

No free lunch theorems for optimization

(David H. Wolpert and William G. Macready)

“Roughly speaking, we show that for both static and time-dependent optimization problems, the average performance of any pair of algorithms across all possible problems is identical.”

“This is true even if one of the algorithms is random; any algorithm a_1 performs worse than randomly just as readily (over the set of all optimization problems) as it performs better than randomly.”

- Search space \mathcal{X} (finite)
- Space of possible cost values \mathcal{Y} (finite)
- Cost function $f : \mathcal{X} \rightarrow \mathcal{Y}$
- Space of all possible problems $\mathcal{F} = \mathcal{Y}^{\mathcal{X}}$ ($|\mathcal{F}| = |\mathcal{Y}|^{|\mathcal{X}|}$)
- Distribution $P(f) = P(f(x_1), \dots, f(x_{|\mathcal{X}|}))$

“a prior specification of the class of the optimization problem at hand”

- Uniform *a priori* distribution $P(f) = 1/|\mathcal{F}|$

NEIGHBORHOOD STRUCTURE NOT TAKEN INTO ACCOUNT IN ANY WAY!

“it is certainly true that any class of problems faced by a practitioner will not have a flat prior [distribution]”

- “comparing algorithms based on the number of *distinct* function evaluations they have performed”
- Oracle tells the values of the function
- “Sample” (time-ordered set of m distinct visited points)
 $d_m \equiv \{(d_m^x(1), d_m^y(1)), \dots, (d_m^x(m), d_m^y(m))\}$
- Black-box optimization algorithm $a : d \in D \mapsto \{x | x \notin d^x\}$
(“all of our results are extensible to nondeterministic [stochastic] algorithms”)
- “any measure of the performance of an algorithm after m iterations is a function of the sample d_m^y
e.g. $\min\{d_m^y(i)\}$ ”

- No free lunch theorem 1:
For any pair of algorithms a_1 and a_2

$$\forall m \in \mathbb{N} : \sum_f P(d_m^y | f, m, a_1) = \sum_f P(d_m^y | f, m, a_2) \quad (1)$$

- With a nonuniform distribution $P(f)$:
 $\sum_f P(d_m^y | f, m, a) P(f)$ should be used instead
- NFL-theorems valid only with special $P(f)$ -distributions!
(“NFL can also be demonstrated for a range of nonuniform priors.”)
- Analogous NFL theorem for a class of time-dependent cost functions
- “the NFL theorems mean that if an algorithm does particularly well on average for one class of problems then it must do worse on average over the remaining problems.”
- “if the practitioner has knowledge of problem characteristics but does not incorporate them into the optimization algorithm, then $P(f)$ is effectively uniform.”

- Lessons from NFL theorems:

“indicate the importance of incorporating problem-specific knowledge into the behavior of the algorithm.”

e.g. local search methods are based on the existence of correlation between neighboring configurations

“demonstrate the danger of comparing algorithms by their performance on a small sample of problems”

- Results of the random algorithm as benchmark measures
- a_1 has better “head-to-head” minimax behavior than a_2 if there are f for which a_1 beats a_2 badly, but none for which a_1 does substantially worse than a_2
- $P(f)$ -independent results
 - one f with all the algorithms (uniform $P(a)$ -distribution)
 - slow vs. fast algorithms on “real” problems

Miksi NFL-teoreemat eivät juurikaan vaikuta kombinatorisen optimoinnin "arkipäivään"?

- Kustannusfunktion f ottaminen satunnaisesti "kaikkien kustannusfunktioiden joukosta" tarkoittaa, että tarkastellaan oikeastaan vain täysin satunnaisten funktioiden luokkaa, jolloin

$$\forall \epsilon > 0 : \lim_{N \rightarrow \infty} P\{r(f_N) > \epsilon\} = 0$$

jossa $r(f_N)$ on yhden askeleen autokorrelaatio eli naapurikonfiguraatioiden välinen korrelaatio maastossa f_N .

- Jos f otetaan jostain tällä kurssilla käsitellyistä ongelmaluokista (MAXSAT, TSP, spinlasi, NK ...) jollain tavallisesti käytettävällä jakaumalla, pätee yleensä

$$\forall \epsilon > 0 : \lim_{N \rightarrow \infty} P\{r(f_N) > 1 - \epsilon\} = 1$$

- Naapurikonfiguraatioiden korreloituneisuus on yleinen ominaisuus!

Satunnaiset osafunktiomaastot (SOFM)

Alkuperä: embedded landscapes (Heckendorn et al. 1998)

Sakari Seitz

- Konfiguraatioavaruus $X = \mathbb{B}^N$
- Kelpoisuusfunktio

$$f(x) = \frac{1}{M} \sum_{j=1}^M g_j(x_{I(j,1)}, x_{I(j,2)}, \dots, x_{I(j,k)})$$

- $M = \lceil aN \rceil$ ($a > 0$)
- Indeksitaulu I kertoo, mitkä x :n positiot vaikuttavat mi-hinkin osafunktioon. I :n alkioiksi arvotaan täysin satunnaisesti indeksejä väliltä $1..N$. (I :hin tulevia indeksejä ei lajitella suuruusjärjestykseen.)
- Osafunktioiden g_j arvon ei tarvitse riippua x :n kaikista k :sta mukana olevasta positiosta (osafunktion arvo voi olla vaikka vakio).
- Parametri k kertoo siis interaktioiden suurimman kertaluvun (ja samalla f :n Walsh-hajotelmaan kuuluvien funktioiden suurimman mahdollisen kertaluvun).
- Osafunktiot g_j otetaan jostakin todennäköisyysjakaumasta \mathcal{G} toisistaan riippumattomasti. \mathcal{G} :hen täytyy kuulua aina myös sellaisia funktioita, joiden arvo ei ole vakio.

