

Simpleksinės paieškos tyrimas tikslinimo etape

A. Dambrauskas, D. Udris

Automatikos katedra, Vilniaus Gedimino technikos universitetas,
Naugarduko g. 41, 03227 Vilnius, Lietuva, tel.+37052745063; el.p. dainius.udris@el.vtu.lt

Įvadas

Paieškinės optimizacijos tikslinimo etape pagrindinės charakteristikos yra ekstremumo nustatymo tikslumas ir patikimumas. Kai paieška atliekama trukdžių aplinkoje, sprendinio tikslumą ir patikimumą galima įvertinti naudojant ribinį atstumo iki tikslo S pasiskirstymo tankį [1]

$$f_s(q) = \int_0^{\infty} f_s(q|\xi) f_s(\xi) d\xi, \quad (1)$$

apskaičiuojamą pagal simplekso centro perėjimo iš būsenos $S = \xi$ į būseną q vietinio pasiskirstymo tankio $f_s(q|\xi)$ vidurkį; čia $S = q/\lambda$ – santykinis orbitos spindulys.

Vidutinė optimizacijos paklaida apskaičiuojama iš išraiškos

$$M[S] = \int_0^{\infty} q f_s(q) dq. \quad (2)$$

Optimizavimo uždavinio sprendimo patikimumą galima įvertinti tikimybe, kad atsitiktinio klaidžiojimo procese simplekso centras neišeis iš hipersferos, kurios spindulys S_+ , ribų [1]:

$$P = p\{S < S_+\} = \int_0^{S_+} f_s(q) dq. \quad (3)$$

Šio darbo tikslas – išanalizuoti būsenas atpažįstančią simpleksinę paiešką tikslinimo etape, surasti jos teorines statistines charakteristikas, kai procesą veikia atsitiktiniai trukdžiai bei tikslo dreifas.

Simpleksinės paieškos tyrimas tikslinimo etape

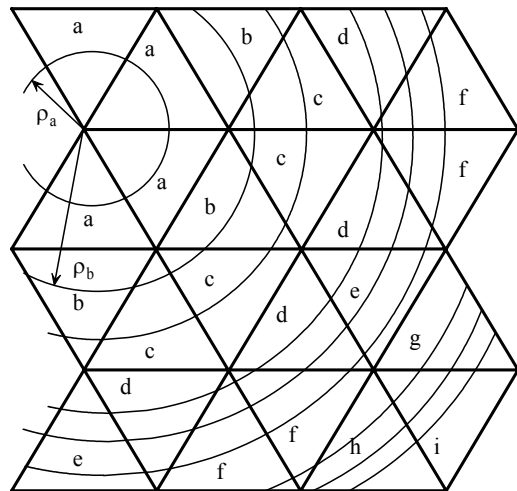
Tarkim, kad optimizuojamo objekto savybės aprašomos lygtimi

$$y = Q(x) + \varepsilon = -\left(a \sum_{i=1}^k x_i^2\right)^{\frac{1}{2}} + \varepsilon. \quad (4)$$

Kiekvienas paieškos netoli ekstremumo žingsnis priklauso nuo simplekso orientacijos ir atstumo iki tikslo. Simplekso padėtys žymimos pagal simplekso centro nuotolį nuo tikslo: $a, b, c, d, e, f, g, \dots$, kurių atstumai iki tikslo – $\rho_a, \rho_b, \rho_c, \rho_d, \rho_e, \rho_f, \rho_g, \dots$. Be to, kiekviena jų turi tam tikrą orientaciją grad Q atžvilgiu (1 pav.).

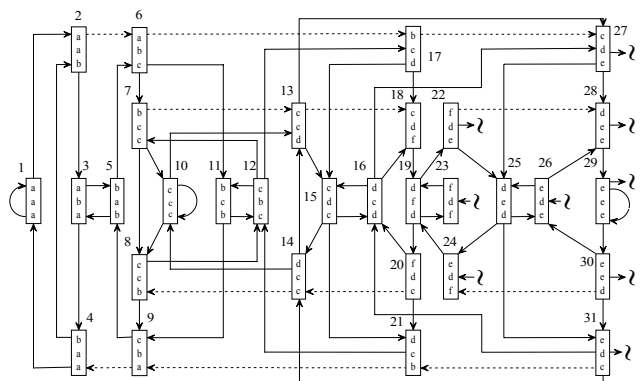
Optimizavimo kriterijumi tikslinimo etape pasirenkamas atsitiktinio simplekso klaidžiojimo aplink ekstremumą vidutinio atstumo iki tikslo minimumas:

$$M[S] \rightarrow \min. \quad (5)$$



1 pav. Simplekso padėtys prie ekstremumo

Bazinę paieškos strategiją lemia laisvai viršūnes atspindintis paieškos algoritmas. Kaip rodo [1] analizė, jis yra optimalus tikslinimo etape. Laisvai viršūnes atspindinčios paieškos procesą tikslinimo etape aprašančios Markovo grandinės stochastinis grafas pavaizduotas 2 pav.



