ANALYSIS OF VARIOUS APPROACHES TO SOLVING OPTIMIZATION TASKS

Jakub Knoflíček
Master Degree Programme (2), FIT BUT
E-mail: xknofl00@stud.fit.vutbr.cz

Supervised by: František Vítězslav Zbořil
E-mail: zboril@fit.vutbr.cz

Abstract: This paper deals with various methods for solving optimization tasks. First we show formal definition of optimization task and used test tasks. Then description of used approaches for solving these problems follows. In this chapter examined parameters of mentioned methods are presented too. At the end some implementation details are presented. In conclusion there are first partial results shown.

Keywords: Optimization tasks, ant colony optimization, particle swarm optimization, genetic algorithm, simulated annealing, reinforcement learning.

1. ÚVOD

V rámci tohoto dokumentu bude prezentována práce na projektu s českým názvem Analýza různých přístupů k řešení optimalizačních úloh. Jak je z názvu patrné, pro pochopení je nutné znát formální definici optimalizační úlohy, kterou si tudíž uvedeme ihned v následující kapitole spolu s konkrétními testovacími úlohami. Tyto slouží pro nalezení optimálního nastavení parametrů dále zmíněných metod pro řešení optimalizačních úloh. Hlavním cílem práce není tedy jen implementace metod, ale především optimalizace jejich nastavení vzhledem k zadaným úlohám.

2. OPTIMALIZAČNÍ ÚLOHA

Matematickou optimalizační úlohu můžeme formálně definovat následovně (např. viz [1]):

$$minimalizace\ f_0(x) \tag{1}$$

$$vzhledem\ k\ f_i(x) \leq b_i, i = 1 \dots m, \tag{2}$$

kde funkce $f_0: \mathbb{R}^n \to \mathbb{R}$ je objektivní funkcí, dále $f_i: \mathbb{R}^n \to \mathbb{R}, i = 1 \dots m$, jsou omezující funkce, přičemž meze jsou dány hodnotami $b_1, \dots, b_m$. Vektor $x = (x_1, \dots, x_n)$ reprezentuje optimalizační proměnnou úlohy. Snahou optimalizačních metod je nalezení globálního (případně lokálního) extrému, který budeme značit x^*. Tento je nazýván optimálním, též řešením úlohy, pokud pro libovolné z, kde $f_i(z) \leq b_i, i = 1 \dots m$, platí $f_0(x^*) \leq f_0(z)$. Uveďme nyní použité testovací úlohy.

2.1. PROBLÉM VÍCE BATOHŮ

Problém více batohů je diskrétní úloha, kde je k dispozici m zdrojů s rozpočtem c_i a n objektů. Každý objekt j potřebuje r_{ij} zdroje i a tvoří zisk p_j. Hledáme podmnožinu objektů, přiřazených různým zdrojům, které v součtu mají maximální zisk, bez překročení rozpočetu jakéhokoliv zdroje.

2.2. PROBLÉM POKRYTÍ MNOŽINY

Jedná se o diskrétní úlohu, kde je dáno univerzum s m prvky. Dále existuje n množin, kde každá množina pokrývá podmnožinu univerza a má cenu c. Hledáme podmnožinu n takovou, že množiny, které jsou jejími prvky, při sjednocení pokryjí celé univerzum a součet jejich cen je minimální.

2.3. HLEDÁNÍ MINIMA RASTRIGINOVY A ACKLEYHO FUNKCE

Úkolem je nalezení minima následující n-dimenzionální Rastriginovy (3) a Ackleyho (4) funkce:

$$f(x) = 10n + \sum_{i=1}^{n}\left[x_i^2 - 10\cos(2\pi x_i)\right] \tag{3}$$

$$f(x) = -a \cdot \exp\left(-b \cdot \sqrt{\frac{1}{n}\sum_{i=1}^{n} x^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(cx_i)\right) + a + \exp(1) \tag{4}$$

3. OPTIMALIZAČNÍ METODY

Pod pojmem optimalizační metody budeme chápat různé přístupy k řešení optimalizačních úloh. V práci bylo rozebráno celkem pět následujících metod.

3.1. OPTIMALIZACE MRAVENČÍ KOLONIÍ

Optimalizace mravenčí kolonií (viz [2]) je heuristickou multiagentní metodou. Činnost je inspirována skutečnými mravenci. Agenti náhodně konstruují řešení a značí svou cestu stavovým prostorem pomocí přidáním různě velké hodnoty feromonu τ, dle toho, k jak moc kvalitnímu výsledku jeho cesta vedla. Lokální aktualizaci feromonové hladiny prování všichni agenti při každém kroku konstrukce svého řešení. Jakmile dosáhnou všichni agenti koncového stavu, agent s nejlepším řešením provede aktualizaci feromonových hladin dle svého řešení. Uveďme, že tyto feromony se s časem samovolně vypařují (snižují svou hodnotu) dle nastavitelného parametru. Dalšími zkoumanými parametry jsou konstanty α, β a q_0 ve vztazích pro výběr následujícího stavu s agenta k (viz (5) a (6)):

$$s = \arg\max_{(ij)\notin tabu_k} \tau_{ij}^{\alpha} + \eta_{ij}^{\beta}\, jestliže\ q \leq q_0, jinak\ použij\ pravidlo\ AS, \tag{5}$$

kde následující AS pravidlo, vyjadřuje pravděpodobnost výběru cesty ij agentem k:

$$q_{ij}^{k} = \frac{\tau_{ij}^{\alpha} + \eta_{ij}^{\beta}}{\sum_{(ix)\notin tabu_k}\left(\tau_{ix}^{\alpha} + \eta_{ix}^{\beta}\right)}\ jestliže\ (ij) \notin tabu_k, jinak\ 0 \tag{6}$$

Významný je také počet agentů, počet iterací algoritmu a obzvláště volba vhodné heuristiky η.

