$ ;
           $Pr := \min(1, \exp(-(f(\mathbf{x}') - f(\mathbf{x})) / T))$ ;
          if random <  $Pr$  then  $\mathbf{x} := \mathbf{x}'$ ;
        end;
       $\mathbf{x}_{out} := \mathbf{x}$ ;
end;

```

Algoritmus 6.1. Implementácia Metropolisovho algoritmu. Procedúra je inicializovaná tak, že počiatočný stav sa položí rovný vstupnému stavu \mathbf{x}_{ini} , opakuje sa k_{max} krát (toto číslo musíme byť dostatočne veľké ktomu, aby sa dosiahla tepelná rovnováha), symbol O_{pertur} modifikuje aktuálny stav \mathbf{x} na \mathbf{x}' . Akceptovanie nového stavu je riešené pomocou Metropolisovho kritéria realizovaného pre teplotu T . Po skončení Metropolisovho algoritmu, výstupný stav \mathbf{x}_{out} je posledný stav \mathbf{x} .

algoritmus 6.2.

```

procedure Simulated_annealing(input:  $T_{min}$ ,  $T_{max}$ ,  $k_{max}$ ,  $\alpha$ ; output:  $\mathbf{x}_{opt}$ ) ;
begin  $\mathbf{x}_{ini} :=$  randomly generated state vector;
       $T := T_{max}$ ;
      while  $T > T_{min}$  do
        begin Metropolis_algorithm( $\mathbf{x}_{ini}$ ,  $\mathbf{x}_{out}$ ,  $k_{max}$ ,  $T$ ) ;
           $\mathbf{x}_{ini} := \mathbf{x}_{out}$ ;
           $T := \alpha * T$ ;
        end;
       $\mathbf{x}_{opt} := \mathbf{x}_{out}$ ;
end;

```

Algoritmus 6.2. Implementácia simulovaného žihania, vstupné parametre sú T_{min} , T_{max} , k_{max} , α , výstupný parameter je \mathbf{x}_{opt} . Algoritmus je inicializovaný náhodným generovaním počiatočného stavu \mathbf{x}_{ini} a maximálnou teplotou T_{max} . While-cyklus sa opakuje tak dlho, pokiaľ platí $T > T_{min}$, teplota T sa znižuje pomocou parametra α vzťahom $T := \alpha * T$. Po ukončení while-cyklu výsledný stav \mathbf{x}_{out} je považovaný za výsledné riešenie označené \mathbf{x}_{opt} .

Príklad 6.1. Implementujte algoritmus simulovaného žihania, pozri algoritmus 6.2. Algoritmus špecifikujte pre funkciu n -premenných, ktoré su binárne reprezentované. Operátor mutácie O_{mut} je potom totožný s podobným operátorom z genetického algoritmu, pozri (1.19) a algoritmus 1.2.

Jeden zo základných problémov simulovaného žihania je špecifikácia teplôt T_{max} , T_{min} a pravidla pre znižovanie teploty T . V literatúre boli navrhnuté rôzne schémy, ako určiť tieto parametre simulovaného žihania [1,2]. Pretože náplňou tejto kapitoly nie je študovať detailnú teóriu simulovaného žihania, budeme používať najjednoduchší prístup pre ochladzovanie (multiplikatívny prístup pomocou parametra α) a počiatočná teplota je zvolená tak, aby približne 50% stavov porušených stavov bolo akceptovaných Metropolisovým algoritmom.

Hlavný cieľ simulovaného žihania je hľadať globálne riešenie optimalizačného problému

$$\mathbf{x}_{opt} = \arg \min_{\mathbf{x} \in D} f(\mathbf{x}) \quad (6.4)$$

kde $f(\mathbf{x})$ je reálna funkcia definovaná nad doménou D (ktorá je obvykle konečná a diskrétna), a \mathbf{x}_{opt} je hodnota vektora nezávislých premenných, ktorá odpovedá globálnemu minimu funkcie $f(\mathbf{x})$ nad doménou D . Premenná

\mathbf{x} je teraz chápaná ako stavový vektor hypotetického fyzikálneho systému a hodnota funkcie $f(\mathbf{x})$ vyjadruje jeho "energiu". Potom, vyššie uvedený optimalizačný problém (6.1) môže byť úspešne študovaný metódou simulovaného žihania. Parameter "teplota" bude v tomto prístupe hrať úlohu riadiaceho parametra metódy.

6.2. Vzťah k štatistickej fyzike

V predchádzajúcej časti tejto kapitoly bola naznačená tesná analógia medzi žiháním tuhého telesa (vo všeobecnosti, fyzikálneho systému obsahujúceho mnoho častíc, ktoré medzi sebou interagujú) a riešením optimalizačného problému (6.4). Fyzikálny proces žihania môže byť úspešne modelovaný na počítači pomocou Metropolisovho algoritmu. Tento algoritmus je založený na štatistickej fyzike, ktorá môže byť chápaná ako centrálna disciplína pre fyziku tuhej fáze. Ukážeme, že existuje veľmi tesný vzťah medzi optimalizačným problémom (6.4) a štatistickou fyzikou. Táto aplikácia simulovaného žihania nie je zaujímavá len ako určitá analógia, ale tiež aj ako vhodný rámec pre štúdium konvergenčných vlastností simulovaného žihania.

Základný predpoklad štatistickej fyziky [6] je, že fyzika mnohočasticových systémov je kompatibilná so štatistickými súbormi a že časové stredné hodnoty mechanických veličín systému v rámci mikroskopickej rovnováhy sú rovné odpovedajúcim stredným hodnotám nad súbormi. V štatistickej fyzike boli definované vhodné veličiny popisujúce fyzikálne vlastnosti systému, a tiež bolo ukázané, ako počítať tieto veličiny pomocou rovnovážneho rozdelenia stavov systému. Ukazuje sa, že v tepelnej rovnováhe rozdelenie pravdepodobnosti je určené Boltzmannovým rozdelením (6.1a-b). Vzťah medzi štatistickou fyzikou a optimalizačným problémom (6.4) môže byť teraz študovaný vo veľmi špecifickej rovine. Predpokladajme, že máme hypotetický fyzikálny systém v tepelnej rovnováhe, ktorého vnútorné stavy \mathbf{x} (premenne z oblasti D) majú distribúciu podľa vzťahu (6.3a-b) (formálne ekvivalentný s (6.1a-b)), potom pre tento optimalizačný problém môžeme definovať rôzne fyzikálne veličiny podobným spôsobom ako v štatistickej fyzike.

