

Összefoglalás

Az Optimalitáselmélet (*Optimality Theory*, OT Prince and Smolensky, 1993) az elmúlt évtized egyik legnépszerűbb elmélete, elsősorban a fonológiában (a hangtanban), de más nyelvészeti területeken is. A jelen disszertáció célja a modell számítógépes implementációja, vagyis olyan algoritmus kidolgozása, amely megtalálja az OT jelöltek halmazán az optimális jelöltet, amely az elmélet szerint az adott nyelv grammatikus alakjának felel meg.

Az optimális jelölt megkeresésére a *szimulált hőkezelés* (*szimulált lehűtés*, *simulated annealing*) nevű algoritmust használtam, amely a statisztikus fizikából a számítástudományba átvett, elterjedt heurisztikus optimalizálási algoritmus. Az *SA-OT Algoritmust* (*Simulated Annealing for Optimality Theory Algorithm*) a 2.8. ábra mutatja be, a 64. oldalon. Mint a legtöbb heurisztikus algoritmus, az SA-OT sem garantálja azt, hogy mindig megtaláljuk az alaphalmaz legjobb elemét, pontossága (a keresett elem megtalálásának a valószínűsége) általában kisebb 100%-nál. Mégis azt állítom, hogy az emberi beszédprodukciónak adekvát modellje, mivel (1) egyszerű, (2) tetszőleges időtartam alatt produkál egy outputot, és (3) ez az időtartam lerövidíthető, az algoritmus felgyorsítható a pontosság rovására. A beszédpartnerünknek nem kell várnia egy komplexebb mentális számítás esetén sem, és ha gyorsan kell beszélnünk, legfeljebb bevállaljuk a hibázás nagyobb esélyét. Ezért az SA-OT Algoritmus a performanciahibák modellezésére is alkalmas.

A nyelvi kompetencia és a nyelvészetileg motivált performancijelenségek egységes kezelésére tett javaslatomat mutatja be a 2.1 táblázat a 43. oldalon. Ha különválasztjuk a nyelvészeti modellt, például egy OT-nyelvtant, annak az implementációjától, akkor az előbbi a nyelvi kompetenciát (a nyelv statikus ismeretét az agyban), utóbbi pedig a performanciát (a dinamikus nyelvprodukciónak) adhatja vissza. A nyelvten jóslatot tesz arra nézve, hogy mely alakokat tartja az anyanyelvi beszélő grammatikusnak, míg a nyelvten implementációja kvantitatív előrejelzéseket tehet a grammatikus, ill. a kevésbé grammatikus alakok előbukkanási gyakoriságára. Az SA-OT Algoritmus, paramétereinek finombeállítására révén, éppen ezen valószínűségek reprodukciójára alkalmas.

Az SA-OT egy *topológiát* (egy *szomszédsági struktúrát*) igényel az OT jelölt-halmazon. Így *lokális optimumokról* is beszélhetünk, vagyis olyan jelöltekről, amelyek harmonikusabbak a szomszédjaiknál, függetlenül attól, hogy az egész halmaznak *globális optimumjai*-e. Az SA-OT outputjai épp ezek a lokális optimumok lesznek, vagyis azok a jelöltek felelnek meg a performanciahibáknak, amelyek globálisan nem optimálisak, de amelyeket, lokális optimumok lévén, az algoritmus kiadhat.

