

Paikalliset (stokastiset) hakumenetelmät

Käsiteltäviä asioita:

1. Paikalliset hakumenetelmät
2. Toteutuvuus haku- ja optimointiongelmana
3. Paikallinen haku toteutuvuusongelmassa
4. Paikalliseen hakuun perustuvia algoritmeja:
 - Anhe paikallinen haku
 - Simuloitu jäähditys
5. Ahneen paikallisen haun muunnelmia
6. Kokeellisia vertailuja

2. Toteutuvuus haku- ja optimointiongelmana

- Hakuongelma (**SAT**): löydettävä annetulle lausejoukolla malli.
- Optimointiongelma (**MAX-SAT**): löydettävä annetulle lausejoukolla totuusjakelu, jossa mahdollisimman moni joukon lauseista on tosi.
- Tyypillisesti käsitellään konjunkttiivisessa normaalimuodossa olevia lausejoukkoja (klausuulijoukkoja).

Esimerkki. Olkoon joukko $S = \{A \vee B, \neg A \vee B, A \vee \neg B\}$.

- Joukon S toteutuvuusongelmalle löytyy ratkaisu $\{A, B\}$.
- Joukko $S \cup \{\neg A \vee \neg B\}$ on toteutumaton, mutta \emptyset , $\{A\}$, $\{B\}$ ja $\{A, B\}$ ovat vastaavan optimointiongelman ratkaisuja.

1. Paikalliset hakumenetelmät

- Paikalliset hakumenetelmät kuten *simuloitu jäähditys* ovat suosittuja kombinatoristen (optimointi)ongelmien ratkaisemisessa.
- Paikallisen haun perusideat ovat seuraava:
 - (i) Aloitetaan satunnaisesti generoidusta ratkaisuehdokkaasta.
 - (ii) Pyritään etenemään paikallisin muutoksin kohti parempaa ratkaisuehdokasta.
 - (iii) Tehdään paikallisia muutoksia joskus myös satunnaisesti, mikä saattaa huonontaa ratkaisuehdokasta.
- Paikalliset hakumenetelmät ovat *epätäydellisiä*: ratkaisun löytäminen ei ole taattua, vaikka sellainen on olemassa.

3. Paikallinen haku toteutuvuusongelmassa

- Paikallinen hakumenetelmä toteutuvuusongelman tapauksessa:
 - (i) Aloitetaan satunnaisesti generoidusta totuusjaketelusta.
 - (ii) Suoritetaan paikallisia muutoksia vaihtamalla atomilauseiden totuusarvoja (engl. termi *flip*).
 - (iii) *Ahne haku*: muutetaan totuusarvo atomilauseelta, joka johtaa suurimpaan laskuun epätosien klausuulien määrässä.
- **Ongelma**: miten päästään pois *paikallisista minimeistä* (totuusjaketelusta, joissa minkään atomilauseen totuusarvon muuttaminen ei laske epätosien lauseiden määrää)?
 - (a) Haun aloittaminen uudelleen
 - (b) Sivuttaissiirrot (epätosien klausuulien määrä ei laske)
 - (c) Satunnaiset siirrot (epätosien klausuulien määrä voi kasvaa)

Esimerkki. Tarkastellaan klausuulijoukon $S = \{A \vee B, \neg A \vee B, A \vee \neg B\}$ osoittamista toteutuvaksi paikallisella haulalla:

Satunnainen totuusjaku: Aloitetaan uudelleen:

$$\{\neg A, \neg B\}$$

(1)

$$\{A, \neg B\} \mid \{\neg A, B\}$$

(1)

Paikallinen minimi.

$$\{\neg A, B\}$$

(1)

$$\{A, B\} \mid \{\neg A, \neg B\}$$

(0)

(1)

Malli löytyi.

- Yllä totuusjakelulle \mathcal{A} on käytetty *literaaliesitystä*
 $\text{lit}(\mathcal{A}) = \mathcal{A} \cup \{\neg A \mid A \in \mathcal{P} - \mathcal{A}\}$.
- Joukon S epätosien klausuulien lukumäärä on annettu suluissa.