Uvažujme funkciu f definovanú nad diskretnou a konečnou doménou D

$$f: D \rightarrow A \subset R_+ \quad (6.5)$$

kde A je doména (diskrétne a konečná podmnožina R_+ obsahujúca nezáporné reálne čísla) funkčných hodnôt f . Rozdelenie pravdepodobnosti stavu \mathbf{x} (t.j. premennej \mathbf{x} z D), ktoré je výsledkom Metropolisovho algoritmu realizovaného pre teplotu T , je určené vzťahmi (6.3a-b). Pre naše ďalšie úvahy bude vhodné poznať hustotu funkčných hodnôt $y=f(\mathbf{x})$, ktorá je určená takto

$$w_T(y) = \frac{|D(y)|}{Q(T)} \exp\left(-\frac{y}{T}\right) \quad (6.6a)$$

$$Q(T) = \sum_{y \in A} |D(y)| \exp\left(-\frac{y}{T}\right) \quad (6.6b)$$

kde $D(y) \subset D$ je množina obsahujúca všetky $\mathbf{x} \in D$ pre ktoré platí $y=f(\mathbf{x})$

$$D(y) = \{\mathbf{x} \in D; y = f(\mathbf{x})\} \quad (6.6c)$$

a $|D(y)|$ je jej kardinalita. Je jasné, že veličina $Q(T)$ určená (6.6b) je identická s jej analógom (6.3b). Nasledujúce "makroskopické" veličiny sú definované

(1) *Stredná hodnota* funkcie $f(\mathbf{x})$

$$\langle f \rangle_T = \sum_{y \in A} y w_T(y) \quad (6.7)$$

(2) *Stredná hodnota* funkcie $f^2(\mathbf{x})$

$$\langle f^2 \rangle_T = \sum_{y \in A} y^2 w_T(y) \quad (6.8)$$

(3) *Disperzia* funkcie $f(\mathbf{x})$

$$\sigma^2(T) = \sum_{y \in A} w_T(y) (\langle f \rangle_T - y)^2 = \langle f^2 \rangle_T - \langle f \rangle_T^2 \quad (6.9)$$

(3) *Entropia*

$$S(T) = - \sum_{y \in A} w_T(y) \ln \left(\frac{w_T(y)}{|D(y)|} \right) \quad (6.10)$$

(4) *Špecifické teplo*

$$C(T) = \frac{\partial}{\partial T} \langle f \rangle_T = \frac{\sigma^2(T)}{T^2} = T \frac{\partial S(T)}{\partial T} \quad (6.11)$$

Táto formula môže byť jednoducho odvodená derivovaním buď (7) alebo (10) vzhľadom k teplote T .

Hustota funkčných hodnôt (6a-b) vyhovuje týmto asymptotickým podmienkam

$$\lim_{T \rightarrow \infty} w_T(y) = w_\infty(y) = \frac{1}{|D|} |D(y)| \quad (6.12a)$$

$$\lim_{T \rightarrow 0} w_T(y) = w_0(y) = \delta(y, y_{opt}) = \begin{cases} 1 & (\text{pre } y = y_{opt}) \\ 0 & (\text{pre } y \neq y_{opt}) \end{cases} \quad (6.12b)$$

kde $y_{opt} = f(x_{opt})$ je optimálna hodnota funkcie $f(x)$ odpovedajúca bodu určenému (4) ako globálne minimum. Druhý asymptotický vzťah (6.12b) je veľmi dôležitý pre metódu simulovaného žihania; podľa tejto formule ak teplota T sa blíži k nule (cez rovnovážne stavy), potom *pravdepodobnosť výskytu stavu odpovedajúceho globálnemu minimu je jednotková*.

Parciálna derivácia $\langle f \rangle_T$ vzhľadom k teplote T je určená vzťahom (6.11), potom

$$\frac{\partial \langle f \rangle_T}{\partial T} = \frac{\sigma^2(T)}{T^2} > 0 \quad (6.13)$$

To znamená, že stredná hodnota $\langle f \rangle_T$ klesá s ak teplota T taktiež klesá, pomocou (6.12a-b) dostaneme

$$\lim_{T \rightarrow \infty} \langle f \rangle_T = \langle f \rangle_\infty = \frac{1}{|D|} \sum_{y \in A} |D(y)| y \quad (6.14a)$$

$$\lim_{T \rightarrow 0} \langle f \rangle_T = \langle f \rangle_0 = y_{opt} \quad (6.14b)$$

Stredná hodnota $\langle f^2 \rangle_T$ vyhovuje podobným asymptotickým podmienkam

$$\lim_{T \rightarrow \infty} \langle f^2 \rangle_T = \langle f^2 \rangle_\infty = \frac{1}{|D|} \sum_{y \in A} |D(y)| y^2 \quad (6.15a)$$

$$\lim_{T \rightarrow 0} \langle f^2 \rangle_T = \langle f^2 \rangle_0 = y_{opt}^2 \quad (6.15b)$$

Pre disperziu (6.9) dostaneme

$$\lim_{T \rightarrow \infty} \sigma^2(T) = \sigma^2(\infty) = \langle f^2 \rangle_\infty - \langle f \rangle_\infty^2 \quad (6.16a)$$

$$\lim_{T \rightarrow 0} \sigma^2(T) = \sigma^2(0) = \langle f^2 \rangle_0 - \langle f \rangle_0^2 = y_{opt}^2 - (y_{opt})^2 = 0 \quad (6.16b)$$

Entropia (6.10) vyhovuje dobre známym dvom asymptotickým podmienkam

$$\lim_{T \rightarrow \infty} S(T) = S(\infty) = \ln |D| \quad (6.17a)$$

$$\begin{aligned} \lim_{T \rightarrow 0} S(T) &= S(0) = \ln |D(y_{opt})| \\ &= \ln 1 = 0 \quad (\text{pre } |D(y_{opt})| = 1) \end{aligned} \quad (6.17b)$$

Dôkaz týchto podmienok je jednoducho realizovaný substitúciou (6.6) do (6.10) a potom asymptotickou zmenou teploty buď $T \rightarrow \infty$ alebo $T \rightarrow 0$. Druhá vlastnosť (6.17b) vo fyzike je známa ako tretí zákon termodynamický. V prípade simulovaného žihania, entropia môže byť interpretovaná ako *kvantitatívna miera stupňa optimality*.