2 pav. Simpleksinės būsenas atpažįstančios paieškos tikslinimo etape Markovo grandinės stochastinis grafas

Paprastai pereinamosios tikimybės gali būti optimizuotos uždraudžiant tiesiaiegi simplekso judėjimą tolyn nuo tikslo. Judėjimo kryptį paieškos metu galima įvertinti netiesioginiu būdu, paieškos algoritme taikant krypties rodiklį d [1]:

$$d = \begin{cases} 1, & \text{kai } \sum_{i=n-v+1}^n K_i \text{ sign } \Delta_i \geq 0, \\ 2, & \text{kai } \sum_{i=n-v+1}^n K_i \text{ sign } \Delta_i < 0; \end{cases} \quad (6)$$

čia K_i – svorio koeficientas; Δ_i – paieškos žingsnio sėkmės rodiklis (ieškant maksimumo $\Delta = y_s^H - y_s$).

Paieškos krypties rodiklis įgalina panaikinti tiesiaiegi simplekso judėjimą tolyn nuo tikslo, t. y. panaikinti 2 pav. punktyru pažymėtus perėjimus tarp būsenų:

$$2 \rightarrow 6, 6 \rightarrow 17, 17 \rightarrow 27, 7 \rightarrow 13, 13 \rightarrow 18, 22 \rightarrow 28, \\ 30 \rightarrow 24, 20 \rightarrow 14, 14 \rightarrow 8, 31 \rightarrow 21, 21 \rightarrow 9, 9 \rightarrow 4,$$

$$(t. y. p_{2_6} = 0, p_{6_{17}} = 0, \dots, p_{9_4} = 0).$$

Taikant šias taisykles bei paieškos krypties rodiklį (6), sudarytas adaptyvus simpleksinės būsenas atpažįstančios paieškos algoritmas tikslinimo etapui. Ieškant maksimumo, paieška atliekama pagal šias taisykles:

1. Iš simplekso viršūnių išrinkti viršūnę v_r , kurios y vertė mažiausia.
2. Priskirti $m = r$.
3. Įvertinti paieškos krypties rodiklį (6):
 - a) kai $d = 1$, vykdyti 4 punktą;
 - b) kai $d = 2$, vykdyti 7 punktą.
4. Atspindėti viršūnę v_m .
5. Surasti y_m^H vertę gautoje naujoje viršūnėje.
6. Vykdyti 1 punktą.
7. Iš visų viršūnių, išskyrus v_r , išrinkti viršūnę v_p , kurios y vertė mažiausia.
8. Priskirti $m = p$ ir vykdyti 4 punktą.

Pereinamųjų Markovo grandinės tikimybių išraiškos gaunamos analizuojant simplekso orientacijas esant atskiroms paprastosioms būsenoms:

$$p_{1_{-1}} = P\left\{(\varepsilon_3 - A < \varepsilon_2) \cap (\varepsilon_3 < \varepsilon_4) \mid (\varepsilon_1 < \varepsilon_3) \cap (\varepsilon_0 < \varepsilon_1)\right\}, \\ p_{1_{-2}} = 1 - p_{1_{-1}}; \quad (7)$$

$$p_{8_{-9}} = P\left\{ \begin{array}{l} \left| (\varepsilon_3 + (1 + \sqrt{3})A < \varepsilon_2) \cap (\varepsilon_3 + (1 + \sqrt{3})A < \varepsilon_4) \right| \\ \left| (\varepsilon_1 + (\sqrt{3} - 2)A < \varepsilon_3) \cap (\varepsilon_0 + (2 - \sqrt{3})A < \varepsilon_1) \right| \end{array} \right\};$$