4. Paikalliseen hakuun perustuvia algoritmeja

- *Ahne* paikallinen haku (**GSAT**)
B. Selman, H.J. Levesque, and D.G. Mitchell [AAAI, 1992]:
A New Method for Solving Hard Satisfiability Problems.
- *Simuloitu jäähditys* (**SA**)
D.S. Johnson, C.R. Aragon, L.A. McGeoch, and C. Schevon [Operations Research, 1991]:
Optimization by Simulated Annealing: an Experimental Evaluation; part II, Graph Coloring and Number Partitioning.
W.M. Spears [Technical Report, 1993]:
Simulated Annealing for Hard Satisfiability Problems.

Esimerkki. Tarkastellaan klausuulijoukkoa

$$S = \{ A \vee \neg B, A \vee \neg C, B \vee \neg A, B \vee \neg C, \\ C \vee \neg A, C \vee \neg B, \neg A \vee \neg B \}.$$

- Totuusarvomuutokset voivat johtaa aina huonompiin malleihin:

$$\{A, B, C\}$$

(1)

$$\{A, \neg B, C\} \mid \{\neg A, B, C\} \mid \{A, B, \neg C\}$$

(2)

(2)

(3)

☞ Turvaudutaan tarvittaessa satunnaiseen siirtymään.

Joitain määritelmiä

Olkoon \mathcal{A} totuusjaku, S klausuulijoukko ja P atomilause.

- $\text{Hb}(S)$ on klausuulijoukossa S esiintyvien atomilauseiden joukko.
- $\text{Flip}(\mathcal{A}, P)$ on totuusjaku, joka eroaa totuusjakelusta \mathcal{A} ainoastaan atomilauseen P osalta.
- $\text{Random}(p)$ on funktio, joka palauttaa totuusarvon tosi todennäköisyydellä $0 \leq p \leq 1$.
- $\text{RC}(J)$ on funktio, joka palauttaa satunnaisen alkion joukosta J .
- $\text{False}(S, \mathcal{A}) = |\{C \in S \mid \mathcal{A} \not\models C\}|$, joten $\text{False}(S, \mathcal{A}) = 0 \implies \mathcal{A} \models S$.
- $\Delta(\mathcal{A}, P) = \text{False}(S, \text{Flip}(\mathcal{A}, P)) - \text{False}(S, \mathcal{A})$.

Ahne paikallinen haku (GSAT)

Function GSAT(S):

for $i := 1$ to MAX-TRIES **do**

$\mathcal{A} := \text{RC}(2^{\text{Hb}(S)});$

for $j := 1$ to MAX-FLIPS **do**

if $\mathcal{A} \models S$ **then** return \mathcal{A}

else

$P := \text{RC}(\{Q \in \text{Hb}(S) \mid \Delta(\mathcal{A}, Q) = \min_{R \in \text{Hb}(S)} \Delta(\mathcal{A}, R)\});$

$\mathcal{A} := \text{Flip}(\mathcal{A}, P)$

endif

endfor

endfor

Huom! $\text{RC}(2^{\text{Hb}(S)})$ voidaan toteuttaa lineaarisessa ajassa ja tilassa.

Arviointia (GSAT)

- Valittava parametrit MAX-TRIES ja MAX-FLIPS.
- Atomilauseiden satunnainen valinta tekee silmukat epätodennäköisiksi.
- Suorituskyky ilman sivuttaissiirtoja selvästi huonompi.
- Ahneet siirrot voivat viedä harhaan; pahimmillaan päädytään toistuvasti samaan paikalliseen minimiin aloitettaessa haku uudelleen.

☞ Tulisi sallia myös satunnaiset siirrot, joilla epätosien klausuulien lukumäärä $\text{False}(S, \mathcal{A})$ saattaa jopa kasvaa.

