Simplexin kritiikki

Aleksi Harri

Laskettavuus

- Laskettavuus tutkii ongelmien ratkeavuutta ja ratkaisemisen tehokkuutta algoritmisesti.
- Simplexin laajan käytettävyyden pohjana on tietokoneiden kyky ratkaista algoritmi.
- Käsittelemme ratkeavia algoritmeja.

Ajalliset rajoitukset

- Ratkeavuus ei takaa käytännöllisyyttä – algoritmin ratkaisuun vaadittava aika voi olla liian pitkä.
- Esim. kauppamatkustajan ongelman ratkaisu 50 kaupungilla vie miljardeja vuosia.
- Algoritmin operaatioihin tarvittava aika riippuu syötteestä – keskitymme pahimpaan mahdolliseen tapaukseen.

Monimutkaisuus

- Algoritmin laskennallista monimutkaisuutta tutkittaessa olemme yleensä kiinnostuneita sen kasvunopeudesta.
- $O(g(n))$ – ilmoittaa maksimin
- $\Omega(g(n))$ – ilmoittaa minimin
- $\Theta(g(n))$ – minimi ja maksimi likimain samat

Tapauksen koko

- Mittaamme algoritmien kompleksisuutta sen syötteen koon funktion mukaan.
- Kombinatoristen optimointiongelmien syöte on kombinatorinen.
- Graafi, kokonaislukujoukko, jne.
- Syötteet muunnetaan symboleiksi, tarkalla tavalla ei yleensä ole merkitystä.

Kokoja

- Kokonaisluku n – $\Theta(\log n)$.
- m x n lineaarisen ohjelma – $\Theta(mn + \log|P|)$, missä P on nollasta poikkevien kertoimien tulo.
- Graafin $G=(V,E)$ – $\Theta(|E|)$.

Algoritmien analyysiä

- Algoritmien maksimiratkaisuajan johtaminen ei ole aina ongelmatonta.
- Simplexin tapauksessa aikaa kuluu alustukseen ja iteraatioihin. Jos A on m x n-matriisi, niin alustukseen menee O(nm) ja iteraatioon O(mn).
- Pahimmassa tapauksessa iteraatioita voi tulla melkein $\binom{m+n}{n}$ kappaletta.

Koska ongelma on ratkaistu?

- Riippuu tunnettujen algoritmien suorituskyvystä.
- Täytyy löytyä algoritmi, joka ratkaisee ongelman tarpeeksi nopeasti.
- Kasvunopeuden pitää olla tarpeeksi pieni.
- Mikä kasvunopeus on hyväksyttävä?

Polynomiaikaiset algoritmit

- Polynomiaikaisten algoritmien monimutkaisuus kasvaa polynomisesti, tai on rajoittunut siten.
- Esim. $O(n)$ ja $O(n^3)$, sekä $O(n^{2,5})$ ja $O(n \log n)$.
- Polynomiset algoritmit voivat käyttää toisia polynomisia algoritmeja ja yhdistelmä on myöskin polynominen.

Eksponentiaaliset algoritmit

- Eksponentiaaliset algoritmit kasvavat nopeasti, kuten $O(2^n)$.
- Laskentatehon lisäys nopeuttaa polynomisten algoritmien laskua paljon eksponentiaalisten laskua enemmän.
- Polynomiaikaiset algoritmit ovat käytännön kokemuksen mukaan parempia ongelmien ratkaisussa.

Simplexin kritiikki

- Simplex ei ole polynomiaikainen algoritmi.
- Voidaan joutua käyttämään eksponentiaalinen määrä iteraatioita.
- Löytyykö LP:n ratkaisuun polynomiaikainen algoritmi?

LP:n ratkaisu polynomiaikaisesti

- Vuonna 1979 julkaistu ellipsoidi-algoritmi oli ensimmäinen joka ratkaisi lineaarisen ohjelman polynomiaikaisesti.
- Simplex ja sen variantit, sekä sisäpistemetodit ovat kuitenkin paljon sitä nopeampia, niin teoriassa kuin käytännössäkin.
- Teoreettisesta ongelmasta huolimatta, simplex on käytännössä pitänyt pintansa.

Kysymyksiä?