$$p_{8_{-12}} = 1 - p_{8_{-9}};$$

$$p_{25_{-31}} = P\left\{ \begin{array}{l} \left| (\varepsilon_3 + (2 - \sqrt{7})A < \varepsilon_2) \cap (\varepsilon_3 + (\sqrt{3} - \sqrt{7})A < \varepsilon_1) \right| \\ \left| (\varepsilon_4 + (\sqrt{7} - \sqrt{3})A < \varepsilon_3) \cap (\varepsilon_1 + (2 - \sqrt{3})A < \varepsilon_2) \right| \end{array} \right\};$$

$$p_{25_{-26}} = P\left\{ \begin{array}{l} \left| (\varepsilon_1 + (\sqrt{7} - \sqrt{3})A < \varepsilon_3) \cap (\varepsilon_1 + (2 - \sqrt{7})A < \varepsilon_2) \right| \\ \left| (\varepsilon_4 + (\sqrt{7} - \sqrt{3})A < \varepsilon_3) \cap (\varepsilon_1 + (2 - \sqrt{3})A < \varepsilon_2) \right| \end{array} \right\};$$

$$p_{25_{-24}} = 1 - p_{25_{-31}} - p_{25_{-26}}.$$

Šių pereinamųjų tikimybių (7) įverčiai skaičiuojami taikant Monte Karlo metodą. Šių tikimybių vertėms apskaičiuoti buvo sudaryta kompiuterinė programa, atsiktinių dydžių, pasiskirsčiusių pagal normalųjį dėsnį su nuline matematine viltimi ir dispersija $\sigma_\varepsilon^2 = 1$, vertes imanti iš 2500 elementų lentelės [4]. Skaičiuojant kiekvienos tikimybės įvertį, atlikta $N = 5000$ bandymų.

Nusistovėjusio stacionaraus simplekso klaidžiojimo tikslo srityje proceso sudėtingųjų būsenų ribinės tikimybės randamos sprendžiant lygčių sistemą:

$$\begin{cases} p_b = \sum_{a=1}^{\psi'} p_a p_{ab}, & b = 1, \dots, \psi' - 1, \\ \sum_{a=1}^{\psi'} p_a = 1. \end{cases} \quad (8)$$

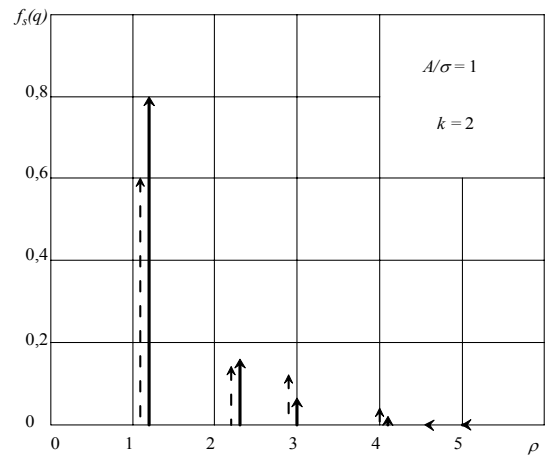
Ribinės paprastųjų būsenų a, b, c, d, e, f tikimybės surandamos sumuojant sudėtingųjų būsenų su vienodomis paskutinėmis paprastosiomis būsenomis tikimybės:

$$\begin{aligned} p_a &= p_1 + p_3 + p_4 + p_9; \\ p_b &= p_2 + p_5 + p_8 + p_{11} + p_{21}; \\ p_c &= p_6 + p_7 + p_{10} + p_{12} + p_{14} + p_{15} + p_{20} + p_{31}; \\ p_d &= p_{13} + p_{16} + p_{17} + p_{19} + p_{25} + p_{30}; \\ p_e &= p_{22} + p_{26} + p_{27} + p_{28} + p_{29}; \\ p_f &= p_{18} + p_{23} + p_{24}. \end{aligned} \quad (9)$$

Tikimybės (9) įgalina apskaičiuoti atsiktinio simplekso klaidžiojimo tikslo rajone ribinį pasiskirstymo tankį:

$$f_s(q) = p_a \delta\left(q - \frac{\rho_a}{\lambda}\right) + p_b \delta\left(q - \frac{\rho_b}{\lambda}\right) + \dots + p_f \delta\left(q - \frac{\rho_f}{\lambda}\right). \quad (10)$$

Šis ribinis pasiskirstymo tankis (10) leidžia įvertinti baigiamojo etapo tikslumą, pavyzdžiui, nustatyti tikimybę, kad paieškos metu simplekso centras neišeis už nustatytos srities ribų.



