

Simulované žíhání jako nástroj k hledání optimálního řešení

Michael Pokorný

Střední škola aplikované kybernetiky s.r.o., Hradecká 1151, Hradec Králové
pokorny.michael@ssakhk.cz

Abstrakt

Simulované žíhání je metoda hledání optimálních řešení úloh využívající fyzikální intuici krystalizujícího kovu. Realizoval jsem jej v jazyce C++ a provedl pokusy na vhodných funkcích.

1 Úvod

Simulované žíhání (angl. simulated annealing) je metoda pro vyhledání přibližného optimálního řešení úlohy, která má velký stavový prostor nebo která nelze optimalizovat analyticky. Jako příklad takové úlohy lze zmínit například problém batohu (jak naplnit batoh omezené nosnosti věcmi s co největší hodnotou?). Oproti prohledání všech možných konfigurací systému sice nezaručuje nalezení doopravdy ideálního výsledku, ale zato v konstantním počtu kroků pravděpodobně najde přinejmenším výsledek velmi dobrý. Byla nezávisle popsána Scottem Kirkpatrickem, C. Danielem Gelattem a Mariem P. Vecchim v roce 1983 a Vladem Černým v roce 1985.

2 Tělo příspěvku

2.1 Inspirace metody

Inspiraci k metodě simulovaného žíhání poskytla metalurgie. Při žíhání se postupně ochlazuje zahřátý kov, ve kterém vznikají velké krystaly s menšími defekty. Při zahřátí se atomy uvolní ze svých počátečních pozic (z lokálních minim vnitřní energie) a začínají náhodně kmitat mezi stavy s vyšší energií. S ochlazováním atomy kmitají čím dál tím méně a více se drží v pozicích s nízkými energiemi.

Simulované žíhání obdobně postupně „ochlazuje“ průběžný stav. Vnitřní energii, v jejímž minimu se stav postupně usadí, nahrazuje minimalizované funkce. V každém kroku najde náhodný bod v „okolí“ současného stavu, porovná jeho „energií“ se současným stavem, a s určitou pravděpodobností závislou na tomto rozdílu a na „teplotě“ se do tohoto nového stavu přesune. Tato pravděpodobnost není nutně nulová v případě, že nový stav je „horší“ než stav původní, proto tato metoda při dostatečné teplotě neuváže v lokálním minimu. Tuto metodu lze použít na funkce nejen nad \mathbb{R} nebo \mathbb{Z} , ale i nad vektory.

Vstup: T_0 (počáteční teplota), \vec{x}_0 (počáteční stav)

$T \leftarrow T_0$, $\vec{x} \leftarrow \vec{x}_0$, $k \leftarrow 0$

opakuji

$$\left| \begin{array}{l} \vec{x}_n \leftarrow \text{stav z okolí } \vec{x} \\ p \leftarrow g\left(\frac{f(\vec{x}) - f(\vec{x}_n)}{T}\right) \\ \vec{x} \leftarrow \vec{x}_n \text{ s pravděpodobností } p \\ k \leftarrow k + 1 \\ T \leftarrow h(T_0, k) \end{array} \right.$$

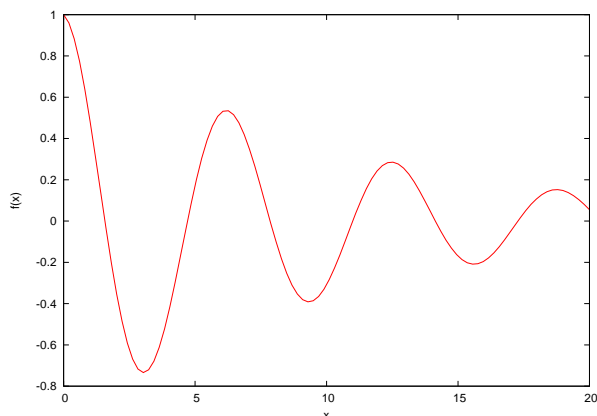
dokud $k < k_{\max}$;

Charakteristika simulovaného žíhání je do velké míry určena charakteristickou skokovou funkcí g a ochlazovací funkcí h . Mezi jejich obvyklé kombinace patří:

- „Klasická“: $g_1(x) = \begin{cases} \exp(x) & x < 0 \\ 1 & x \geq 0 \end{cases}$, $h_1(T_0, k) = T_0 \cdot Q^{k-1}$
- „Moderní“: $g_2(x) = (1 + \exp(-x))^{-1}$, $h_2(T_0, k) = \frac{T_0}{\log_2(K+2)}$
- FSA (fast simulated annealing): $g_3(x) = \frac{1}{2} + \frac{\arctan x}{\pi}$, $h_3(T_0, k) = \frac{T_0}{1 + \frac{k}{N_0}}$