Esimerkki. Tarkastellaan klausuulijoukkoa $S =$

$$\{ A \vee B \vee C, \quad A \vee B \vee \neg C, \quad A \vee \neg B \vee C, \quad A \vee \neg B \vee \neg C, \\ \neg A \vee B \vee C, \quad \neg A \vee \neg B \vee C, \quad \neg A \vee B \vee \neg C \}.$$

Ahne paikallinen haku voi edetä esim. seuraavasti:

$$\{\neg A, \neg B, \neg C\} \quad (1)$$

| (A)

$$\{A, \neg B, \neg C\} \quad (1)$$

| (B)

$$\{A, B, \neg C\} \quad (1)$$

| (C)

$$\{A, B, C\} \quad (0)$$

☞ GSAT-algoritmi mahdollistaa sivuttaissiirrot.

Simuloitu jäähdytys (SA)

Function SA(S, T):

$\mathcal{A} := \text{RC}(2^{\text{Hb}(S)});$

repeat

if $\mathcal{A} \models S$ **then** return \mathcal{A}

else

$P := \text{RC}(\text{Hb}(S));$

if $\Delta(\mathcal{A}, P) \leq 0$ **or** $\text{Random}(e^{-\Delta(\mathcal{A}, P)/T})$ **then**

$\mathcal{A} := \text{Flip}(\mathcal{A}, P)$

endif;

$T := \text{Cool}(T)$

endif

until False

Esimerkki. Palataan edellisen esimerkin klausuulijoukkoon $S =$

$$\{ A \vee B \vee C, \quad A \vee B \vee \neg C, \quad A \vee \neg B \vee C, \quad A \vee \neg B \vee \neg C, \\ \neg A \vee B \vee C, \quad \neg A \vee \neg B \vee C, \quad \neg A \vee B \vee \neg C \}.$$

Simuloituun jäähtytykseen perustuva haku voi edetä esim. seuraavasti:

$$\begin{array}{ll} \{A, B, \neg C\} & (1) \\ | (B) & \Delta(\{A, B\}, B) = 0 \\ \{A, \neg B, \neg C\} & (1) \\ | (A) & \Delta(\{A\}, A) = 0 \\ \{\neg A, \neg B, \neg C\} & (1) \\ \vdots & \end{array}$$

Satunnaissiirtymät eivät tule kysymykseen tässä esimerkissä!

- Jäähdytysaikatauluja:
 - Geometrinen aikataulu: $\text{Cool}(T) = c * T$, missä $0 < c < 1$.
 - Lämpötila vakio: $\text{Cool}(T) = T$.
- Äärellinen jäähtytysaikataulu ei takaa ratkaisun löytymistä, joten joudutaan käyttämään uudelleen aloittamista.
- Keskeisiä eroja **GSAT**:iin nähden:
 1. **GSAT** tekee *aina* muutoksen, joka vähentää epätosien klausuulien määrää, jos sellainen on olemassa.
 2. Muuttujan (alustava) valinta on **SA**:ssa satunnainen. Vaikka $\Delta(\mathcal{A}, P) < 0$, voi löytyä toinen atomilause Q siten, että $\Delta(\mathcal{A}, Q) < \Delta(\mathcal{A}, P)$. Tällaisessa tilanteessa **GSAT** ei valitsisi atomilauseita P .

Arviointia (SA)

- Parametri T (lämpötila) määrää todennäköisyyden sille, milloin huonompaan suuntaan johtava muutos tehdään.

| T | $\Delta(\mathcal{A}, P)$ | $e^{-\Delta(\mathcal{A}, P)/T}$ |
|-----|--------------------------|---------------------------------|
| 5 | 1 | 0.819 |
| 0.5 | 1 | 0.135 |
| 0.5 | 2 | 0.0183 |
| 0.2 | 1 | 0.00674 |
| 0.2 | 2 | 0.0000454 |

- Valittava tapa lämpötilan T vähentämiseksi (jäähdytysaikataulu).