3 pav. Simplekso klaidžiojimo tikslo rajone teoriškai apskaičiuotas ribinis pasiskirstymo tankis, kai $k = 2$ ir $A/\sigma = 1$. Ištininė linija atitinka būsenas atpažįstantį algoritmą, trūkioji – laisvai viršūnes atspindintį algoritmą [1]

Simplekso klaidžiojimo tikslo rajone ribinio pasiskirstymo tankio verčių skaičiavimo rezultatai pateikti 3 pav. Trukdžiai ε – atsiktiniai dydžiai, pasiskirstę pagal normalųjį dėsnį su nuline matematine viltimi ir dispersija $\sigma_\varepsilon^2 = 1$. Palyginimui iš [1] pateikti laisvai viršūnes

atspindinčio simpleksinės paieškos algoritmo teorinių tyrimų rezultatai.

Siekiant patikrinti teorinių tyrimų rezultatus, pagal simpleksinės būsenas atpažįstančios paieškos algoritmą buvo sudaryta kompiuterinė programa. Eksperimento metu veikiantys trukdžiai – atsitiktiniai dydžiai, pasiskirstę pagal normalųjį dėsnį su nuline matematine viltimi ir dispersija $\sigma_\epsilon^2 = 1$ iš 2500 verčių lentelės [4]. Eksperimentinių tyrimų metu atlikta po 5000 paieškos žingsnių. Tokiomis pačiomis sąlygomis ištirtas ir laisvai viršūnes atspindintis simpleksinės paieškos algoritmas. Be to, abu algoritmai ištirti, kai trukdžiai yra atsitiktiniai tolygiai pasiskirstę dydžiai su nuline matematine viltimi ir dispersija $\sigma_\epsilon^2 = 1$, kurių vertės paimtos iš 6250 elementų lentelės [4]. Optimizuojamas rodiklis atitinka (4) lygtį, pradinis simpleksas orientuotas pagal 1 pav. – viena iš viršūnių yra ekstremume. Taigi galimų simplekso centro orbitų spinduliai, kai $k = 2$, yra lygūs:

$$\rho_a = R = \frac{L}{\sqrt{3}};$$

$$\rho_b = h + r = \frac{2L}{\sqrt{3}};$$

$$\rho_c = \sqrt{L^2 + R^2 - 2LR \cos \frac{5\pi}{6}} = \sqrt{\frac{7}{3}}L;$$

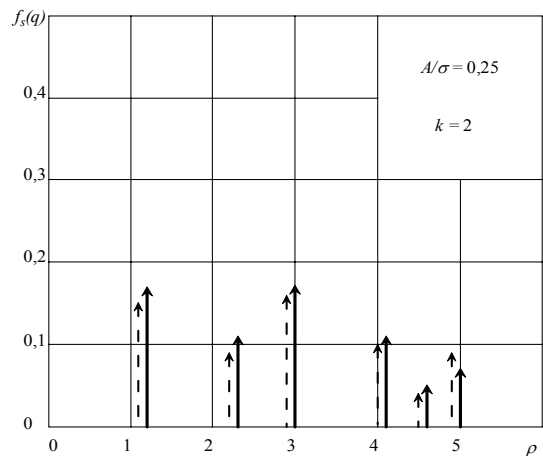
$$\rho_d = \sqrt{L^2 + (h+r)^2 - 2L(h+r) \cos \frac{5\pi}{6}} = \sqrt{\frac{13}{3}}L;$$

$$\rho_e = 2h + R = \frac{4}{\sqrt{3}}L;$$

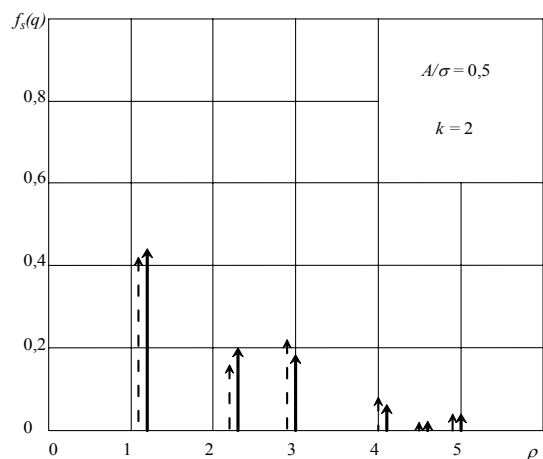
$$\rho_f = \sqrt{4L^2 + R^2 - 4LR \cos \frac{5\pi}{6}} = \sqrt{\frac{19}{3}}L.$$

Kai $L = 2$, $\rho_a = 1,165$, $\rho_b = 2,319$, $\rho_c = 3,065$, $\rho_d = 4,173$, $\rho_e = 4,629$, $\rho_f = 5,043$.