FSA také zavádí oproti jednodušším variantám novinku: nové řešení se nyní určuje jako $x_{\text{nov}} = x + \Delta \cdot \xi$, kde ξ je náhodná veličina s Cauchyho rozdělením a $\Delta_K = \frac{\Delta_0}{1 + \frac{K}{N_0}}$ ($N_0 \in \mathbb{N}$, $\Delta_0 \in \mathbb{R}$; N_0 je stejné jako v chladící funkci). Toto umožňuje FSA se snižováním teploty nejenom „méně poskakovat“, ale i „lépe mířit“.

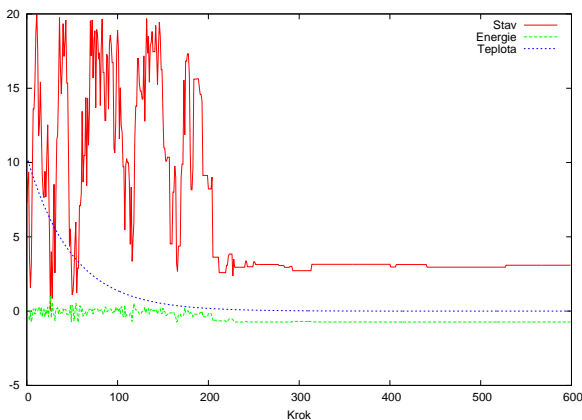
2.2 Pokusy



Obrázek 1: Graf $f(x) = \exp\left(\frac{-x}{10}\right) \cdot \cos(x)$ na $\langle 0; 20 \rangle$

Svůj program jsem nejdříve testoval minimalizací $f(x) = \exp\left(\frac{-x}{10}\right) \cdot \cos(x)$ na intervalu $\langle 0; 20 \rangle$. Tato funkce má na tomto intervalu 3 lokální minima v bodech π , 3π a 5π , přičemž první z nich je minimem celého intervalu.

Otestoval jsem jak „klasickou“ a „moderní“ sadu skokových a chladících funkcí, tak i FSA. FSA oproti jednodušším strategiím dosáhlo minima za mnohem kratší dobu, a navíc získalo výsledek s větší přesností.

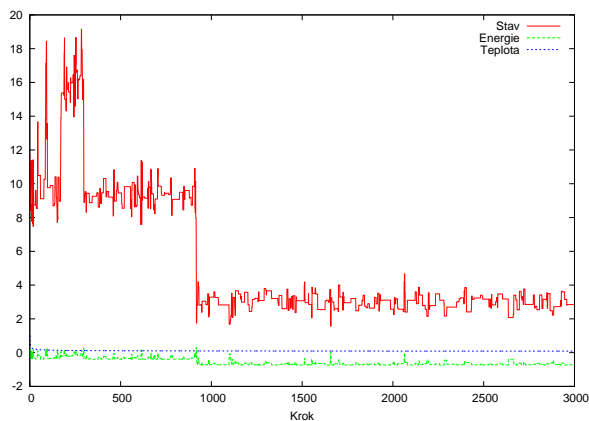


Obrázek 2: Porovnání různých strategií žíhání - klasická

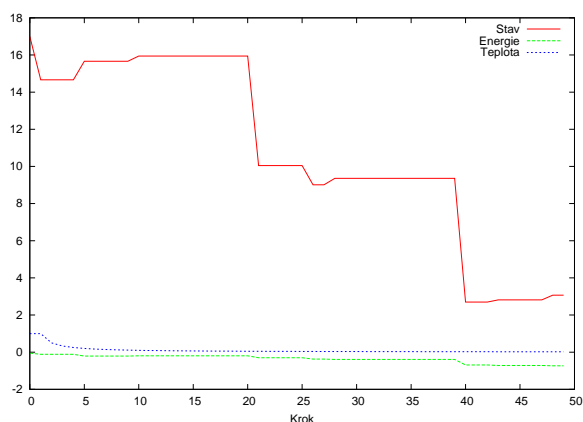
Dále jsem program upravil na hledání výsledku ve formě vektoru a vyzkoušel jej na Rosenbrockově a Himmelblauově funkci.

