Toteutuvuusongelman ratkaiseminen paikallisilla hakumenetelmillä

- Paikalliset hakumenetelmät (esim. simuloitu jäähdytys) suosittuja kombinatoristen (optimointi)ongelmien ratkaisemisessa.
- Idea: (i) Lähdetään liikkeelle satunnaisesti generoidusta ratkaisuehdokkaasta, (ii) pyritään paikallisin muutoksin etenemään kohti parempaa ratkaisuehdokasta, (iii) tehdään paikallisia muutoksia myös satunnaisesti (ei aina suuntaan, joka johtaa paikallisesti parempaan ratkaisuehdokkaaseen).
- Paikalliset hakumenetelmät ovat *epätäydellisiä*:
  ne eivät takaa, että ratkaisu löytyy (vaikka sellainen on olemassa).

Toteutuvuusongelma

- Hakuongelma (**SAT**): löydettävä annetulle lausejoukolle malli.
- Optimointiongelma (**MAX-SAT**): löydettävä annetulle lausejoukolle malli, jossa mahdollisimman moni joukon lause on tosi.
- Tyypillisesti käsitellään konjunktiivisessa normaalimuodossa olevia lausejoukkoja (klausuulijoukkoja).

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

Joukon S toteutuvuusongelmalle ratkaisuja ovat esim. mallit $\{A, B\}$ ja $\{A, B, C\}$.

Joukon $S \cup \{\neg A \vee \neg B\}$ toteutuvuusongelmalle ei ole ratkaisua mutta optimointiongelmalle löytyy ratkaisuksi esim. mallit $\{A, B\}$ ja $\{B\}$

Paikallinen haku toteutuvuusongelmassa

- Idea: (i) Aloitetaan satunnaisesti generoidusta mallista.
  (ii) Suoritetaan paikallisia muutoksia vaihtamalla atomilauseiden totuusarvoja (flip).
  (iii) Ahne haku: muutetaan totuusarvo sellaiselta atomilauseelta, joka johtaa suurimpaan laskuun epätosien lauseiden määrässä.
- Ongelma (tyypillinen paikallisille hakumenetelmille):
  miten päästään pois paikallisista minimeistä (malleista, joissa mikään totuusarvon muutos ei laske epätosien lauseiden määrää)?
  (a) Uudelleen aloitus
  (b) Sivuttaissiirrot (epätosien lauseiden määrä ei laske)
  (c) Satunnaiset siirrot (epätosien lauseiden määrä saattaa kasvaa)

Esimerkki. $S = \{A \vee B, \neg A \vee B, A \vee \neg B\}$.
(Merkitään malli literaalijoukkona ja sulkuihin epätosien lauseiden määrä.)

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

$\{A, \neg B\}$ (1) | $\{\neg A, B\}$ (1)

Paikallinen minimi

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

$\{A, B\}$ (0) | $\{\neg A, \neg B\}$ (1)

Uudelleen aloitus: malli löytyy

Esimerkki. On mahdollista, että jäljellä on vain nykyistä huonompia tilanteita.

$S =$		$\{A, B, C\}$	
$\{A \vee \neg B,$		(1)	
$A \vee \neg C,$	$\{A, \neg B, C\}$	$\{\neg A, B, C\}$	$\{A, B, \neg C\}$
$B \vee \neg A,$	(2)	(2)	(3)
$B \vee \neg C,$			
$C \vee \neg A,$			
$C \vee \neg B,$			
$\neg A \vee \neg B\}$			

Ahne paikallinen haku (GSAT)

Sallitaan sivuttaissiirrot (totuusarvomuutokset, jotka eivät laske epätosien lauseiden määrää.) [Selman/Levesque/Mitchell, 1992]

GSAT(S)
for $i := 1$ to MAX-TRIES **do**
  M := satunnaisesti generoitu malli;
  for $j := 1$ to MAX-FLIPS **do**
    if $M \models S$ **then** return M
    else Valitaan satunnaisesti jonkin atomilause P niistä, joiden totuusarvomuutoksella saadaan suurin lasku epätosien lauseiden määrässä ja muutetaan P:n totuusarvo
    endif
  endfor
endfor

Esimerkki.