V dôsledku platnosti (6.11) parciálna derivácia entropie vzhľadom k teplote T je určená takto

$$\frac{\partial S(T)}{\partial T} = \frac{\sigma^2(T)}{T^3} > 0 \quad (6.18)$$

Potom s klesajúcou teplotou T (cez rovnovážne stavy) entropia S taktiež musí klesať, $S(T) \rightarrow 0$ pre $T \rightarrow 0$ (pozri rov. (6.17b)). V okolí bodu $T=0$ entropia $S(T)$ môže byť aproximovaná mocninným rozvojom teploty T

$$S(T) = A_0 + A_1 T + A_2 T^2 + \dots \quad (6.19)$$

kde $A_0 = \ln |D(y_{opt})|$, a $A_1 > 0$. Potom tepelná kapacita $C(T)$ v okolí bodu $T=0$ (pozri rov. (6.11)) je aproximovaná takto

$$C(T) = T \frac{\partial S(T)}{\partial T} = T (A_1 + 2A_2 T + \dots) \quad (6.20)$$

V dôsledku platnosti tohto vzťahu a asymptotickej vlastnosti (6.16a) disperzie, tepelná kapacita $C(T)$ spĺňa tieto dve asymptotické podmienky

$$\lim_{T \rightarrow \infty} C(T) = \lim_{T \rightarrow 0} C(T) = 0 \quad (6.21)$$

To znamená, že $C(T)$ musí mať pre $0 < T < \infty$ aspoň jedno maximum a ak teplota asymptoticky sa blíži k nule alebo k nekonečnu, tepelná kapacita $C(T)$ sa blíži k nule.

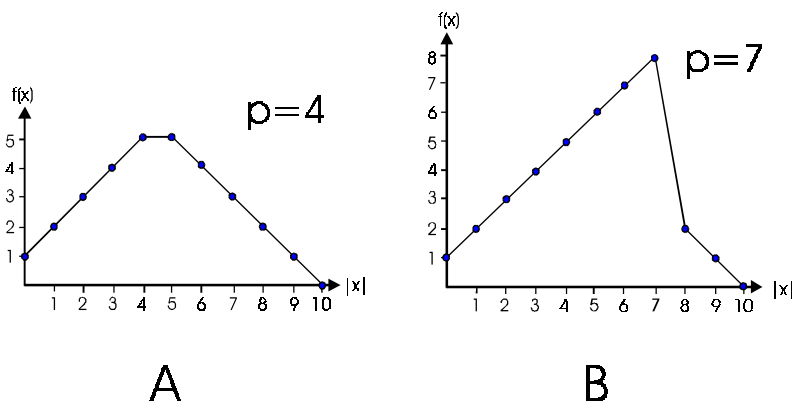


Figure 6.2. Priebeh funkcie (6.22) pre $N=10$ a $p=4, 7$. Pre $p=7$ funkcia obsahuje "dlhé" úpätie za ktorým je oblasť prudkého spádu do globálneho miníuma.

6.3. Modelové výpočty

Nech zobrazenie $f: D = \{0,1\}^N \rightarrow A \subset \mathbb{R}_+$, s definičným oborom obsahujúcim binárne vektory dĺžky N , je definované takto

$$f(x) = f(x_1, x_2, \dots, x_N) = \begin{cases} |x| + 1 & (\text{pre } |x| \leq p) \\ N - |x| & (\text{pre } |x| > p) \end{cases} \quad (6.22)$$

kde $|x| = \sum x_i$ je L_1 norma vektora $x \in \{0,1\}^N$, $0 \leq |x| \leq N$. Parameter p je ohraničený podmienkou $0 \leq p \leq N$. Pre $p < N$ táto funkcia má dve minima, prvé minimum je $f(x)=1$, pre $|x|=0$, druhé (globálne) minimum je $f(x)=0$, pre $|x|=N$. Pre $p=N$, funkcia má len jedno (t.j. globálne) minimum $f(x)=1$, pre $|x|=0$. Množina A funkčných hodnôt obsahuje nezáporné celé čísla, pričom $A = \{0, 1, 2, \dots, f_{max}\}$, kde $f_{max} = \max(p+1, N-p-1)$. Priebeh tejto funkcie pre $N=10$ a pre $p=4$ a $p=7$ je znázornený na obrázku 6.2. Parameter p určuje stupeň *deceptívnosti* (falošnosti) tejto funkcie. Ak je p o trochu menšie ako N , potom funkcia obsahuje dlhé "úpätie" začínajúce v $|x|=0$ a končiacie v $|x|=p$, za týmto úpätím (pre $p+1 \leq |x| \leq N$) funkcia rýchle klesá k nule (pozri obr. 6.2, diagram B). Táto vlastnosť funkcie $f(x)$ sa nazýva *deceptívnosť*, predstavuje obtiažnu úlohu pre evolučné metódy ak majú hľadať globálne minimum.