Rosenbrockova funkce je definována jako $R(x, y) = (1-x)^2 + 100(y-x^2)^2$. Často se využívá na testování optimalizačních algoritmů, protože její graf je „údolí“, ve kterém se na souřadnicích $(1, 1)$ nachází globální minimum 0. Je jednoduché najít „údolí“, ale je poměrně složité najít toto „mělké“ globální minimum.

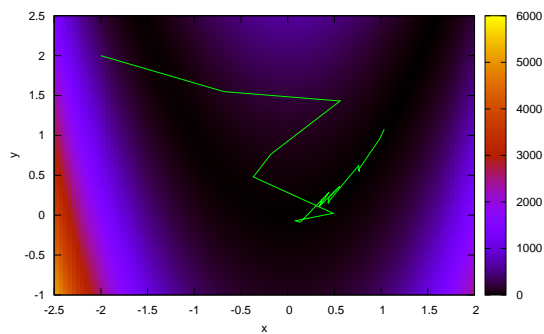
Himmelblauova funkce je opět polynom: $H(x, y) =$



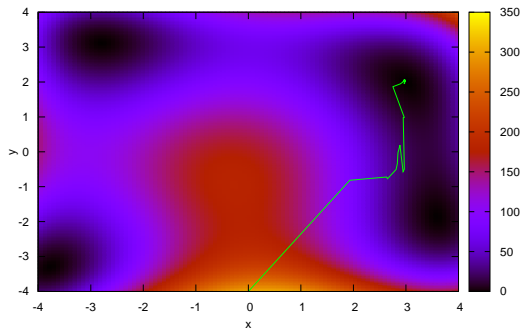
Obrázek 3: Porovnání různých strategií žíhání - moderní



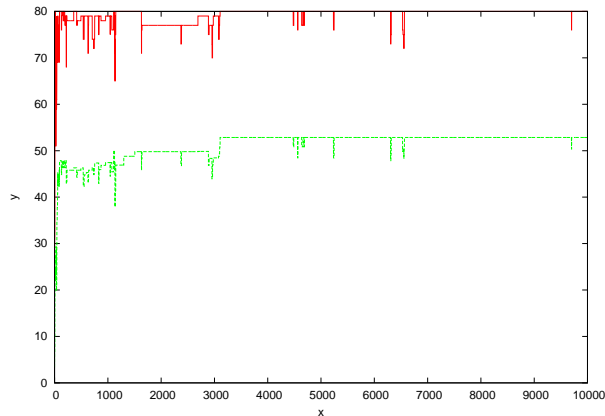
Obrázek 4: Porovnání různých strategií žíhání - FSA



Obrázek 5: Průběh hledání minima Rosenbrockovy funkce



Obrázek 6: Průběh hledání minima Himmelblauovy funkce



Obrázek 7: Průběh hledání řešení na problém batohu

$(x^2 + y - 11)^2 + (x + y^2 - 7)^2$. Má 4 minima s hodnotou 0 na přibližných souřadnicích $(3, 2)$, $(-2.8, 3.1)$, $(-3.8, -3.2)$ a $(3.6, -1.8)$.

Můj poslední pokus byla optimalizace 0-1 problému batohu (0-1 knapsack problem). Jedná se o klasický NP-kompletní problém: máme různé předměty s různou hmotností a různou cenou a batoh s omezenou nosností. Jak lze naskládat do batohu předměty tak, aby jejich celková cena byla maximální? Protože se jedná o NP-kompletní problém, bylo by neúčinné jej v praxi řešit hledáním přesného dokonalého řešení. V případě, že stačí jen dostatečně dobré, ale nikoliv optimální řešení, je na tuto úlohu simulované žihání vhodný nástroj. Stav se nyní skládá z množiny předmětů, které jsou v batohu, a okolí stavu jsou množiny, ve kterých některé předměty přidáme nebo odebereme. Množství těchto změn je závislé v každém kroku na $\xi \cdot \Delta$. Ohodnocovací funkcí stavu je celková hodnota předmětů v batohu. Program opět našel dobré řešení.

3 Shrnutí

Popsal jsem základní využití techniky simulovaného žíhání k hledání optimálních řešení. Otestoval jsem svoji implementaci na několika testovacích funkcích pro optimalizační algoritmy a na problému batohu.

4 Poděkování

Děkuji doc. Ing. Jaromíru Kukulovi, Ph. D. za inspirující odborné vedení miniprojektu a Ing. Vojtěchu Svobodovi, CSc. za organizaci Týdne vědy na Jaderce 2011.