



Využití umělé inteligence při vytváření testovacích funkcí pro evoluční algoritmy

Autor: Ing. Petr Vykoukal

Vedoucí práce: doc. Ing. Michal Pluháček, Ph.D.

Univerzita Tomáše Bati ve Zlíně – Fakulta aplikované informatiky

Motivace

Za účelem srovnání výkonnosti evolučních algoritmů jsou používány testovací funkce, případně specifické sady testovacích funkcí, které reprezentují optimalizační problém. Mezi nejznámější sady patří IEEE CEC a BBOB, které jsou vytvářeny pro srovnávací soutěže. Pro obměnu testovacích funkcí v těchto sadách jsou existující funkce posunuty a otočeny, případně kombinovány. Podle No Free Lunch Theorem platí, že neexistuje žádný univerzální algoritmus, který by dosahoval nejlepších výsledků při řešení všech optimalizačních problémů [1]. Pokud jsou definovány specifické vlastnosti testovacích funkcí, mohou být na základě těchto vlastností umístěny do prostoru problémů. Bylo zjištěno, že používané sady testovacích funkcí vytváří v prostoru problémů shluky na základě jejich podobných vlastností [2][3]. Práce zkoumá možnosti využití techniky umělé inteligence k tvorbě nových testovacích funkcí, které by svými měřitelnými vlastnostmi lépe pokrývaly prostor problémů.

Referenční data

Pro rozdělení testovacích funkcí do prostoru problémů byla použita průzkumná analýza krajiny **ELA**. Vybráno bylo 27 různých vlastností s minimální korelací. Pro získání počátečního pokrytí prostoru problémů byly analyzovány sady testovacích funkcí IEEE CEC 2017, IEEE CEC 2022 a dalších 20 testovacích funkcí, používaných v testovacích sadách, ale bez jejich rotace a posunutí, nebo funkce používané autory studií pro provádění srovnávacích testů. Analýza každé testovací funkce byla provedena 30x a testovací funkce byly analyzovány s 10ti vstupními parametry. Na získaných datech byla provedena analýza hierarchického shlukování a data byla zobrazena redukcí dimenzionality z 27D do 2D algoritmem t-SNE.

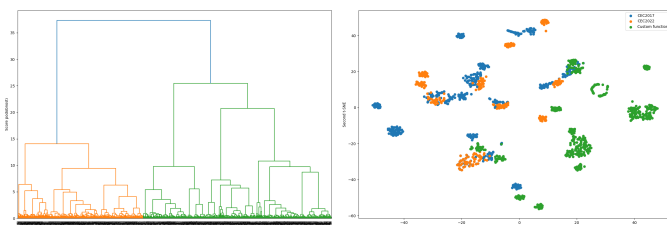


Figure 1. Hierarchické shlukování (vlevo) a t-SNE (vpravo)

Postup generování nových funkcí

Původní testovací funkce byly analyzovány pro zjištění základních stavebních kamenů, vhodných pro syntetizování nových funkcí. Na základě této analýzy byla sestavena sada **General Function Set**.

GFS_2	$m + n, m - n, m \cdot n, \frac{m}{n}$
GFS_1	$\frac{1}{n}, n , n^2, \sin(n), \sin^2(n), \cos(n), \sqrt{ n }$
GFS_0	$x_i, x_{i+1}, \sqrt{x_i^2 + x_{i+1}^2}, \pi, e, K_{\mathbb{N}}, K_{\mathbb{R}}, \omega$

Table 1. Použitý set GFS

Symbole $K_{\mathbb{N}}$ a $K_{\mathbb{R}}$ reprezentují celočíselné a desetinné konstanty a symbol ω reprezentuje využití posílené evoluce.

Pro generování nových testovacích funkcí byl použit algoritmus **Analytické programování** s funkcí posílené evoluce [4], ve spojení s evolučním algoritmem **DISH** [5], založeným na diferenciální evoluci.

Účelová funkce

V rámci jednotlivých experimentů byly sestaveny dvě účelové funkce pro vyhodnocení vhodnosti nově generovaných testovacích funkcí. Účelem optimalizace tak byla maximalizace mocniny Euklidovské vzdálenosti škálovaných **ELA** vlastností a centroidů získaných z původních funkcí. Centroidy byly vypočítány jako

- střed shluků získaných algoritmem k-means
- střed shluků 30ti měření každé funkce

Ukázka generované funkce

$$f(\vec{x}) = \sum_{i=1}^{D-1} \left(\left(\frac{1}{\cos\left(\frac{\pi}{\sin(\omega)}\right)} \right) \cdot \cos(0.35270892) \right)$$

$$\text{kde } \omega = \left(\pi - \cos\left(\sin^2\left(\sqrt{x_i^2 + x_{i+1}^2}\right)\right) \right)$$

Závěr

Byla vytvořena sada 70ti testovacích funkcí, u kterých analýza pomocí t-SNE ukázala, že pokrývají rozsáhlejší prostor problémů, než původní sady testovacích funkcí. Nově generované funkce nevytváří shluky a existujícím shlukům se ve většině případů vyhýbají.

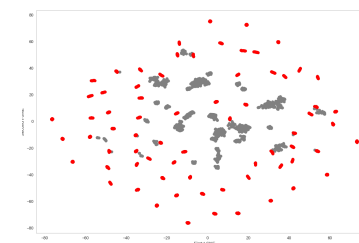


Figure 2. Zobrazení prostoru problémů algoritmem t-SNE (červeně zvýrazněné generované funkce)

Výsledné funkce byly optimalizovány algoritmy **DISH**, **CLPSO** a **Db1_SHADE**. Výsledky byly statisticky vyhodnoceny Friedmanovým testem, kde bylo zjištěno, že je statisticky významný rozdíl v použití jednotlivých algoritmů na testovací sadě.

Možným směrem budoucího výzkumu by bylo zkoumání, zda by vytvořením větší testovací sady bylo možné analyzovat prostor problémů z hlediska výkonnosti různých evolučních algoritmů. Tedy, zda je nějaký evoluční algoritmus úspěšnější, pokud se testovací funkce nachází v konkrétní oblasti vlastností **ELA**. To by mohlo vést k trénování klasifikátoru, který by dokázal doporučit vhodný evoluční algoritmus k řešení nové optimalizační úlohy na základě zjištění jejich **ELA** vlastností.

Zdroje

- [1] WOLPERT D.H. a MACREARY W.G. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.
- [2] GARDEN Robert W. a ENGELBRECHT Andries P. Analysis and classification of optimisation benchmark functions and benchmark suites. online. *2014 IEEE Congress on Evolutionary Computation (CEC)*, pages 1641–1649, 2014.
- [3] LANG Ryan Dieter a ENGELBRECHT Andries Petrus. An exploratory landscape analysis-based benchmark suite. *Algorithms*, 14(3), 2021.
- [4] ZELINKA Ivan KOMÍNKOVÁ OPLATKOVÁ Zuzana a NOLLE Lars. Analytic programming symbolic regression by means of arbitrary evolutionary algorithm. *International Journal of Simulation Systems, Science & Technology*, 6(9):44–56, 2005.
- [5] VIKTORIN Adam SENKERIK Roman PLUHACEK Michal KADAVY Tomas a ZAMUDA Ales. Distance based parameter adaptation for success-history based differential evolution. *Swarm and Evolutionary Computation*, 50, 2019.