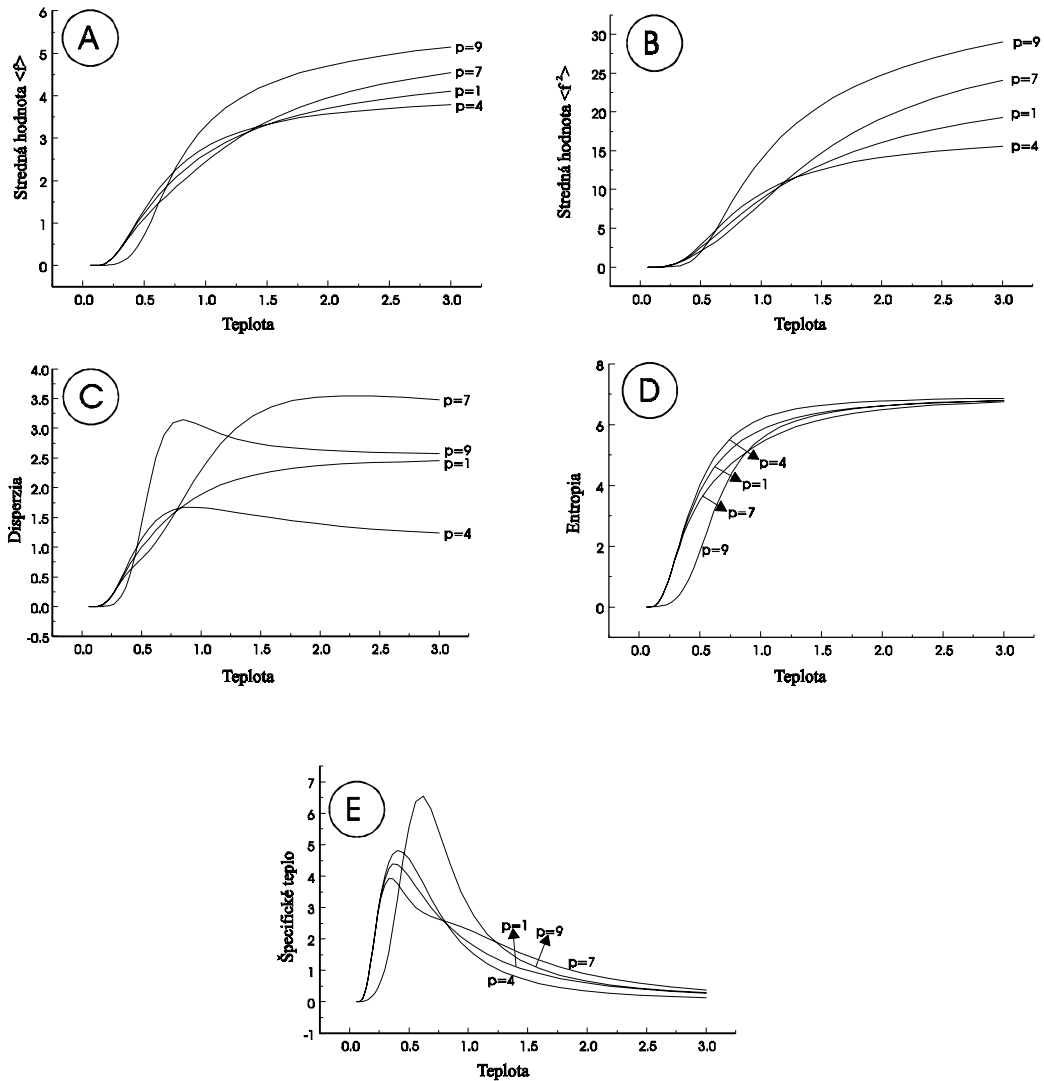
Operátor $O_{perturb}$ z Metropolisovho algoritmu transformujúci binárny vektor $x \in \{0,1\}^N$ na iný binárny vektor $x' \in \{0,1\}^N$ je v tomto prípade realizovaný pomocou operátora mutácie $O_{mut}^{(P_{mut})}$ (terminológia je prevzatá z genetického algoritmu)

$$\begin{aligned} x' = (x'_1, x'_2, \dots, x'_N) &= O_{mut}^{(P_{mut})}(x) \\ &= O_{mut}^{(P_{mut})}(x_1, x_2, \dots, x_N) \end{aligned} \quad (6.23)$$

kde jednotlivé komponenty x' sú určené takto

$$\forall i: x'_i = \begin{cases} 1 - x_i & (\text{ak } random < P_{mut}) \\ x_i & (\text{ostatné prípady}) \end{cases} \quad (6.24)$$

kde P_{mut} je pravdepodobnosť "mutácie" premennej x_i (pre $\forall i$) na jej komplement (ak $random < P_{mut}$) alebo nie (ostatné prípady), a $random$ náhodné číslo z intervalu $[0,1]$ s rovnomerným rozdelením pravdepodobnosti. Pravdepodobnosť P_{mut} je veľmi malá ($0 < P_{mut} \ll 1$), v opačnom prípade by mutačný operátor produkoval stavy, ktoré sú veľmi vzdialené od pôvodného stavu x .



Obrázok 6.3. Priebehy makroskopických veličín vypočítaných teoreticky pomocou formúl (6.7-11) pre parametre $p=1,4,7,9$ (diagramy A - E).

