



Universiteit  
Leiden  
The Netherlands

## **Benchmarking discrete optimization heuristics: from building a sound experimental environment to algorithm configuration**

Ye, F.

### **Citation**

Ye, F. (2022, June 1). *Benchmarking discrete optimization heuristics: from building a sound experimental environment to algorithm configuration*. Retrieved from <https://hdl.handle.net/1887/3304813>

Version: Publisher's Version

License: [Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden](#)

Downloaded from: <https://hdl.handle.net/1887/3304813>

**Note:** To cite this publication please use the final published version (if applicable).

# Samenvatting

In zowel het dagelijks leven als in wetenschappelijk onderzoek worden we vaak geconfronteerd met moeilijke optimalisatie-problemen. Om deze problemen op te lossen zijn er veel verschillende algoritmes ontworpen, maar we hebben meestal geen duidelijke conclusies over hoe effectief deze algoritmes zijn voor specifieke type problemen. Gelukkig kunnen benchmark studies ons helpen om de effectiviteit van deze algoritmes op een onbevooroordeelde manier te vergelijken.

In deze thesis introduceren we de IOHprofiler, een benchmark omgeving die de transitie van de implementatie van algoritmes naar het analyseren en visualiseren van hun effectiviteit mogelijk maakt. Deze software bestaat uit twee componenten: IOHexperimenter, een gebruiksvriendelijke en makkelijk aanpasbare module voor het uitvoeren van de benchmark experimenten en het genereren van de bijbehorende data; en IOHanalyzer, een module voor het verwerken van deze data en het genereren van gedetailleerde statistische analyses.

We maken gebruik van de verscheidene functionaliteiten van IOHprofiler om de effectiviteit van evolutionaire algoritmes voor het optimaliseren van discrete problemen te analyseren. We bestuderen specifiek de impact van de mutatie-graaden de grootte van de populatie in  $(1 + \lambda)$  EAs en de impact van de crossover kans binnen  $(\mu + \lambda)$  EAs op OneMax en LeadingOnes. Verder vergelijken we twaalf heuristieken en verschillende versies van genetische algoritmes op vijftientig pseudo-Booleaanse problemen. Geïnspireerd door deze resultaten introduceren we de gestandaardiseerde genormaliseerde bit mutatie voor EAs en stellen we voor om niet-asymptotische looptijd-analyse (d.w.z., genen die houden voor een specifieke dimensie van een probleem in plaats van grote-O-notatie) te gebruiken voor theoretische studies over het gedrag van EAs.

Vervolgens gebruiken we verschillende algoritme-configuratie methoden (irace, MIP-EGO en MIES) om de parameters van een genetisch algoritme af te stemmen. De experimentele resultaten geven ons inzicht in veelbelovende configuraties van het

## Samenvatting

---

genetische algoritme voor verschillende type problemen. Ook analyseren we de impact van de kost-metriek op het configuratie proces. Onze resultaten suggereren dat zelfs wanneer het doel is om de verwachte optimalisatie-tijd te minimaliseren het vaak wenselijk is om andere metrieken, die gebruik maken van de effectiviteit op meerdere punten in de optimalisatie (bijvoorbeeld oppervlakte onder de curve) te gebruiken tijdens het configureren.

Tot slot maken we gebruik van de verzamelde benchmark-data voor dynamische selectie van optimalisatie-algoritmes, waarvan de resultaten laten zien dat het wisselen van een configuration van het genetisch algoritme naar een andere configuratie tijdens het optimalisatie proces een verbeterde effectiviteit geeft vergeleken met de statische configuraties.