HOROLEZECKÝ ALGORITMUS A ÚLOHA CELOČÍSELNÉHO PROGRAMOVÁNÍ V PROSTŘEDÍ MATLAB

Radomil MATOUŠEK

Ústav automatizace a informatiky, FSI VUT v Brně

Ústav přístrojové techniky, AV ČR

email: matousek@uai.fme.vutbr.cz

Abstrakt: Tento příspěvek diskutuje užití specifického heuristického algoritmu pro řešení zvolené celočíselné optimalizační úlohy. Daným heuristickým algoritmem je původní binární varianta všeobecně známého tzv. horolezeckého algoritmu. Řešeným problémem a zároveň komparativní úlohou je návrh parametrů složeného ozubeného převodu.

1. Úvod

V řadě praktických situací je třeba hledat řešení daného problému částečně či zcela v celočíselném oboru. Matematicky je takováto úloha označena jako úloha celočíselného programování. Ve strojírenské praxi touto úlohou může být například problematika návrhu složeného ozubeného převodu.

Diskutovaná úloha je prezentována jako komparativní studie řešení návrhu počtu zubů ve složeném převodu. Úloha byla předložena v roce 1990 [Sandgren90] a od té doby využita mnoha autory k prověření různých optimalizačních postupů. V případě tohoto příspěvku je prezentovaná optimalizační metoda založená na původní binární variantě tzv. horolezeckého algoritmu (HC). HC algoritmus je implementován jako samostatná optimalizační metoda (s označením HCA5) v prostředí Matlab.

Mnohé úlohy celočíselného programování lze řešit optimalizačními postupy, které dávají řešení v reálném číselném oboru, přičemž následným převodem (zaokrouhlením, odseknutím) obdržíme řešení celočíselné. Takovéto postupy mohou být v některých případech uspokojivé, v jiných však mohou vést k velkým chybám či nepřípustným řešením. Prezentovaná metoda HCA5 je schopna, na rozdíl od mnoha metod obsažených v *Optimization Tolboxu*, pracovat přímo v celočíselném oboru. Equation Chapter 1 Section 1

2. Obecná formulace úlohy celočíselného programování

Úloha celočíselného programování může být obecně formulována následovně:

$$\min f(x_1, x_2, \ldots, x_n) \tag{1.1}$$

za podmínek

$$\begin{gathered} g_i(x_1, x_2, \ldots, x_n) \le 0, \quad i = 1, 2, \ldots, m, \\ x_j \in M_j \subseteq \mathbb{Z}, \quad j \in J \end{gathered} \tag{1.2}$$

kde $J \neq 0$, $J \subseteq \{1, 2, \ldots, n\}$ a $\mathbb{Z}$ je množina celých čísel. Úlohy celočiselného programování [KlDvPo01] pak dělíme podle charakteru funkcí f, $g_1, g_2, \ldots, g_m$ na *lineární* a *nelineární*.

Pokud jsou podmínkou celočíselnosti vázány všechny proměnné jedná se o *ryze celočíselnou úlohu*. Jestliže se podmínka celočíselnosti týká pouze některých proměnných (tj. $J \subset \{1, 2, \ldots, n\}$), hovoříme o *částečně celočíselné úloze*. V rámci úloh celočíselného programování tvoří zvláštní skupinu úlohy tzv. 0/1 programování, pro které $M_j = \{0,1\}$, $j = 1, 2, \ldots, n$. Pro všechny tyto třídy úloh je možné metodu HCA5 využít.

3. Horolezecký algoritmus – metoda HCA5

V kontextu optimalizačních metod můžeme HC algoritmus zařadit k metodám, které směr nejvhodnějšího postupu určují na základě prohledání svého okolí. Z tohoto faktu vyplývá, že k výpočtu není potřeba gradient, ale pouze apriorní znalost hodnot účelové funkce.

Obecný popis algoritmu HCA5 [Matousek04] je následující:

- Pro aktuální řešení (tj. i první aproximace) se v metrickém Hammingově prostoru $\mathbb{H}$ generuje pomocí konečného souboru transformací specifické okolí.
- Poté se objektivní funkce extremalizuje na tomto okolí, přičemž může být zahrnuto i původní (zdrojové) řešení.
- Získané řešení se použije jako základ pro generování nového okolí.
- Ukončovací kritérium může být voleno například na základě max. počtu iterací nebo na základě neschopnosti algoritmu generovat lepší řešení.

