ზოგიერთი რიცხვითი ექსპერიმენტი სიმეტრიულ თამაშებზე

კობა გელაშვილი

koba.gelashvili@tsu.ge

კომპიუტერული მეცნიერების დეპარტამენტი
ივანე ჯავახიშვილის სახელობის თბილისის სახელმწიფო უნივერსიტეტი
უნივერსიტეტის ქ. 13, თბილისი 0186, საქართველო

არსებობის 80 წლის მანძილზე წრფივი პროგრამირება (LP) იქცა მძლავრ იარაღად თეორიულ კომპიუტერულ მეცნიერებასა და ალგორითმების აგებაში. ამჟამად, LP ეფექტურად ხსნის მთელრიცხვა პროგრამირების, გრაფთა თეორიის და სხვა ამოცანებს და ფართოდ გამოიყენება რთული ამოცანების მიახლოებითი ალგორითმების ასაგებად. ამას გარდა, მრავალი გამოთვლითი ამოცანისთვის, ეფექტური ალგორითმების დამტკიცებები ეფუძნება LP-ს.

წრფივ პროგრამირებას აქვს რამდენიმე ეკვივალენტური ჩამოყალიბება. ჩვენ ექსპერიმენტებს ვატარებთ სიმეტრიულ თამაშებზე. ამ შემთხვევაში, შესაძლებელია საჯარიმო ფუნქციების გამოყენება ვრცელი სპექტრის მქონე უპირობო მინიმიზაციის ამომხსნელ პროგრამებთან ერთად. მაგალითად, როგორიცაა l-bfgs (იხ.[1]) ან მოდიფიცირებული მძიმე ბრითვი (იხ. [2]). შედეგად ვიღებთ ალგორითმს, რომელიც კონკურენტუნარიანია ღია კოდის მქონე და კომერციულ ამომხნელ პროგრამებთან. ამ მიდგომაში მთავარი სირთულე არის შეზღუდვა ცვლადის ნიშანზე და ჩვენ ორ ეტაპად ვხსნით ამ ამოცანას. უხეშად, მაგრამ გარკვეული დაწვრილმანებით განვიხილოთ ეს საკითხი.

ირიბსიმეტრიული A მატრიცის მქონე სიმეტირული თამაშის ამოხსნა ეკვივალენტურია შემდეგი სისტემის

$$\begin{cases} A_0 x \le 0, \dots, A_{n-1} x \le 0 \\ x_0 + \dots + x_{n-1} = 1 \\ x_0 \ge 0, \dots, x_{n-1} \ge 0 \end{cases}$$

ამოხსნის. შეზღუდვების მიღწევადობაზე ავიღოთ ნაგულისხმევი სიზუსტე 1E-6.

პირველ ეტაპზე, ამოცანას ვხსნით ჯარიმით:

$$P_1(x) = \frac{1}{3}\left(\sum_{i=0}^{n-1} (A_i x \le 0)\,?\,0 : (A_i x)^3 + 100 \cdot \sum_{i=0}^{n-1} (x_i \ge 0)\,?\,0 : (-x_i)^3 + (x_0 + \dots + x_{n-1} - 1)^3 \right)$$

ოღონდ სიზუსტით 1E-7. შემდეგ განვაახლებთ საწყის იტერაციას ცვლადის გარდაქმნით:

$$\texttt{if}(\, x_i > 0 \,) \quad x_i = \log(x_i)/10\,;$$

$$\texttt{else} \quad x_i = \log(1\mathrm{E}-10);$$

და ხელახლა ვხსნით ნაგულისხმევი სიზუსტით და ჯარიმით

$$P_2(t) = \frac{1}{3}\left(\sum_{i=0}^{n-1} A_i x \le 0\,?\,0 : (A_i x)^3 + \left(e^{10t_0} + \dots + e^{10t_{n-1}} - 1\right)^3 \right).$$

ამჯერად, მეორე ეტაპს აქვს მცირე ზედნადები ხარჯი და საწყის ცვლადებში დაბრუნების შემდეგ ვღებულობთ ამონახსნს მხოლოდ არაუარყოფითი კომპონენტებით.

ლიტერატურა:

1. Jorge Nocedal Stephen J. Wright. Numerical Optimization, Second Edition, Springer, 2006.
2. Koba Gelashvili, Irina Khutsishvili, Luka Gorgadze, Lela Alkhazishvili. Speeding up the convergence of the Polyak's Heavy Ball algorithm (in print)