5. Ahneen paikallisen haun muunnelmia

- **GSAT** ja satunnainen *vaellus* (**RW**)
B. Selman, and H.A. Kautz [IJCAI, 1993]:
Domain-Independent Extensions to GSAT: Solving Large Structured Satisfiability Problems.
- **GSAT** ja satunnainen *kohina* (**RN**)
- **GSAT** ja satunnainen vaellus II (**WSAT**)
B. Selman, H.A. Kautz, and B. Cohen [DIMACS, 1993]:
Local Search Strategies for Satisfiability Testing.
<http://www.cs.washington.edu/homes/kautz/walksat/>
(nykyinen versionumero 45)

GSAT ja satunnainen vaellus (RW)

Function GSAT+RW(S, p):
for $i := 1$ to MAX-TRIES **do**
 $\mathcal{A} := \text{RC}(2^{\text{Hb}(S)});$
for $j := 1$ to MAX-FLIPS **do**
if $\mathcal{A} \models S$ **then** return \mathcal{A}
else
if Random(p) **then** $P := \text{RC}(\text{Hb}(\{C \in S \mid \mathcal{A} \not\models C\}))$
else $P := \text{RC}(\{Q \mid \Delta(\mathcal{A}, Q) = \min_R \Delta(\mathcal{A}, R)\})$ **endif**;
 $\mathcal{A} := \text{Flip}(\mathcal{A}, P)$
endif
endfor
endfor

© 2005 TKK, Tietojenkäsittelyteorian laboratorio

GSAT+RW (arviointia)

- Vaatii parametrinä olevan todennäköisyyden p valinnan.
- Tyypillisesti $0.5 \leq p \leq 0.6$.
- Huomaa satunnaisen vaeltamisen tavoitehakuisuus: mahdollisuuksien suuntaan (epätosien klausuulien määrä kasvaa) tapahtuvat siirtymät liittyvät vielä epätosiin klausuuleihin.

Esimerkki. Olkoon

$$S = \{A \vee \neg B, A \vee \neg C, B \vee \neg A, \neg B \vee \neg C\}.$$

Totuusjakeleen $\{A, B, C\}$ tapauksessa satunnainen siirtymä on mahdollinen vain atomilauseille B ja C .

© 2005 TKK, Tietojenkäsittelyteorian laboratorio

 Satunnaisen vaeltamisen ja ahneen haun yhdistelmä.

Esimerkki. Olkoon $S = \{A \vee E, \neg A \vee B, \neg B \vee C, \neg C \vee D \vee E\}$.

Kun $p = 1$ (ei ahneita siirtoja), haku voi edetä esim. seuraavasti:

$$\{\neg A, \neg B, \neg C, \neg D, \neg E\} \quad (1)$$

| (A)

$$\{A, \neg B, \neg C, \neg D, \neg E\} \quad (1)$$

| (B)

$$\{A, B, \neg C, \neg D, \neg E\} \quad (1)$$

| (C)

$$\{A, B, C, \neg D, \neg E\} \quad (1)$$

| (E)

$$\{A, B, C, \neg D, E\} \quad (0)$$

© 2005 TKK, Tietojenkäsittelyteorian laboratorio

GSAT ja satunnainen kohina (RN)

Function GSAT+RN(S, p):
for $i := 1$ to MAX-TRIES **do**
 $\mathcal{A} := \text{RC}(2^{\text{Hb}(S)});$
for $j := 1$ to MAX-FLIPS **do**
if $\mathcal{A} \models S$ **then** return \mathcal{A}
else
if Random(p) **then** $P := \text{RC}(\text{Hb}(S))$
else $P := \text{RC}(\{Q \mid \Delta(\mathcal{A}, Q) = \min_R \Delta(\mathcal{A}, R)\})$ **endif**;
 $\mathcal{A} := \text{Flip}(\mathcal{A}, P)$
endif
endfor
endfor

 Yhdistää puhtaasti satunnaiset muutokset ja ahneen haun.