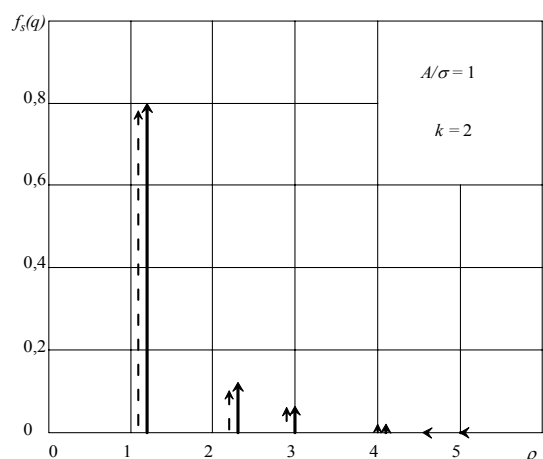
Analizuojant eksperimento rezultatus, išryškėjo paieškos tikslumo priklausomybė nuo (6) išraiškoje parenkamų dydžių – analizuojamų paieškos žingsnių skaičiaus ν , svorio koeficientų K_i bei slenkstinės vertės (formulėje ji lygi nuliui). Pavyzdžiui, esant normaliai pasiskirsčiusiems trukdžiams, geresni paieškos rezultatai gauti ($K_i = 1$), kai $\nu = 3$ ir slenkstis lygus -1 (4 pav., kai $A/\sigma = 0,25$, 5 pav., kai $A/\sigma = 0,5$, 6 pav., kai $A/\sigma = 1$), o esant tolygiai pasiskirsčiusiems trukdžiams, – kai $\nu = 2$ ir slenkstis lygus nuliui (7 pav., $A/\sigma = 0,25$). Be to, simpleksinės paieškos klaidžiojimo apie tikslą trukdžių sąlygomis atveju būtina įvesti sąlygą, padedančią išvengti sukimosi apie klaidingą ekstremumą (kai netoli ekstremumo esančią viršūnę paveikia stiprus trukdys). Ši sąlyga būtina tiek būsenas atpažįstančios paieškos algoritmui, tiek laisvai viršūnes atspindinčiam algoritmui. Algoritme tai gali būti įgyvendinta papildomai matuojant tikslo funkcijos vertę, jei viršūnė nebuvo atspindėta $k+n$ kartų. Pateikti rezultatai atitinka $n=2$, t. y. jei viršūnė nebuvo atspindėta per 4 paieškos žingsnius.



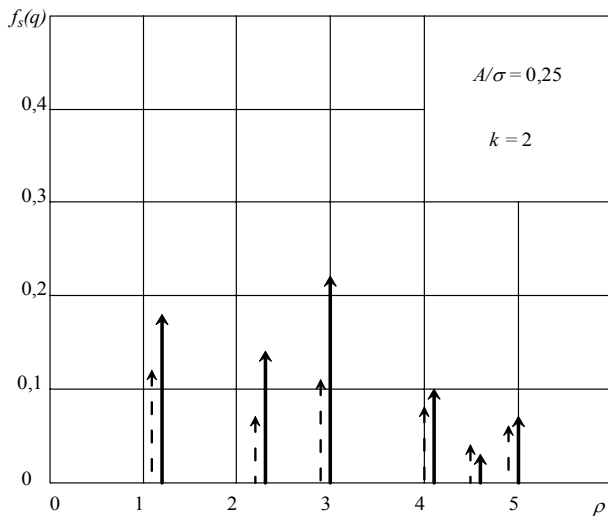
4 pav. Simplekso klaidžiojimo tikslo rajone ribinis pasiskirstymo tankis, kai $k = 2$ ir $A/\sigma = 0,25$. Ištininė linija atitinka būsenas atpažįstantį algoritmą, trūkioji – laisvai viršūnes atspindintį algoritmą. Trukdžiai – normaliai pasiskirstę atsitiktiniai dydžiai



5 pav. Simplekso klaidžiojimo tikslo rajone ribinis pasiskirstymo tankis, kai $k = 2$ ir $A/\sigma = 0,5$. Ištininė linija atitinka būsenas atpažįstantį algoritmą, trūkioji – laisvai viršūnes atspindintį algoritmą. Trukdžiai – normaliai pasiskirstę atsitiktiniai dydžiai