Satunnaisotokseen perustuva optimointialgoritmi

Arvotaan N^b satunnaista konfiguraatiota SOF-maastosta f ja valitaan niistä paras.

Kun $N \rightarrow \infty$, algoritmin tulos lähenee maaston keskiarvoa μ .

Perustelu:

- Olkoon μ ja σ osafunktioiden g arvojen odotusarvo ja keskihajonta \mathcal{G} -jakaumassa.
- Suurilla N :illä f :n arvot ovat suunnilleen normaalijakautuneita μ :n läheisyydessä.

- $z_\epsilon = \mu + \epsilon\sigma = \mu + \epsilon_N\sigma_N$ ($\epsilon_N = \epsilon\sqrt{M}$, $\sigma_N = \sigma/\sqrt{M}$)

- $P\{Z_0 > z_0\}$ on todennäköisyys normaalijakaumassa $\Phi(0, 1)$

-

$$\lim_{N \rightarrow \infty} \frac{\ln P\{Z_0 > \epsilon\sqrt{M}\}}{M} = -\frac{\epsilon^2}{2}$$

\Rightarrow suurilla N :

$$P\{Z_0 > \epsilon_N\} \approx e^{-\epsilon^2 M/2} \approx e^{-\epsilon^2 a N/2}$$

\Rightarrow

$$\lim_{N \rightarrow \infty} N^b P\{Z_0 > \epsilon_N\} = 0 \quad \forall b > 0$$

$N^b P\{Z_0 > \epsilon_N\}$ on (approks.) odotusarvo sille, kuinka moni N^b -kokoisen satunnaisotoksen arvoista on suurempia kuin $\mu + \epsilon\sigma$ pienillä ϵ :illa.

Record-to-Record Travel -algoritmi (Gunter Dueck 1993)

Muistuttaa hill climbing -algoritmia, jossa kokeillaan bitin flip-pausta satunnaisesti valituissa positioissa ja hyväksytään ne flip-paukset, jotka eivät vähennä kelpoisuutta. RRT-algoritmissa hyväksytään myös huonontavia askeleita, jos ne toteuttavat ehdon:

$$f(x) \geq f(x^*) - \delta$$

jossa $f(x^*)$ on suurin siihen mennessä löytynyt arvo (ennätys) ja δ on kynnyks, joka määrittää "neutraalin" alueen leveyden.

Hypoteesi SOF-maastoille

Olkoon $R_{\mathcal{G}}(N, i, \delta)$ parhaan löydetyn konfiguraation kelpoisuuden odotusarvo silloin kun N -kokoiselle ongelmainstanssille on RRT-algoritmillä tehty iN flippausyritystä kynnyksarvon ollessa δ/M .

Kullakin \mathcal{G} :lla (jos \mathcal{G} :n funktioiden arvojen keskiarvo ja varianssi ovat äärellisiä ja varianssi nollaa suurempi) pätee:

$$\forall i \geq 0 : \quad \lim_{N \rightarrow \infty} R_{\mathcal{G}}(N, i, \delta) = h_{\mathcal{G}}(i, \delta), \quad h_{\mathcal{G}}(i+1, \delta) > h_{\mathcal{G}}(i, \delta)$$

($i = 0$ vastaa satunnaista aloituskonfiguraatiota.)

SOFM (jatkoa)

- Erikoistapaus (symmetrinen):

$$\forall g \in \mathcal{G}, x \in \mathbb{B}^N : \quad g(x) = g([11 \dots 1] - x)$$

Varsinkin spinmallilla, jossa on vain ferromagneettisia kytkentöjä, on erikoisia ominaisuuksia, ks. Svenson 2001.

- Yleistys:
Kullekin positiolle arvotaan toisistaan riippumattomasti painokerroin (≥ 0) jostakin todennöisyysjakaumasta \mathcal{W} . Indeksitaulun I arvonnassa käytetään sitten näitä kertoimia positioiden valintatodennäköisyyden painottamiseen.

Sekalaista

- Olisi mukavaa pystyä osoittamaan myös, että

$$\lim_{N \rightarrow \infty} R_{\mathcal{G},b}(N, i, \delta) = h_{\mathcal{G}}(i, \delta)$$

jossa $R_{\mathcal{G},b}(N, i, \delta)$ on parhaan löytyneen kelpoisuuden odotusarvo, kun RRT-algoritmia toistetaan $\lceil N^b \rceil$ kertaa (satunnaisista konfiguraatioista lähtien).

Eli polynominen määrä toistoja ei paranna raja-arvoa, koska suurilla N :illä kaikki toistot tuottavat lähes saman arvon.

- Kaikilla a :n arvoilla ($M = \lceil aN \rceil$) saadaan yhtä suuri odotusarvo autokorrelaatiolle (\mathcal{G} :n ja N :n funktiona).