LIETUVOS MATEMATIKOS RINKINYS. LMD DARBAI
52 tomas, 2011, 205–207
ISSN 0132-2818
www.mii.lt/LMR/

Nikolaj Grigorjev, Gediminas Stepanauskas

Vilniaus universitetas, Matematikos ir informatikos fakultetas
Naugarduko 24, LT-03225 Vilnius
E. paštas: nikolaj20@gmail.com; gediminas.stepanauskas@mif.vu.lt

Santrauka. Straipsnyje nagrinėjamas paskaitų tvarkaraščių konstravimo uždavinys naudojant Monte Karlo metodą. Sukurta tvarkaraščio optimizavimo programa remiasi vėsinimo (annealing) algoritmu.

Raktiniai žodžiai: Monte Karlo metodas, paskaitų tvarkaraščiai, Markovo grandinės.

Gero paskaitų tvarkaraščio sudarymas yra sudėtingas uždavinys, reikalaujantis ir daug laiko, ir geros kvalifikacijos. Tokie uždaviniai nėra nauji. Jiems spręsti naudoti globalaus optimizavimo metodai, tame tarpe ir genetiniai algoritmai bei Monte Karlo paieškos metodai [1]. Kai tvarkaraštis nėra perdaug sudėtingas, tokie metodai duoda patenkinamus rezultatus, jei sudėtingesnis, sprendimas reikalauja per didelio kompiuterio laiko.

Šio darbo tikslas – patyrinėti tvarkaraščio sudarymo procesą naudojant Markovo grandinių Monte Karlo metodą. Sudarinėsime savaitinius tvarkaraščius. Visą galimų tvarkaraščių aibę $\{\xi\}$ laikysime grafu. Grafo viršūnėmis bus galimi tvarkaraščiai. Kad galėtume pritaikyti taip vadinamą vėsinimo (annealing) procedūrą [2], turime taip apibrėžti kaimynus grafe, kad grafas būtų jungus. Nuo kaimynų apibrėžimo labai priklauso metodo efektyvumas. Geriau, kai grafo viršūnės turi nedaug kaimynų. Taigi kaimynais laikysime tokius tvarkaraščius, kurių dvi paskaitos sukeistos vietomis. Keitimas galimas tarp dviejų paskaitų ir tarp paskaitos bei „tuščios" paskaitos, t. y. „lango" (žr.1 lentelę).

Įveskime pažymėjimus. Tegul G yra studentų grupių skaičius, D – dėstytojų skaičius, a_i – i-osios studentų grupės „langų" visame tvarkaraštyje skaičius, $b_{k,i}$ – i-osios studentų grupės paskaitų k-ąją savaitės dieną skaičius, c_i – i-osios studentų grupės užimtų dienų savaitėje skaičius, u_i – i-ojo dėstytojo „langų" skaičius, $v_{k,i}$ – i-ojo dėstytojo paskaitų k-ąją savaitės dieną skaičius, w_i – i-ojo dėstytojo užimtų dienų savaitėje skaičius. Tvarkaraščio optimizavimui naudosime taip vadinamą tikslo

1 lentelė. Perėjimas prie kaimyninio tvarkaraščio.

Tvarkaraštis ξ						
Pask. 12	*Pask. 3*	Pask. 64	...	Pask. 31	*Pask. 37*	...

$\Downarrow$

Tvarkaraštis ξ'						
Pask. 12	*Pask. 37*	Pask. 64	...	Pask. 31	*Pask. 3*	...

funciją (galima būtų vadinti ir baudos funkcija):

$$
\begin{aligned}
f(\xi) &= \sum_{i=1}^{G}\left(5a_i + 20c_i + 50\sum_{k=1}^{5}\max(0, b_{k,i} - 4)\right) \\
&\quad + \sum_{i=1}^{D}\left(3u_i + 50w_i + 50\sum_{k=1}^{5}\max(0, v_{k,i} - 4)\right).
\end{aligned}
$$

Ieškosime šios funkcijos minimumo. Pagal šios funkcijos reikšmes galėsime spręsti apie tvarkaraščio gerumą. Geresni bus tie tvarkaraščiai ξ, kurių $f(\xi)$ reikšmės mažesnės.

Pats tvarkaraščio optimizavimo procesas vyksta taip. Atsitiktiniu būdu generuojamas (gali būti ir parenkamas) bet koks pradinis tvarkaraštis ξ. Su lygiomis tikimybėmis pasirenkamas vienas iš ξ kaimyninių tvarkaraščių ξ', t. y. atsitiktine tvarka sukeičiamos dvi paskaitos vietomis. Nuo tvarkaraščio ξ į kaimyninį tvarkaraštį ξ' pereinama su Bolcmano tikimybe

$$
p = \min\left\{\exp\left(\frac{f(\xi) - f(\xi')}{T}\right), 1\right\},
$$

čia parametras $T > 0$ vadinamas vėsinimo temperatūra. Jo reikšmės artinamos į 0. Artėjimo į 0 greitis parenkamas eksperimentuojant. Teoriniai rezultatai (žr. 1 teoremą) garantuoja, kad, T artinant į 0 pakankamai lėtai, visada surasime tvarkaraštį ξ, tokį, kad tikslo funkcija $f(\xi)$ įgytų jame absoliutų minimumą. Iš kitos pusės, jei T į 0 artėja labai lėtai, gero tvarkaraščio suradimas užtrunka labai ilgai. Garantinis laikas, per kurį galima surasti patį geriausią ξ, praktiškai nėra pritaikomas, nes per ilgas. Paprastai apsiribojama rezultatais, gautais per kažkokį realų laiką. Numatomas procedūros stabdymo laikas.