Stručná syntaxe funkce, realizující algoritmus HCA5, je uvedena následující tab. 1. Podrobný popis je k dispozici v režimu *help*. K řešení prezentované úlohy byla použita implementace HCA5 s metodou transformace HC12, která se ukázala jako zcela vyhovující.

Metoda:	**HCA5**	
Popis:	Realizuje HC algoritmus s využitím zvolené/zvolených binárních transformací.	
Syntaxe:	`[vecA] = HCA5(nParam,nBitParam,iParam,iType,mCode,… mHC,mView,funOpt,funName,varargin)`	
parametr	Hodnoty	
`mHC`	`'HC1'`	Generuje okolí se vzdáleností $\rho_H = 1$ od původního řešení.
	`'HC2'`	Generuje okolí se vzdáleností $\rho_H = 2$ od původního řešení.
	`'HC12'`	Generuje okolí se vzdálenostmi $\rho_H = 1$ a $\rho_H = 2$ od původního řešení.
	`'HC1n'`	Generuje okolí se vzdálenostmi $\rho_H = 1$ a $\rho_H = (n-1)$ od původního řešení.

Tab. 1: Stručná syntaxe metody HCA5.

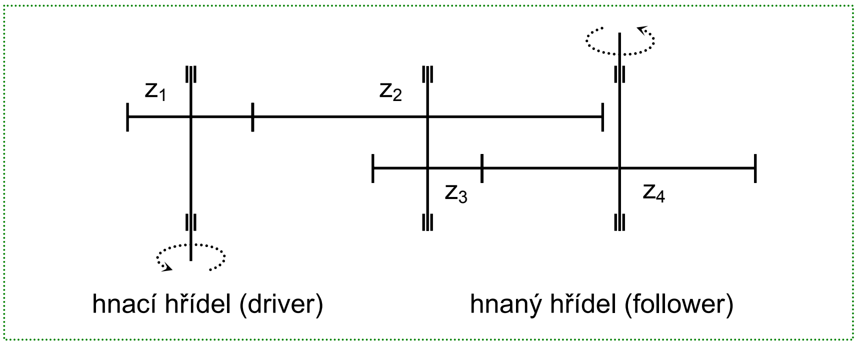
4. Návrh parametrů ozubeného soukolí

Při návrhu strojních součástí sloužících k převádění točivého pohybu (tedy převodů), může být optimalizační otázka postavena tak, abychom pro koncepčně navržený složený převod (soukolí) s předem daným požadavkem převodového poměru $i_{\text{požadavek}}$, navrhli například konfiguraci počtu zubů jednotlivých ozubených kol.

Převodový poměr i vyplývá z normálových složek vektorů rychlostí v bodu dotyku dvou zubů a je dán poměrem úhlových rychlostí ω (tedy též počtem otáček) nebo obráceným poměrem poloměru základních nebo valivých kružnic R_v. Prakticky je převodový poměr dán poměrem počtu zubů z.

$$i = \frac{\omega_1}{\omega_2} = \frac{R_{v2}}{R_{v1}} = \frac{z_2}{z_1} \tag{1.3}$$

Případ řešeného soukolí a výpočet převodového poměru je následující:



Obr.1: Schéma řešeného složeného převodu.

$$i = \frac{\omega_D}{\omega_F} = \frac{z_1 z_3}{z_2 z_4} \tag{1.4}$$

Optimalizační úloha je pro daný $i_{\text{požadavek}}$ = 1/6,931 a možné počty zubů z následující:

$$F_z = (i_{požadavek} - i)^2 = \left(\frac{1}{6.631} - \frac{z_1 z_3}{z_2 z_4} \right)^2 \tag{1.5}$$