6 pav. Simplekso klaidžiojimo tikslo rajone ribinis pasiskirstymo tankis, kai $k = 2$ ir $A/\sigma = 1$. Ištininė linija atitinka būsenas atpažįstantį algoritmą, trūkioji – laisvai viršūnes atspindintį algoritmą. Trukdžiai – normaliai pasiskirstę atsitiktiniai dydžiai



7 pav. Simplekso klaidžiojimo tikslo rajone ribinis pasiskirstymo tankis, kai $k = 2$ ir $A/\sigma = 0,25$. Išsistinė linija atitinka būsenas atpažįstančią algoritmą, trūkioji – laisvai viršūnes atspindintį algoritmą. Trukdžiai – tolygiai pasiskirstę atsitiktiniai dydžiai

Pateikti tyrimų rezultatai įrodo simpleksinės būsenas atpažįstančios paieškos algoritmų tikslinimo etapui pranašumą palyginti su geriausiu šiam tikslui pripažintu [1] laisvai viršūnes atspindinčiu algoritmu, esant ypač aukštam trukdžių lygiui ($A/\sigma = 0,25 \div 1$).

Kombinuotosios būsenas atpažįstančios paieškos algoritmo analizė

Tyrimuose [2] gauti kombinuotosios simpleksinės paieškos pranašumai, esant stebimiems nevaldomiems parametrams bei būsenas atpažįstančios paieškos rodikliai, esant ypač aukštam trukdžių lygiui, leidžia sudaryti kombinuotosios simpleksinės būsenas atpažįstančios paieškos algoritmus. Pagal aprašytą kombinuotosios būsenas atpažįstančios paieškos algoritmų sudarymo metodiką sudarytas maksimumo paieškos algoritmas:

1. Aplik pradinį tašką x_{0i} , sudaryti simpleksą. Jo viršūnių koordinatės $x_{ji} = x_{0i} + \eta_{ji} L_0$. Visose pradinio simplekso viršūnėse surasti tikslo funkcijos vertės $y_j, j = 1, \dots, k+1$ ir $z_j, j = 1, \dots, k+1$. Išrinkti pradinio simplekso viršūnę v_s su mažiausia y_j verte, t. y. $y_s = \inf y_j$.

2. Atspindėti viršūnę v_s ir apskaičiuoti koordinates

$$x_{si} = \frac{2}{k} \sum_{j=1}^{k+1} x_{ji} - \frac{2+k}{k} x_{si}, \quad i = 1, \dots, k. \text{ Surasti}$$

$$y_s^H, y_j := y_s^H (j = s); z_s^H, z_j := z_s^H (j = s);$$

$$m_j := m_j + 1, m_s := 0, f := s, x_{ji} := x_{si}^H (j = s); n := n + 1.$$

3. Iš visų simplekso viršūnių, išskyrus naujai gautą v_f^H , išrinkti viršūnę v_s su mažiausia y_j verte, t. y.

$$y_s = \inf_{j \neq f} y_j, \quad j = 1, \dots, k+1.$$

4. Atspindėti viršūnę, kurios numeris s , t. y. surasti viršūnės x_{si}^H koordinates, $m_j := m_j + 1, j = 1, \dots, k+1; m_s := 0; n := n + 1.$

5. Surasti paskutinio simplekso, kurio numeris n , seniausios viršūnės numerį $r: m_r = \max \{m_j\}, j = 1, \dots, k+1.$

6. Netikrinant viršūnės, kurios numeris r , surasti kitą seniausią viršūnę, kurios numeris $p: m_p = \max_{j \neq r} \{m_j\}, j = 1, \dots, k+1.$

7. Naudojant priešpaskutinio simplekso (kurio numeris $n-1$) ir simplekso $n-2$ viršūnių duomenis surasti Δy ir Δz vertes $\Delta y = y_p^H + \frac{2}{k} y_p^H + y_p - \frac{2}{k} \sum_{j=1}^{k+1} y_j$

$$(\text{sumos } \sum_{j=1}^{k+1} y_j \text{ pavyzdys: } -\frac{2}{k} y_1^H - \frac{2}{k} y_2 - \frac{2}{k} y_3);$$

$$\Delta z = z_p^H + \frac{2}{k} z_p^H + z_p - \frac{2}{k} \sum_{j=1}^{k+1} z_j; \quad q = \frac{\Delta y}{\Delta z}.$$

8. Apskaičiuoti koreguotas tikslo funkcijos vertes simplekso $(n-2)$ viršūnėse $E_p = y_p - q \cdot z_p$, simplekso $(n-1)$ viršūnėse $E_p^H = y_p^H - q \cdot z_p^H$.