3.2. OPTIMALIZACE HEJNEM ČÁSTIC

Optimalizace hejnem částic je stochastická metoda, založená na množině částic, které jsou umístěny ve stavovém prostoru. U každé částice pozorujeme pozici $v_{id}(0)$ (8) a vektor rychlosti (7). Pozice ve stavovém prostoru vyjadřuje řešení úlohy. V práci hledáme optimální počet částic, velikost počáteční rychlosti $v_{id}(0)$ každé částice i v dimenzi d a též nastavení konstant c_1 a c_2 v následující rovnici, kde p_{id} je nejlepší dosažená hodnota částicí, p_{gd} nejlepší hodnota v rámci celé topologie a t iterace. Také pozorujeme vliv různých topologií částic a modifikací rovnice. Více např. v [3].

$$v_{id}(t) = v_{id}(t-1) + c_1 \cdot rand() \cdot (p_{id} - x_{id}(t-1)) + c_2 \cdot rand() \cdot \left(p_{gd} - x_{id}(t-1)\right) \tag{7}$$

$$x_{id}(t) = (t) = x_{id}(t-1) + v_{id}(t) \cdot \Delta t \tag{8}$$

3.3. GENETICKÉ ALGORITMY

Genetický algoritmus (viz [4]) je metoda pro hledání aproximačních řešení optimalizačních problémů, založená na genetickém modelu. Ve zkoumané variantě máme populaci jedinců, přičemž každý jedinec je reprezentován chromozomem, jenž kóduje možné řešení. Dva jedinci se kříží a tvoří tak své potomky vzájemnou výměnou částí svých chromozomů nebo uniformním stochastickým výběrem každého genu chromozomu od jednoho z rodičů. V práci proto hledáme optimální nastavení výběru rodičů ke křížení (ruleta, stochastické náhodné vzorkování, turnaj či výběr elity), dále počet míst křížení chromozomu a způsob zařazení nových potomků do stávající populace (model s překrytím generací, generační a inkrementační model). Diverzifikace je zajištěna mutací (mutace náhodného genu na jiný), čili též hledáme vhodnou pravděpodobnost mutace chromozomu. Konečně je důležité nalézt vhodný počet iterací algoritmu a velikost počáteční populace.

3.4. SIMULOVANÉ ŽÍHÁNÍ

Simulované žíhání (více v [5]) je optimalizační metoda inspirovaná žíháním kovů. Aktuální stav C vyjadřuje současné dosažené řešení. Dále existují následující stavy N, přičemž rozdíl energií (ohodnocení) mezi N a C značíme ΔE. Nový stav N přijmeme za současný, pokud má menší energii než C, nebo jej přijmeme s pravděpodobností $p(N) = e^{-\frac{\Delta E}{T}}$. Parametr T, zvaný teplota, klesá s každou iterací dle vztahu $T_{i+1} = \alpha T_i$, dokud nedosáhne nulové teploty. Hledáme optimální počáteční teplotu T_0 a parametr klesání teploty α.

3.5. POSILOVANÉ UČENÍ

Posilované učení (viz [6]) je metoda založená na ohodnocování stavů (TD learning, (9)) či akcí v určitých stavech (Q learning, (10)) při konstrukci řešení. Pro ohodnocení jsou generovány náhodné konstrukční cesty, kdy na každý krok je aplikováno aktualizační pravidlo. Nejprve budeme hledat dostatečný počet těchto cest. Snahou pak bude nalézt optimální nastavení parametru učení α a discount faktoru γ v těchto rovnicích, kde s a s' jsou minulý a současný stav, $U(s)$ resp. $Q(s,a)$ je ohodnocení stavu s resp. akce a ve stavu s, $R(s)$ odměna za dosažení stavu s:

$$U(s) = U(s) + \alpha\big(R(s) + \gamma U(s') - U(s)\big) \tag{9}$$

$$Q(s,a) = Q(s,a) + \alpha\big(R(s) + \gamma \max_{a'} Q(s',a') - Q(s,a)\big) \tag{10}$$

4. IMPLEMENTACE

Implementace je provedena v jazyce Java verze 7, přičemž pro vyhnutí se redundance kódu je obecná část každé metody implementována za pomoci abstraktních metod a generických datových typů. Pro konkrétní řešení každé z úloh je pak pouze implementována pro ni specifická část. Z tohoto pohledu lze na projekt také nahlížet na jakousi malou knihovnu s metodami, které lze s malým doplněním užít pro libovolnou úlohu.

5. ZÁVĚR

Na závěr uveďme, že proces hledání optimálního nastavení parametrů metod není zatím dokončen, ale první měření naznačují signifikantní rozdíly při různém nastavení optimalizačních metod, čili dokazují smysl této práce. Navíc se ukazuje, že i posilované učení má určitý potenciál i v oblastech optimalizace, kde doposud není příliš využíváno.

PODĚKOVÁNÍ

Tento příspěvek vznikl za podpory grantu FIT-S-11-1 a výzkumného záměru MSM 0021630528.

REFERENCE

[1] Boyd, S., Vandenberghe, L.: Convex Optimization, Cambridge University Press

[2] Doringo, M., Stützle, T.: Ant Colony Optimization, Bradford Books, 2004

[3] Settles, M.: An Introduction to Particle Swarm Optimization, 2005

[4] Sivanandam, S. N., Deepa, S. N.: Introduction to Genetic Algorithms, Springer, 2008,

[5] Lam, J.: An Efficient Simulated Annealing Schedule, Yale University, 1988

[6] Russell, S.,Norvig, P.: Arfiticial Intelligence: A Modern Approach, Prentice Hall, 2010, ISBN 0136042597