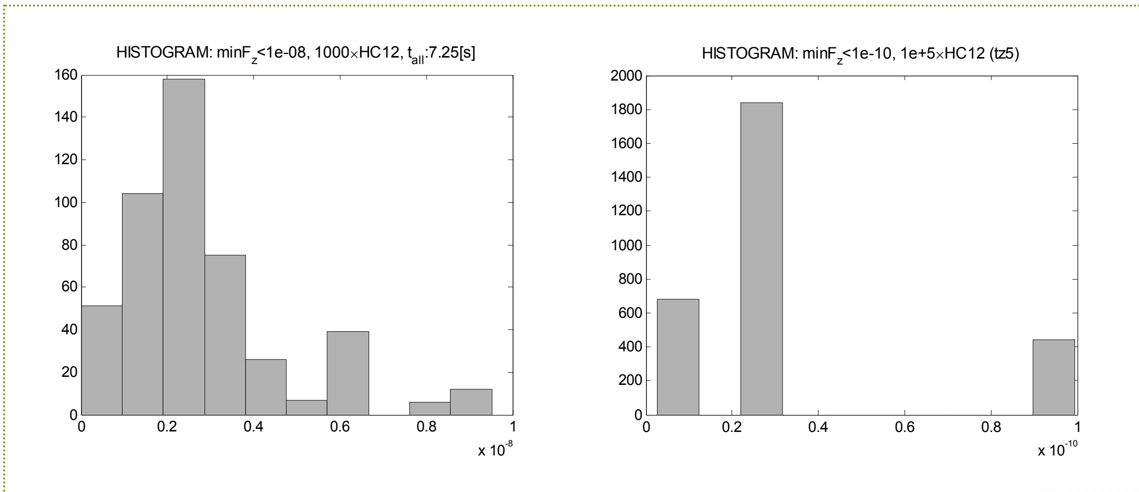
$$\min \left\{ F_z \mid z \in \{12, 13, \ldots, 60\} \right\}$$

Z hlediska terminologie optimalizace se jedná o celočíselnou úlohu s omezením. Úloha byla navržena [Sandgren90] a řešena mnoha autory i rozdílnými numerickými postupy. Výsledky řešení, včetně odkazů na autory výpočtů jsou předmětem následující tabulky, částečně převzaté z uvedené literatury.

$i_{\text{požadavek}}$ = 1/6.931 = 0.14427932477276		Nalezená řešení – navržené počty zubů									
		Sandgren [Sandgren90]	Fu et al. [FuFeGl91]	Loh [LoHaPa91]	Zhang [ZhChVa93]	Lin [LiZhCh95]	Wu a Chow [WuCho95]	Cao a Wu [CaoWu97]	Lampinen [LamZel99]	Zelinka [Zelinka02]	Matoušek
Ozubení	z_1	18	14	19	30	19	19	30	16	19	19 (16)
	z_3	22	29	16	15	16	16	15	19	16	16 (19)
	z_2	45	47	42	52	49	43	52	43	43	49 (43)
	z_4	60	59	50	60	43	49	60	49	49	43 (49)
minF_z		5.7e-06	4.5e-06	2.3e-07	2.4e-09	2.7e-12	2.7e-12	2.4e-09	2.7e-12	2.7e-12	2.7e-12
i		0.146666	0.146411	0.144762	0.144231	0.144281	0.144281	0.144231	0.144281	0.144281	0.144281
Užitá metoda		Metoda větví a mezí, kvadratické programování	Celočíselné nelineární programování	Sekvenční lineární algoritmus	Simulované žíhání	Modifikovaný GA	Meta GA	Evoluční programování	Diferenciální evoluce	Algoritmus SOMA	HC algoritmus HC12

Tab. 2: Přehled výsledků řešení pro testovací úlohu návrhu parametrů složeného převodu.

Užitý optimalizační algoritmus byl v rámci jednoho výpočtu 1000x spuštěn, s celkovou časovou režií 7,25 sekund. Těchto testovacích běhů bylo kontrolně provedeno 500, přičemž vždy bylo dosaženo předpokládaného optima. Jednomu výpočtu průměrně odpovídalo 6,2 optim. Distribuce výpočtu je prezentována pro dané přesnosti pomocí histogramů (Obr.2), osa *x* reprezentuje hodnotu účelové funkce.



Obr.2: Histogramy distribuce provedených optimalizačních výpočtů, dle hodnot účelové funkce (zobrazeno pro hodnoty $F_z < 1e{-}08$ a $F_z < 1e{-}10$). Vlevo pro 1× výpočet (1000× restart), respektive vpravo pro 100× výpočtů (100000× restart).

Základní nastavení metody HCA5 umožňuje pracovat přímo s celočíselnou reprezentací. V rámci dekódování tak bylo realizováno zobrazení binárních řetězců na celočíselný interval hledaných parametrů o rozsahu [1, 64]. Protože specifikum optimalizačního problému dle (1.5) vyžadovalo rozsah [12, 60], bylo, vzhledem k síle omezení, užito penalizační funkce přidávající hodnotu jedna za každé nedodržení tolerance intervalu daným parametrem.