$S =$	$\{A \vee B \vee C,$	$\{\neg A, \neg B, \neg C\}(1)$
	$\neg A \vee B \vee C,$	$\mid (A)$
	$A \vee \neg B \vee C,$	$\{A, \neg B, \neg C\}(1)$
	$A \vee B \vee \neg C,$	$\mid (B)$
	$\neg A \vee \neg B \vee C,$	$\{A, B, \neg C\}(1)$
	$A \vee \neg B \vee \neg C,$	$\mid (C)$
	$\neg A \vee B \vee \neg C\}$	$\{A, B, C\}(0)$

- Valittava parametrit MAX-TRIES ja MAX-FLIPS.
- Atomilauseiden satunnainen valinta tekee silmukat epätodennäköisiksi.
- Sivuttaissiirrot oleellisia:
  ilman niitä suorituskyky selvästi huonompi.
- Ahneet siirrot voivat viedä harhaan.
  (Pahimmillaan samaan paikalliseen minimiin useimmilla uudelleen aloituksilla.)

☞ Sallitaan satunnaiset siirrot, joissa epätosien lauseiden määrä saattaa jopa kasvaa.

Simuloitu jäähdytys (simulated annealing)

M := satunnaisesti generoitu malli;
repeat
  if $M \models S$ **then** return M
  else Valitse satunnaisesti atomilause A;
    Laske muutos epätosien lauseiden määrässä (δ),
    kun A:n totuusarvo vaihdetaan;
    if $\delta \leq 0$ (määrä ei kasva) **then**
      muuta mallia M vaihtamalla A:n totuusarvo
    else vaihda A:n totuusarvo todennäköisyydellä $\mathrm{e}^{-\delta/T}$
    endif
  endif
until False

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

T	δ	$\mathrm{e}^{-\delta/T}$
5	1	0.819
0.5	1	0.135
0.5	2	0.0183
0.2	1	0.00674
0.2	2	0.0000454

- Jäähdytysaikataulu:
  — Geometrinen aikataulu: $T_{i+1} = c * T_i$, missä $0 < c < 1$
  — Lämpötila vakio
- Käytetään uudelleen aloitusta.
- Huomaa ero: Ahne paikallinen haku tekee *aina* muutoksen, joka vähentää epätosien lauseiden määrää, jos sellainen on olemassa.

Esimerkki.

$$
\begin{array}{lcll}
S & = & \{A \vee B \vee C, & \{A, B, \neg C\}(1) \\
 & & \neg A \vee B \vee C, & \mid (B) \\
 & & A \vee \neg B \vee C, & \{A, \neg B, \neg C\}(1) \\
 & & A \vee B \vee \neg C, & \mid (A) \\
 & & \neg A \vee \neg B \vee C, & \{\neg A, \neg B, \neg C\}(1) \\
 & & A \vee \neg B \vee \neg C, & \vdots \\
 & & \neg A \vee B \vee \neg C\} &
\end{array}
$$

Satunnainen vaellus (GSAT + walk)

[Selman/Kautz; 1993]

for $i := 1$ to MAX-TRIES **do**
  M := satunnaisesti generoitu malli;
  for $j := 1$ to MAX-FLIPS **do**
    if $M \models S$ **then** return M
    else Todennäköisyydellä p: valitse satunnaisesti atomilause, joka
      esiintyy jossain epätodessa lauseessa ja muuta sen totuusarvo.
      Todennäköisyydellä $1 - p$: muuta minkä tahansa atomilauseen
      totuusarvo, jolla saadaan suurin lasku epätosien lauseiden määrässä.
    endif
  endfor
endfor

- Vaatii parametrin p valinnan.
- Tyypillisesti $0.5 \leq p \leq 0.6$.
- Huomaa tavoitehakuisuus: siirtymiset mahd. huonompaan suuntaan (epätosien lauseiden määrä kasvaa) liittyvät vielä epätosiin lauseisiin.