9. Surasti $y_s^H, y_j := y_s^H (j = s); z_s^H, z_j := z_s^H (j = s), x_{ji} := x_{si}^H (j = s).$

10. Išanalizuoti:

a) jeigu $E_p^H - E_p \geq 0$, tai $s := r$;

b) jeigu $E_p^H - E_p < 0$, tai $s := p$,

ir pereiti į 5 punktą.

Siekiant iširti šio kombinuotosios simpleksinės būsenas atpažįstančios paieškos algoritmo efektyvumą, jis modeliuotas, kai optimizuojamas objektas apibūdinamas matuojamu tikslo dreifu bei veikiamas aukšto lygio atsitiktinių trukdžių [2]. Sudaryta kompiuterinė programa, atliekanti paiešką pagal aprašytą algoritmą. Trukdžiai – atsitiktiniai dydžiai, pasiskirstę pagal normalųjį dėsnį; jų matematinė viltis lygi nuliui, o dispersija $\sigma^2 = 1$. Atliekant paiešką, trukdžių vertės imamos iš 2500 elementų lentelės [4]. Paieškos tyrimo rezultatai, kai tikslo dreifo greitis $V = 0,1$, o trukdžių lygis $A/\sigma = 0,5$, pateikti 8 pav. Palyginimui 9 pav. pateikti kombinuotosios uždraustos grįžties paieškos modeliavimo rezultatai tokiais pat sąlygomis. Iš šių diagramų akivaizdu, kad, esant matuojamam nedideliam tikslo dreifui bei aukštam trukdžių lygiui, kombinuotoji simpleksinė būsenas atpažįstanti paieška yra du kartus efektyvesnė už kombinuotąją uždraustos grįžties simpleksinę paiešką.

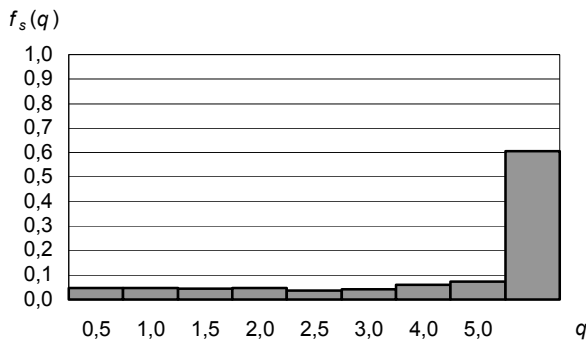
Iš atliktų teorinių tyrimų bei eksperimentų rezultatų galima daryti išvadą, kad įvairiomis paieškos sąlygomis skirtingi algoritmai pasižymi skirtingu efektyvumu – vieni geriau tinka esant vienoms paieškos sąlygoms, kiti – kitoms. Remiantis šia išvada, galima sudaryti kintamosios strategijos paieškines sistemas, kai paieškos metu yra optimizuojamas ir pats paieškos algoritmas.

Išvados

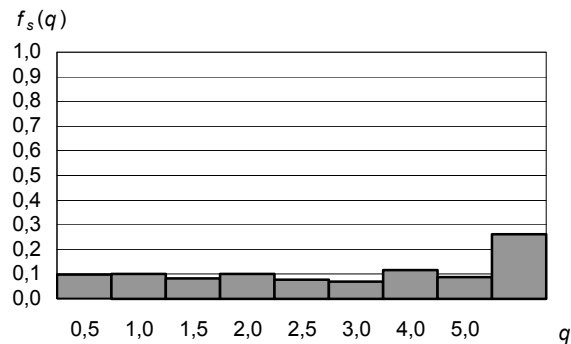
Būsenas atpažįstanti simpleksinė paieška pasižymi ypatingu efektyvumu, esant aukštam trukdžių lygiui:

1. Simpleksinės būsenas atpažįstančios paieškos algoritmas tikslinimo etapui už simpleksinės paieškos laisvai viršūnes atspindintį algoritmą esant tolygiai pasiskirsčiusiam trukdžiui, kai $A/\sigma_\varepsilon = 0,25$, yra 30 % tikslesnis (7 pav.).