K tomu, aby sme mohli vypočítať makroskopické veličiny definované v predchádzajúcej kapitole 6.2 musíme poznať kardinality množín $D(y)$ pre funkciu (6.19). Tak napr. pre $N=10$ a pre $p=4$ a $p=7$, tieto kardinality sú určené takto

$$\begin{aligned}
 p=4 : & |D(0)|=1, \quad |D(1)|=11, \quad |D(2)|=55, \quad |D(3)|=165, \quad |D(4)|=330, \quad |D(5)|=462 \\
 p=7 : & |D(0)|=1, \quad |D(1)|=11, \quad |D(2)|=55, \quad |D(3)|=45, \quad |D(4)|=120, \quad |D(5)|=210, \\
 & |D(6)|=252, \quad |D(7)|=210, \quad |D(8)|=120
 \end{aligned}$$

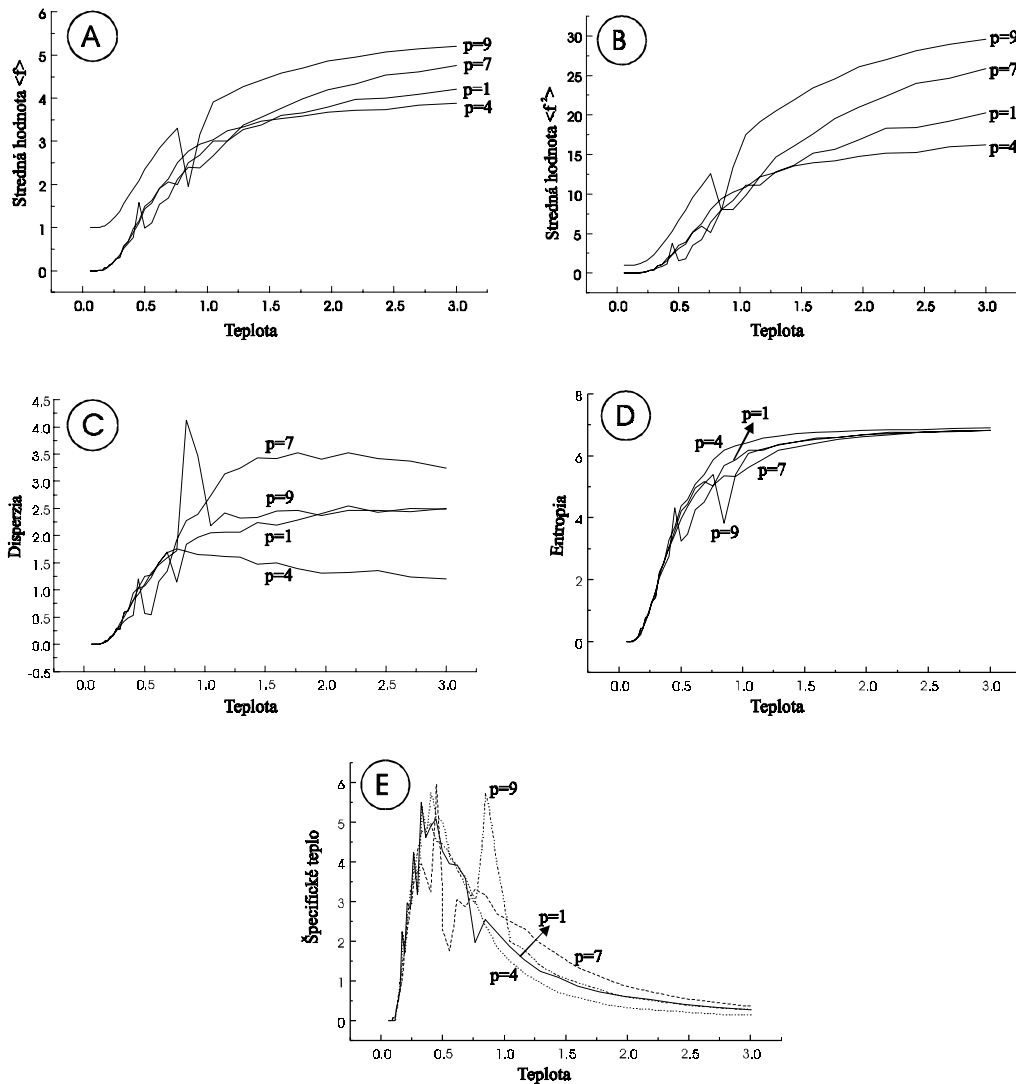
Priebehy makroskopických veličín pre funkciu $f(x)$ špecifikovanú konštantami $N=10$ a $p=1,4,7,9$ sú znázornené na obr. 6.3, diagramy A - E. Ako vyplýva z tohto obrázku, všetky asymptotické vlastnosti, ktoré boli predpovedané v predchádzajúcej časti tejto kapitoly pre makroskopické veličiny sú splnené. Menovite, stredná hodnota $\langle f \rangle_T \rightarrow 0$ pre $T \rightarrow 0$. To znamená, že pre malé teploty stav s nulovou funkčnou hodnotou je dominantný s jednotkovou pravdepodobnosťou výskytu.

Pre simulované žihanie realizované na počítači, výpočet makroskopických veličín môže byť uskutočnený pomocou približných hustôt funkčných hodnôt, ktoré sú aproximované výskytom funkčných hodnôt akceptovanými Metropolisovým algoritmom pre fixnú teplotu T . Nech $\chi_T(y)$ je počet výskytov stavu x , ktorý odpovedá funkčnej hodnote y a ktoré sa vyskytujú v Metropolisovom algoritme pre danú teplotu T . Potom, hustota $w_T(y)$ môže byť približne určená takto

$$w_T(y) \approx \frac{\chi_T(y)}{\sum_{y' \in A} \chi_T(y')} \quad (6.25)$$

Pre k_{max} dostatočne veľké, táto aproximácia by mala poskytovať výsledky, ktoré sú veľmi blízke k presným hodnotám $w_T(y)$. Priebehy makroskopických vlastností (6.7-11) s hustotami aproximovanými vzťahom (6.25) sú znázornené na obr. 6.4. diagramy A - E. Parametre simulovaného žihania použité vo výpočtoch mali tieto hodnoty

$$k_{max}=10^4, P_{mut}=0.1, T_{max}=3, T_{min}=3/50, \text{ and } \alpha=0.95 \quad (6.26)$$



Obrázok 6.4. Priebeh makroskopických veličín vypočítaných pomocou metódy simulovaného žihania, ktorá je špecifikovaná parametrami (6.26). Pre prípad s najväčšou deceptívnosťou ($p=9$), finálny stav x_{opt} odpovedá jednotkovej funkčnej hodnote (lokálne minimum), t.j. metóda nebola schopná poskytnúť globálne minimum.