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

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

Esimerkki. Olkoon $p = 1$ (ei ahneita siirtoja).

$$
\begin{aligned}
S = \{ & A \vee E, \\
& \neg A \vee B, \\
& \neg B \vee C, \\
& \neg C \vee D \vee E\}
\end{aligned}
$$

$$
\begin{array}{c}
\{\neg A, \neg B, \neg C, \neg D, \neg E\}(1) \\
\mid (A) \\
\{A, \neg B, \neg C, \neg D, \neg E\}(1) \\
\mid (B) \\
\{A, B, \neg C, \neg D, \neg E\}(1) \\
\mid (C) \\
\{A, B, C, \neg D, \neg E\}(1) \\
\mid (E) \\
\{A, B, C, \neg D, E\}(0)
\end{array}
$$

Satunnainen kohina (random noise)

for $i := 1$ to MAX-TRIES **do**
  M := satunnaisesti generoitu malli;
  for $j := 1$ to MAX-FLIPS **do**
    if $M \models S$ **then** return M
    else
      Todennäköisyydellä p: valitse satunnaisesti jokin atomilause ja muuta sen totuusarvo.
      Todennäköisyydellä $1 - p$: muuta minkä tahansa atomilauseen totuusarvo, jolla saadaan suurin lasku epätosien lauseiden määrässä.
    endif
  endfor
endfor

Satunnainen vaellus II (WSAT)

for $i := 1$ to MAX-TRIES **do**
  M := satunnaisesti generoitu malli;
  for $j := 1$ to MAX-FLIPS **do**
    if $M \models S$ **then** return M
    else Valitse satunnaisesti epätosi lause;
      Todennäköisyydellä p: valitse satunnaisesti siitä atomilause ja muuta sen totuusarvo.
      Todennäköisyydellä $1 - p$: valitse siitä jokin atomilause, jonka totuusarvo muuttamalla mahdollisimman vähän lauseita muuttuu todesta epätodeksi ja muuta atomilauseen totuusarvo.
    endif
  endfor
endfor

- Vaatii parametrin p valinnan
- WSAT: vaellus + ahne haku
- Huomaa ero verratuna GSAT + walk.
  Vaikka $p = 1$ (ei ahneita siirtoja), algoritmit toimivat eri tavoin: WSAT suosii satunnaisia siirtoja, joissa atomilause esiintyy monessa epätodessa lauseessa.

TABU-haku parametrillä t:
Valitaan epätosi lause ja lauseesta jokin atomilause, jonka totuusarvon muutos johtaa pienimpään epätosien lauseiden määrään.
Jos atomilauseen totuusarvo on muutettu viimeisen t askeleen aikana (tabu-lista), valitaan toinen atomilause. Jos kaikki lauseen atomit ovat tabu-listalla, valitaan toinen epätosi lause. Jos kaikki atomit kaikissa epätosissa lauseissa ovat tabu-listalla, ei huomioida tabu-listaa.

Esimerkki. WSAT, kun $p = 1$ (ei ahneita siirtoja).

$$
\begin{array}{rcll}
S & = & \{A \vee E, & \{\neg A, \neg B, \neg C, \neg D, \neg E\}(2) \\
 & & E \vee B, & (A \vee E / E) \\
 & & \neg B \vee C, & \{\neg A, \neg B, \neg C, \neg D, E\}(1) \\
 & & \neg A \vee D, & (\neg E \vee C \vee D / C) \\
 & & \neg E \vee C \vee D, \} & \{\neg A, \neg B, C, \neg D, E\}(0)
\end{array}
$$

Kokeellisia vertailuja

- Vertailut perustuvat usein satunnaisesti generoituihin esimerkkiluokkiin.
- Parhaat variantit GSAT + walk ja WSAT.
  (WSAT näyttäisi olevan näistä parempi.)
- Kumpikin tehokkaampia kuin esim. simuloitu jäähdytys.
- Vaellus parempi kuin kohina.
- Joskus paikalliset hakumenetelmät ratkaisevat paljon suurempia ongelmia kuin täydelliset menetelmät (esim. Davis-Putnam).

Arviointia

- Paikalliset hakumenetelmät suoraviivaisia toteuttaa.
- Toteutuvuusongelmaan suunnitelluilla menetelmillä parempi suorituskyky kuin yleisemmillä.
- Hyvät parametrien arvot oleellisia suorituskyvyn kannalta.
  Esim. WSAT:ssa satunnaisen muutoksen todennäköisyys p tai SA:ssa jäähdytysaikataulu (*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.