© 2005 TKK, Tietojenkäsittelyteorian laboratorio

GSAT ja satunnainen vaellus II (WSAT)

Function $WSAT(S, p)$:

for $i := 1$ to MAX-TRIES do

$\mathcal{A} := RC(2^{Hb(S)});$

 for $j := 1$ to MAX-FLIPS do

 if $\mathcal{A} \models S$ then return \mathcal{A}

 else

$C := RC(\{D \in S \mid \mathcal{A} \not\models D\});$

 if Random(p) then $P := RC(Hb(C))$

 else $P := RC(\{Q \in Hb(C) \mid \Delta(\mathcal{A}, Q) = \min_{R \in Hb(C)} \Delta(\mathcal{A}, R)\})$ endif

$\mathcal{A} := Flip(\mathcal{A}, P)$

 endif

 endfor

endfor

WSAT (arviointia)

- Vaatii parametrinä olevan todennäköisyyden p valinnan.
- **WSAT** yhdistää satunnaisen vaeltamisen ja ahneen haun.
- Huomaa ero verrattuna algoritmiin **GSAT+RW**:
Esim. kun $p = 1$ (ei ahneita siirtoja), algoritmit toimivat eri tavoin: **WSAT** suosii satunnaisissa siirroissa atomilauseita, jotka esiintyvät useissa epätosissa klausuuleissa.

Esimerkki. Olkoon

$$S = \{A \vee E, E \vee B, \neg B \vee C, \neg A \vee D, \neg E \vee C \vee D\}.$$

Kun $p = 1$ (ei ahneita siirtoja), **WSAT** voi edetä esim. seuraavasti:

$$\{\neg A, \neg B, \neg C, \neg D, \neg E\} \quad (2)$$

$$\quad | (E) \quad A \vee E \text{ epätosi}$$

$$\{\neg A, \neg B, \neg C, \neg D, E\} \quad (1)$$

$$\quad | (C) \quad \neg E \vee C \vee D \text{ epätosi}$$

$$\{\neg A, \neg B, C, \neg D, E\} \quad (0)$$

6. Kokeellisia vertailuja

- Vertailut perustuvat usein satunnaisesti generoituihin esimerkkiluokkiin.
- Parhaat variantit **GSAT+RW** ja **WSAT** [SKC, 1993] (**WSAT** näyttäisi olevan näistä kahdesta parempi).
- Kumpikin tehokkaampia kuin **SA** (simuloitu jäädytys), joka on tehokkaampi kuin **GSAT** [Spears, 1993].
- **GSAT+RW** parempi kuin **GSAT+RN**.
- Joskus paikalliset hakumenetelmät ratkaisevat paljon suurempia ongelmia kuin täydelliset menetelmät (esim. Davis-Putnam).

Johtopäätöksiä

- Paikalliset hakumenetelmät suoraviivaisia toteuttaa.
- Toteutuvuusongelmaan suunnitelluilla menetelmillä parempi suorituskyky kuin yleisemmillä.
- Hyvät parametrien arvot oleellisia suorituskyvyn kannalta. Esim. **WSAT**:issa satunnaisen muutoksen todennäköisyys p tai SA:ssa jäähtytysaikataulu (*parametrien viritysongelma*).
- Paikallisilla hakumenetelmillä ongelmia, kun tehtävässä rakennetta (atomilauseiden välillä monimutkaisia riippuvuuksia).
- *Paikalliset hakumenetelmät eivät ole täydellisiä:*
 - (a) ne eivät takaa ratkaisun löytymistä;
 - (b) ne eivät kykyne päättämään, ettei ratkaisua ole olemassa;
 - (c) ne eivät takaa ratkaisun optimaalisuutta.

Oppimistavoitteet

- Ymmärrät paikallisen haun peruseriaatteet
- Tunnet muutamia perusalgoritmeja, jotka ratkaisevat lauselogiikan toteutuvuusongelmaa näillä periaatteilla
- Osaat simuloida näiden algoritmien toimintaa
- Tiedät mitä algoritmien epätäydellisyydestä seuraa
- Tiedät mistä algoritmien toteutuksia on saatavissa