1 teorema [Geman, Geman [2]]. *Jei temperatūra $T = T_n$ n-uoju laiko momentu konverguoja į 0 pakankamai lėtai, t. y.*

$$
T_n \geqslant \frac{c(\max_{\xi\in\{\xi\}} f(\xi) - \min_{\xi\in\{\xi\}} f(\xi))}{\log n}
$$

visiems pakankamai dideliems n, tai tikimybė, kad n-uoju laiko momentu funkcija f įgis absoliutų minimumą, artėja į 1, kai $n \to \infty$.

Taigi, jeigu ξ' geresnis už ξ ($f(\xi) > f(\xi')$), pereiname prie tvarkaraščio ξ'. Jeigu ξ' blogesnis už ξ ($f(\xi) < f(\xi')$), tai su tikimybe $q = \exp(\frac{f(\xi)-f(\xi')}{T})$ pereiname prie ξ' ir su tikimybe $1-q$ liekame prie senojo tvarkaraščio ξ. Ir pagaliau, jeigu $f(\xi) = f(\xi')$, tai su tikimybe $1/2$ pereiname prie ξ' ir su tikimybe $1/2$ liekame prie senojo tvarkaraščio ξ. Kartais grįžtama į blogesnius tvarkaraščius tam, kad neliktume tvarkaraštyje, kuriame tikslo funkcija f įgyja lokalųjį minimumą. Pasirinkus tvarkaraštį ξ ar ξ', vėl renkamės kaimyną, ir eksperimentas kartojamas.

Bandymams buvo pasirinktas realus tvarkaraštis, t. y. VU Fizikos fakulteto tvarkaraštis. Pasirinkta pseudoatsitiktinė seka, sugeneruota tiesiniu kongruentiniu metodu, patikrinta keletu testų. Pradinis tvarkaraštis sugeneruotas atsitiktinis. Temperatūros T artėjimo į 0 greitis reguliuojamas etapais. Pradinė T reikšmė prilyginta 500.

2 lentelė. Eksperimento duomenys.

Stabdymo laikas	Tikslo funkcijos reikšmė skaičiavimų pabaigoje	Skaičiavimai užtruko
100	17 876	10 min.
1 000	16 500	24 min.
10 000	16 208	52 min.

Priimtini rezultatai gauti, kai temperatūra T nekeičiama 1000 žingsnių, po to sumažinama 30%, ir procesas kartojamas. Svarbus uždavinio sprendimo parametras yra stabdymo laiko nustatymas. Skaičiavimai buvo baigiami, kai po nustatyto žingsnių skaičiaus (stabdymo laiko) nesikeitė tikslo funkcijos reikšmė. Eksperimento duomenys pateikti 2 lentelėje, kai tikslo funkcijos reikšmė skaičiavimų pradžioje buvo lygi 24 859.

Paskaičiavę tikslo funkcijos reikšmę tikrąjam Fizikos fakulteto tvarkaraščiui, gavome 20 190. Matome, kad tai geresnis tvarkaraštis už atsitiktinai sugeneruotą, bet blogesnis už eksperimento metu gautus tvarkaraščius.

Tvarkaraščio konstravimo programa suprogramuota C++ kalba naudojant Microsoft® Visual C++® 2008 Express integruotą kūrimo aplinką. Bandymai buvo atliekami su Dell Inspiron kompiuteriu, turinčiu Intel i7-2630QM 2.0 Ghz procesorių bei 6 GB 1333 MHz DDR3 laisvosios prieigos atminties.

Literatūra

[1] M. Balnys. *Genetinių algoritmų pritaikymo klasifikavimo uždaviniams spręsti tyrimas*. Informatikos mokslų magistro baigiamasis darbas, KTU IF Kompiuterių katedra, 2004. Adresas internete: http://vddb.library.lt/fedora/get/LT-eLABa-0001:E.02 2004 D_20040528_155652-64676/DS.005.0.01.ETD.

[2] O. Häggström. *Finite Markov Chains and Algorithmic Aplications*. Cambridge University Press, 2002.

SUMMARY

Schedules of lectures and Monte Carlo method

N. Grigorjev, G. Stepanauskas

In this paper the problem of the construction of schedule of lectures is considered. The Markov chain Monte Carlo method is used. A particular program based on simulated annealing algorithm was created.

Keywords: Monte Carlo method, schedules of lectures, Markov chains.