```

procedure Elitism_Metropolis_algorithm(input : $x_{in}$ ,  $k_{max}$ ,  $T$ ;
                                         output: $x_{opt}$ );
begin  $k:=0$ ;  $x:=x_{in}$ ;  $x_{opt}:=x_{in}$ ;
      while  $k < k_{max}$  do
        begin  $k:=k+1$ ;
           $x' := O_{pertur}(x)$ ;
           $Pr := \min(1, \exp(-(f(x') - f(x))/T))$ ;
          if random  $< Pr$  then
            begin  $x:=x'$ ;
              if  $f(x') < f(x_{opt})$  then  $x_{opt}:=x'$ ;
            end;
          end;
      end;
end;

```

Algorithm 6.3. Implementácia Metropolisovho algoritmu so zahrnutým elitizmom. Ako vstupné parametre sú x_{in} , k_{max} , T . A algoritmus je inicializovaný vstupným vektorom x_{in} , najlepšie riešenie je uložené do x_{opt} , výsledná hodnota tohto vektora slúži ako výstupný parameter procedúry.

Vidíme, že simulované žihanie pre $p=1, 4$, a 7 poskytuje správne výsledky pre strednú hodnotu $\langle f \rangle_T$ (t.j. $\langle f \rangle_T \rightarrow 0$ pre $T \rightarrow 0$), žiaľ pre $p=9$ (funkcia $f(x)$ má najväčší stupeň deceptívnosti) metóda simulovaného žihania nie je schopná poskytnúť korektný výsledok, (dostali sme $\langle f \rangle_T \rightarrow 1$ pre $T \rightarrow 0$). To znamená, že pre $p=9$ simulované žihanie je schopné nájsť len suboptimálne riešenie $x=(0, 0, \dots, 0)$, $|x|=0$, pre ktoré platí $f(x)=1$. Existuje interval kritických teplôt $[0.7, 0.9]$, nad týmto intervalom metóda poskytuje hustoty stavu $\nu=0$, ktoré sú blízke teoretickým hodnotám, avšak vo vnútri intervalu výskyt stavov s $\nu=0$ bol skoro nulový. To znamená, že pre nižšie teploty Metropolisov algoritmus nie je schopný kreovať stavy s $\nu=0$ (v dôsledku širokej bariéry funkcionálnych hodnôt, pozri obr. 6.2, diagram B). Táto skutočnosť je indikovaná napr. vysokými maximami pre disperziu a špecifické teplo, ktoré môžu byť fyzikálne interpretované ako "fázové prechody" medzi konfiguráciami, ktoré sú separované vysokými bariérami funkčných hodnôt. To znamená, že tieto dve makroskopické veličiny môžu byť použité ako vhodný indikátor toho tých tepelných intervalov, kde znižovanie teploty T musí byť veľmi malé, aby sa úspešne prekonal vyššie zmienený "fázový prechod".

Uvedieme niekoľko vysvetľujúcich poznámok k simulovanému žihaniu aplikovanému na deceptívnu funkciu $f(x)$ s hodnotou parametru $p=9$. Ukázali sme, že simulované žihanie s parametrami (6.26) je neschopné nájsť správne riešenie. Vo všeobecnosti, táto neschopnosť môže byť jednoducho prekonaná tak, že pravdepodobnosť P_{mut} sa podstatne zvýši, napr. $P_{mut}=0.5$. Avšak potom simulované žihanie nemôže byť chápané ako evolučná optimalizačná metóda, prehľadávanie priestoru riešení je viac náhodné ako systematické. Simulované žihanie je pre vysoké hodnoty pravdepodobnosti P_{mut} redukované na "slepé" hľadanie. Pre malé dimenzie tento prístup môže byť úspešný, pre veľké dimenzie je tento prístup úplne beznádejný. V nasledujúcej časti tejto kapitoly uvedieme dve rôzne modifikácie štandardnej verzie simulovaného žihania tak, že bude schopný poskytnúť správne riešenie aj pre vysoko deceptívne funkcie a súčasne bude zachovaný evolučný charakter metódy.

6.3.1 Simulované žihanie s elitizmom

V originálnej verzii simulovaného žihania výsledný stav z aplikácie Metropolisovho algoritmu (realizovaného pri teplote T) slúži ako počiatočný stav nasledujúceho Metropolisovho algoritmu (realizovaného pri zníženej teplote αT). Tento základný predpoklad simulovaného žihania bude teraz modifikovaný tak, že Metropolisov algoritmus je inicializovaný najlepším riešením, ktoré bolo získané v priebehu predchádzajúcej histórie metódy (t.j. pre epochy simulované žihanie s vyššími teplotami ako súčasná teplota); takúto jednoduchú modifikáciu simulovaného žihania budeme nazývať *simulované žihanie s elitizmom* [7]. Modifikácia procedúr Metropolis_algorithm a Simulated_annealing, ktoré boli uvedené v predchádzajúcej časti tejto kapitoly je realizovaná v Alogitmoch 6.3 a 6.4.

Vo všeobecnosti, simulované žihanie s elitizmom nespĺňa dôležitú podmienku metódy, a to že konštanta k_{max} je dostatočne veľká a pre diskrétné kroky teploty Metropolisov algoritmus je v tepelnej rovnováhe (t.j. stavy majú distribúciu podľa Boltzmannu (6.3)). Pretože v každom kroku zníženia teploty Metropolisov algoritmus je inicializovaný najlepším riešením, ktoré bolo zaznamenané v celej predchádzajúcej histórii algoritmu, môže nastať taká situácia (menovite pre nižšie teploty), že výsledné riešenie produkované posledným behom Metropolisovho algoritmu odpovedá lokálnemu minimu, ktoré je podstatne odlišné od globálneho minima.