2. Esant aukštam trukdžių lygiui ir nedideliame stebimam parametrai tikslinga naudoti kombinuotąją simpleksinę būsenas atpažįstančią paiešką: kai $k = 2$, $V = 0,1$ ir $A/\sigma_\varepsilon = 0,5$, kombinuotosios būsenas atpažįstančios paieškos algoritmas yra du kartus tikslesnis už kombinuotosios uždraustos grįžties paieškos (8 ir 9 pav.).



8 pav. Kombinuotosios simpleksinės uždraustos grįžties paieškos nuotolio nuo ekstremumo pasiskirstymas, kai $k = 2$, $V = 0,1$ ir $A/\sigma_\varepsilon = 0,5$



9 pav. Kombinuotosios simpleksinės būsenas atpažįstančios paieškos nuotolio nuo ekstremumo pasiskirstymas, kai $k = 2$, $V = 0,1$ ir $A/\sigma_\varepsilon = 0,5$

Literatūra

1. **Dambrauskas A.** Simpleksinės paieškos metodai. Vilnius: Technika, 1995. – P.192–202.
2. **Dambrauskas A., Udris D.** Statistical characteristics of combined simplex search // Information Technology and Control. – Kaunas, 2003. – Nr. 1(26). – P. 72–74.
3. **Дамбраускас А. П., Кушнир В. П.** Симплексный поиск с распознаванием состояния // Математическое обеспечение исследований стохастических и детерминированных моделей. – Новосибирск: НЭТИ, 1986. – С. 90-102.
4. **Большев Л. Н., Смирнов Н. В.** Таблицы математической статистики. – М.: Наука, 1983.

Pateikta spaudai 2004 03 01

A. Dambrauskas, D. Udris. Simpleksinės paieškos tyrimas tikslinimo etape // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2004. – Nr. 5(54). – P. 47–51.

Taikant daugiaryšes Markovo grandines, atlikta simpleksinės paieškos statistinių savybių tikslinimo etape analizė. Sudarytas simpleksinės būsenas atpažįstančios paieškos algoritmas tikslinimo etapui. Gautos teorinės statistinės charakteristikos – paieškos žingsnių tikimybės, simplekso klaidžiojimo tikslo rajone ribinis pasiskirstymo tankis ir kt. – leidžia įvertinti simpleksinės paieškos trukdžių aplinkoje tikslumą ir patikimumą. Atlikta kombinuotos būsenas atpažįstančios simpleksinės paieškos eksperimentinė analizė, proceso modeliavimo būdu gautos paieškos nuotolio nuo ekstremumo pasiskirstymo charakteristikos. Darbo rezultatai gali būti panaudoti kuriant naujus efektyvius simpleksinės paieškos algoritmus. Il. 9, bibl. 4 (lietuvių kalba; santraukos lietuvių, anglų ir rusų k.).

A. Dambrauskas, D. Udris. Analysis of Simplex Search within Specification Stage // Electronics and Electrical Engineering. – Kaunas: Technologija, 2004. – No. 5(54). – P. 47–51.

A synthesis method of simplex search with state recognition during the specification stage was presented. The statistical analysis of simplex search with state recognition was made using a multiple Markov chain to describe the search process in noisy surrounding. Results of experimental investigation of simplex search with state recognition were presented too. The investigation showed that simplex search with state recognition is more accurate than ordinary simplex search, especially when noise level is very high. An analysis of combined simplex search with state recognition during the specification stage was performed, and advantages were found, when measurable aim drift presents. These results can be used to found new effective algorithms of simplex search. Ill. 9, bibl. 4 (in Lithuanian, Summaries in Lithuanian, English and Russian).

А. Дамбраускас, Д. Удрис. Анализ симплексного поиска на этапе доводки // Электроника и электротехника. – Каунас: Технология, 2004. – № 5(54). – С. 47–51.

Произведен анализ статистических свойств симплексного поиска на этапе доводки, применяя многосвязные цепи Маркова. Разработан алгоритм симплексного поиска с распознаванием состояния для этапа доводки. Полученные теоретические вероятностные характеристики – вероятности шагов поиска, предельная плотность распределения блуждания симплекса в районе цели и др. – позволяют оценить точность и надежность симплексного поиска в обстановке помех. Приведен экспериментальный анализ комбинированного симплексного поиска с распознаванием состояния, способом моделирования процесса получены характеристики распределения расстояния поиска до экстремума. Результаты работы могут быть использованы при создании новых эффективных алгоритмов симплексного поиска. Ил. 9, библи. 4 (на литовском языке; рефераты на литовском, английском и русском яз.).