5. Závěr

Užitý algoritmus HCA5-HC12 našel, dle následující tabulky, všechny alternativy předpokládaného minimálního řešení, přičemž samozřejmě odhalil i další alternativní řešení s horší hodnotou účelové funkce.

Předpokládaná optimální řešení a další alternativy

Fz	**z1**	**z2**	**z3**	**z4**	alt_min_tZ3
2.7009e-012	16	19	43	49	**A: 16.0%**
2.7009e-012	16	19	49	43	**B: 21.8%**
2.7009e-012	19	16	43	49	**C: 22.0%**
2.7009e-012	19	16	49	43	**D: 40.2%**
2.3078e-011	13(20)	20(13)	34(53)	53(34)	další přípustná, neoptimální, řešení
2.3078e-011	13(30)	30(13)	51(53)	53(51)	
2.3078e-011	15(26)	26(15)	51(53)	53(51)	
9.9399e-011	13(31)	31(13)	49(57)	57(49)	

HC algoritmus HC12

Tab. 3: Výběr nalezených řešení algoritmem HCA-HC12 pro testovací úlohu Fz (složený převod).

Prezentované alternativy nalezeného nejlepšího řešení je samozřejmě možné dovodit z matematické zákonitosti vztahu (1.4). Otázka stability výpočtu z hlediska dosažení minima je pro danou úlohu v podstatě zodpovězena počtem kontrolních běhů. Poslední otázka při provedených testech byla, zda je nalezení jednotlivých alternativ předpokládaných optimálních řešení daným algoritmem stejně pravděpodobné. Odpověď je záporná (viz. Tab. 3), přičemž lze konstatovat, že alternativa A (16%) je nejméně pravděpodobná a alternativa D (40%) nejvíce pravděpodobná.

Uvedená heuristická metoda HCA5 (včetně helpu), je pro akademické účely k dispozici na odkazu www.uai.fme.vutbr.cz/~matousek/gate.html.

Kontrolní běh (100× výpočet)	a	b	c	d	e	ϕ na jeden výpočet
Počet dosažených optim	553	666	605	594	679	6,194
Test homogenity dvou binomických rozdělení	n_a	n_e	p_a	p_e	P	H_0: $p_a = p_e$, $\alpha = 0.01$
Případ a × Případ e	1e+5	1e+5	5,53e-3	6,79e-3	0.9717	Nezamítnuto

Test homogenity dvou binomických rozdělení				H_0: $p_i = p_j$, $\alpha = 0.01$				Z … zamítnuto N … nezamítnuto
Počet Instancí alternativ optimálního řešení		suma: n=605		A	B	C	D	
	A = 97	P	A	16,0%	N	Z	Z	
	B = 132		B	0,0101	21,8%	N	Z	
	C = 133		C	0,0079	0,9330	22,0%	Z	
	D = 243		D	0,0000	0,000	0,000	40,2%	

Tab. 4: Statistické testy pro úlohu Fz.

Poděkování

Práce byla podpořena projekty MŠMT No.: KONTAKT ME 526, MSM 2611 00009 a MSM 2611 00013.

Literatura

[Karpíšek03] Karpíšek, Z.: *Matematika IV, statistika a pravděpodobnost*. Akademické nakladatelství CERM Brno (2. doplněné vydání), Brno, 2003, ISBN 80-214-2522-93.

[KlDvPo01] Klapka J., Dvořák, J., Popela, P., Metody operačního výzkumu, skriptum, nakladatelství VUTIUM, Brno, 2001, ISBN 80-214-1839-7.

[LamZel99] Lampinen, J., Zelinka, I.: *New Ideas in Optimisation – Mechanical Enginering Design Optimisation by Differential Evolution.* Volume 1, McGraw-Hill, London, 1999, ISBN 007-709506-5.

[Maroš99] Maroš, B.: *Empirické modely*, skriptum VUT-FSI Brno, Brno, 1999.

[Matoušek04] Matoušek, R.: *Vybranné metody umělé inteligence – implementace a aplikace.* Disertační práce v oboru technická kybernetika při VUT Brno, (VUT FSI Brno, UPT AV CR), 2004.

[Sandgren90] Sandgren, E.: *Nonlinear integer and discrete programmingin mechanical design optimisation*. Transactions of the ASME, Journal of Mechanical Design, 112(2), 1990, pp.223-229, ISSN 0738-0666

[Zelinka02] Zelinka I.: *Umělá Inteligence – v problémech globální optimalizace*. Nakladatelství BEN, Praha, 2002, ISBN 80-7300-069-5.