Hogyan történik ez? Az SA-OT Algoritmus egy véletlen bolyongást valósít

meg a jelöltek halmazán. Ha a véletlen bolyongó épp a w pozícióban található, a szomszéd jelöltek közül kiválaszt egyet, w' -t, $P_{choice}(w'|w)$ a priori valószínűséggel. Ezek a valószínűségek állandóak, függetlenek a jelöltek harmóniájától, a constraint-ek rendezésétől, és a topológia definíciójának a részei. Majd összehasonlítja w -t és w' -t, és egy másik $P(w \rightarrow w'|T)$ valószínűséggel átlép w' -be. Utóbbi valószínűségek függenek a jelöltek constraint-sértéseitől, és változnak az algoritmus T paramétere (a „hőmérséklet”) függvényében. Ha a w' jelölt nem rosszabb, mint w , akkor ez a valószínűség mindig 1 (harmonikusabb jelöltre mindig át szabad lépni), ellenkező esetben pedig a szimuláció elején 1, majd fokozatosan lecsökken 0-ra. Ezért a véletlen bolyongó a szimuláció elején még ki tud szabadulni a lokális optimumokból, míg a végén belejük ragad, és ez a végállapot válik az algoritmus outputjává. Ha a T „hőmérsékletet” lassabban csökkentjük, vagyis több lépést, több iterációt engedünk meg, akkor sok esetben (de nem mindig, lásd a 6. fejezetet) megnő annak a valószínűsége, hogy a véletlen bolyongó az algoritmus végére megtalálja a globális optimumot, amelyet a hagyományos Optimalitáselmélet a grammatikus alaknak feleltet meg.

Az 1. fejezet bemutatja az Optimalitáselméletet és néhány változatát, valamint a sztochasztikus módszerek nyelvészeti relevanciáját veti fel. A 2. fejezet a heurisztikus módszerek mellett érvel, majd bevezeti az SA-OT Algoritmust. A 3. fejezet az Optimalitáselmélet matematikai megalapozását nyújtja, annak érdekében, hogy az SA-OT Algoritmust formális eszközökkel is bevezethesse. Bemutatja azt, hogy miképp lehet a jelöltek harmóniáját polinomokkal, valamint (transzfinit) rendszámokkal ábrázolni. A 4. fejezet spekulációi az SA-OT és a lexikon, illetve a tanulhatóság viszonyát feszegetik.

Az 5. fejezettől kezdve konkrét nyelvészeti példákon teszteljük az SA-OT-t. Az 5. fejezet a holland hangsúlyok eltolódását szimulálja gyorsbeszédben. A lassan lefuttatott szimuláció a két lehetséges alak normális tempójú beszédben előforduló gyakoriságait, míg a gyorsan lefuttatott szimuláció a gyorsbeszédbeli gyakoriságait hivatott visszaadni. A 6. fejezet a regresszív és progresszív zöngésségi harmóniát, míg a 7. fejezet a magyar névelő tapadását és a Prince és Smolensky-féle szótagolás-modellt tárgyalja.

Ezekben a fejezetekben a sok ponton támadható fonológiai modelleknél fontosabb az, ahogyan az SA-OT lehetőségeit fokozatosan kiaknázzuk. Az említett jelenségek ürügyén az algoritmus paramétereit és a különféle topológiákat teszteljük, különböző trükköket alkalmazunk. A rendszer viselkedését, a kísérletek mellett, a 6. fejezetben analitikus eszközökkel is igyekszünk megérteni. A 6. és a 7. fejezetben egy topológia-típust fokozatosan teszünk egyre összetettebbé. A 6. fejezetben arra is példát látunk, hogy egy nyelvtant meg lehet őrizni egyszerűnek, ha a „szabálytalan” alakokat a performanciamodell területére száműzzük: a vizsgált példában az algoritmus mindig elő fogja állítani a globálisan nem optimális lokális minimumot, vagyis azt jósoljuk, hogy a nyelvtannak ellentmondó alak minden körülmények közt elő fog fordulni. Érveket láttunk amellett is, hogy a felszínen soha meg nem jelenő alakokat is bevegünk a jelöltek halmazába, és az SA-OT olyan constraintek rendezésében is segít, amelyeket a hagyományos OT nem tudna rendezni.

Az 5-7. fejezet tanulságait részletesebben a 8.1 alfejezet foglalja össze. A 8.2 alfejezet az Optimalitáselmélet korábban már tárgyalt változataival veti össze az SA-OT Algoritmust. Végezetül pedig az SA-OT-t visszahelyezzük abba a környezetbe, amelyben az Optimalitáselmélet eredetileg született, a kognitív tudományok közé.