```

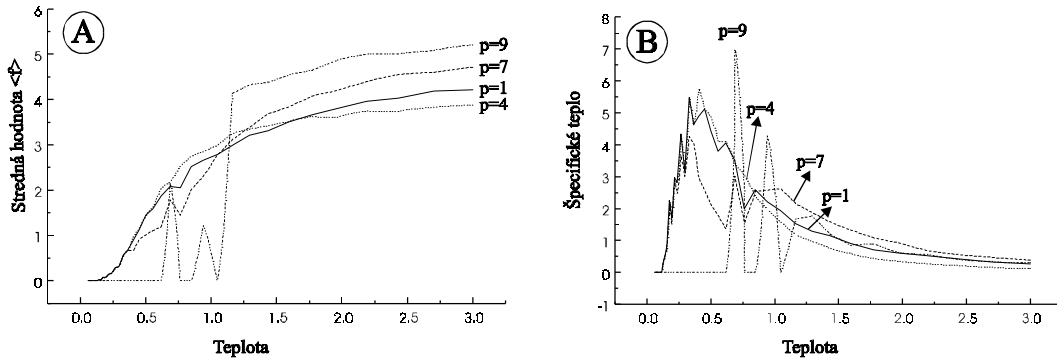
procedure Elitism_Simulated_annealing(input : $T_{min}, T_{max}, k_{max}, \alpha$ ;
                                     output: $x_{opt}$ );
begin  $x_{opt}$  := randomly generated state vector;
       $T := T_{max}$ ;
      while  $T > T_{min}$  do
        begin Forced_Metropolis_algorithm( $x_{opt}, k_{max}, T$ );
           $T := \alpha * T$ ;
        end;
      end;

```

Algorithm 6. 4. Implementácia simulovaného žihania s elitizmom. Vstupné parametre do procedúry sú $T_{min}, T_{max}, k_{max}, \alpha$ a výstupný parameter je x_{opt} . Stav x_{opt} , inicializovaný náhodne, zaznamenáva najlepšie riešenie vyskytujúce sa v celej histórii algoritmu.

Numerické výsledky získané simulovaným žiháním s elitizmom pre funkciu (6.22) špecifikovanú hodnotami $N=10$ a $p=1, 4, 7, 9$ sú znázornené na obr. 6.5 pre strednú hodnotu $\langle f \rangle$ a špecifickú teplotu, diagrami A a B. Vidíme, že pre $p=9$ (najdeceptívnejší tvar funkcie) zmeny na priebehu sú veľmi dramatické. Pre stredné teploty Metropolisov algoritmus bol neschopný udržať nenulovú distribúciu stavu s nulovou funkčnou hodnotou. Pretože najlepšie riešenie s nulovou funkčnou hodnotou je zaznamenané už v prvých etapách simulovaného žihania, Metropolisov algoritmus produkuje pre malé teploty ($T < 0.6$) akceptované stavy s nulovou funkčnou hodnotou, ktoré odpovedajú globálnemu minimu.

Slabšia verzia simulovaného žihania s elitizmom je založená na požiadavke, že Metropolisov algoritmus je inicializovaný najlepším riešením získaným len v predchádzajúcom kroku a nie v celej predchádzajúcej histórii algoritmu. Žiaľ, tento prístup k najdeceptívnejšej forme optimalizovanej funkcie pre $p=9$ nebol úspešný v prekonaní veľkej bariéry so spádom do lokálneho minima (s jednotkovou funkčnou hodnotou)



Obrázok 6. 5. Priebeh strednej hodnoty $\langle f \rangle$ a špecifického tepla, ktoré boli spočítané pomocou simulovaného žihania s elitizmom. Výsledný stav x_{opt} odpovedá pre všetky hodnoty parametra p globálnemu minimu s nulovou funkčnou hodnotou (prípomeňme, že štandardná verzia simulovaného žihania poskytuje pre $p=9$ len lokálne minimum s jednotkovou funkčnou hodnotou).

6.3.2 Paralelné simulované žihanie

Táto verzia simulovaného žihania bola s úspechom použitá pre riešenie komplikovaných kombinatoriálnych grafovo-teoretických problémov [8,9], kde štandardná verzia simulovaného žihania nie je schopná poskytnúť správne globálne riešenie. Základná idea paralelného simulovaného žihania spočíva v súčasnom aplikovaní simulovaného žihania na množinu P stavov x, x', x'', \dots ktoré sú "synchronizované" tak, že Metropolisové algoritmy bežiacie nad nimi majú rovnakú teplotu. K tomu, aby stavi medzi sebou interagovali, zaviedla sa medzi stavmi operácia *križenia* podobným spôsobom ako v genetických algoritmoch. Nech $x, x' \in P$ sú dva náhodné vybrané stavy, operácia križenia vytvorí z týchto dvoch "rodičovských" stavov dva nové stavy - potomkov

$$(z, z') = O_{cross}(x, x') \quad (6.27)$$

Formálne, nech $x=(x_1,x_2,\dots,x_N)$ and $x'=(x'_1,x'_2,\dots,x'_N)$ sú dva rodičovské stavy vyjadrené ako reťazce znakov dĺžky N , potom potomkovia sú definovaný takto

$$z=(x_1,\dots,x_r,x'_{r+1},\dots,x'_N) \quad (6.28a)$$

$$z'=(x'_1,\dots,x'_r,x_{r+1},\dots,x_N) \quad (6.28b)$$

kde r je náhodne vybraný bod kríženia, $1 < r < N$.

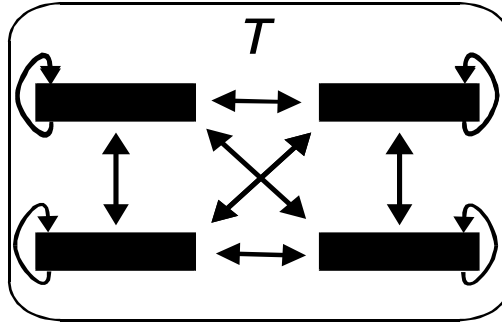


Figure 6.6. Schematické znázornenie paralelného simulovaného žihania. Na jednotlivé stavy (znázornené čiernymi blokmi) sú aplikované nezávislé Metropolisove algoritmy s rovanakou teplotou T . Šípky znázorňujú, že stavy medzi sebou interagujú s malou pravdepodobnosťou P_{cross} .

```

procedure Parallel_Metropolis_algorithm(input:  $P_{\text{ini}}, k_{\text{max}}, T$ ;
                                         output:  $P_{\text{out}}$ );
begin  $k:=0$ ;  $P:=P_{\text{ini}}$ ;
      while  $k > k_{\text{max}}$  do
        begin  $k:=k+1$ ;
          if  $\text{random} < P_{\text{cross}}$  then
            begin select randomly  $x \in P$ ;
                   $x' := O_{\text{pertur}}(x)$ ;
                   $Pr := \min(1, \exp(-(f(x') - f(x))/T))$ ;
                  if  $\text{random} < Pr$  then  $P := (P \setminus \{x\}) \cup \{x'\}$ ;
            end else
              begin select randomly  $x, x' \in P$ ;
                     $(z, z') := O_{\text{cross}}(x, x')$ ;
                     $Pr_1 := \min(1, \exp(-(f(z) - f(x))/T))$ ;
                     $Pr_2 := \min(1, \exp(-(f(z') - f(x')/T))$ ;
                    if  $\text{random} < Pr_1$  then  $P := (P \setminus \{x\}) \cup \{z\}$ ;
                    if  $\text{random} < Pr_2$  then  $P := (P \setminus \{x'\}) \cup \{z'\}$ ;
              end;
            end;
           $P_{\text{out}} := P$ ;
        end;
end;

```

Algoritmus 6.5. Implementácia paralelného Metropolisovho algoritmu. Vstupné parametre sú P_{ini} (počiatočná množina stavov), k_{max} , T , výstupný parameter je P_{out} (konečná množina stavov). Vonkajší while cyklus sa opakuje k_{max} krát. Pravdepodobnosť P_{cross} určuje náhodný výber jednoduchej poruchy vybraného stavu x alebo či vykoná kríženie nad dvojicou náhodne vybraných stavov x a x' . Dvojica nových stavov, ktoré sú produkované krížením, je akceptovaná do množiny stavov pomocou Metropolisovho kritéria aplikovaného nezávisle nad dvoma dvojicami stavov (x, z) a (x', z') .

```

procedure Parallel_Simulated_annealing(input:  $T_{max}, T_{min}, k_{max}, \alpha$ ;
                                         output:  $x_{opt}$ );
begin P:=randomly generated pool of states  $x, x', \dots$ ;
      T:= $T_{max}$ ;
      while  $T > T_{min}$  do
        begin  $P_{ini} := P$ ;
          Parallel_Metropolis_algorithm( $P_{ini}, k_{max}, T, P_{out}$ );
           $P := P_{out}$ ;
           $T := \alpha * T$ ;
        end;
       $x_{opt} := \arg \min \{f(x), x \in P\}$ ;
end;

```

Algoritmus 6.6. Implementácia paralelného simulovaného žihania, vstupné parametre sú $T_{max}, T_{min}, k_{max}, \alpha$, výstupný parameter je x_{opt} . Algoritmus je inicializovaný náhodnou generáciou množiny stavov z P . Po skončení algoritmu, najlepší stav z P je výstup označený x_{opt} .

Proces interakcie dvoch náhodne vybraných stavov $x, x' \in P$ v rámci Metropolisovho algoritmu je určená pravdepodobnosťou $0 < P_{cross} < 1$. Táto pravdepodobnosť určuje proces výmeny, t.j. namiesto mutácie stavu $x \in P$ interakcia dvoch stavov $x, x' \in P$ pomocou operátora kríženia (6.27) je uskutočnená. Potom, problém, či nové stavy z a z' sú akceptované je riešený pomocou Metropolisovho kritéria (2) aplikovaného separátne pre páry (x, z) a (x', z') zvlášť. Tak napríklad, ak Metropolisove kritérium akceptuje nový stav z , potom množina P je obnovená tak, že starý stav x je substituovaný novým stavom z , formálne $P := (P \setminus \{x\}) \cup \{z\}$. Výsledné riešenie je určená ako najlepšie riešenie z množiny P

$$x_{opt} = \arg \min_{x \in P} f(x) \quad (6.29)$$

kde P je množina riešení rezultujúca z dokončeného simulovaného žihania. Procedúry Metropolis_algorithm a Simulated_annealing môžu byť jednoducho modifikované pre potreby paralelného simulovaného žihania. pozri algoritmy 6.5 a 6.6.

Príklad 6.2. Formuláciou simulovaného žihania sme zavřřili celú plejádu evolučných optimalizačných algoritmov: (1) horolezecký algoritmus (rôzne verzie, s učením, metóda zakázaného prehladávanía, evolučné programovanie), (2) genetické algoritmy a (3) simulované žihanie. Zvolte si určité modelové úlohy a na týchto testujte efektivnosť vyššie uvedených metód a vykonajte ich porovnanie.

Literatúra

- [1] P. J. M. van Laarhoven and E. H. L. Aarts, *Simulated Annealing. Theory and Applications*. Reidel, Dordrecht (1987).
- [2] R. H. J. M. Otten and L. P. P. van Ginneken, *Annealing Algorithm*. Kluwer, Boston (1989).
- [3] S. Kirkpatrick, C. D. Gelatt Jr. and M. P. Vecchi, Optimization by Simulated Annealing. *Science* **220**, 671-680 (1983).
- [4] J. Černý, Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm. *J. Opt. Theory Appl.* **45**, 41-51 (1985).
- [5] N. Metropolis, A. W. Rosenbluth, N. N. Rosenbluth, A. H. Teller and E. Teller, Equation of State Calculations for Fast Computing Machines. *J. Chem. Phys.* **21**, 1087-1092 (1953).
- [6] M. Toda, R. Kubo, and N. Saito, *Statistical Physics*. Springer Verlag, Berlin (1983).
- [7] V. Kvasnička, J. Pospíchal, Simulated Annealing. *MATCH (Communications in Mathematical and Computer Chemistry)* **34** (1996), 7-49.
- [8] S. W. Mahfoud, D. E. Goldberg, A genetic algorithm for parallel simulated annealing. in *Parallel Problem Solving from Nature, 2*, ed. by R. Manner, B. Manderick, pp. 301-310, North-Holland, Amsterdam (1992).
- [9] J. Pospíchal, V. Kvasnička, Fast Evaluation of Chemical Distance by Simulated Annealing Algorithm. *J. Chem. Inf. Comp. Sci.* **33** (1993), 879-885.