



Universiteit  
Leiden  
The Netherlands

## Self-adjusting surrogate-assisted optimization techniques for expensive constrained black box problems

Bagheri, S.

### Citation

Bagheri, S. (2020, April 8). *Self-adjusting surrogate-assisted optimization techniques for expensive constrained black box problems*. Retrieved from <https://hdl.handle.net/1887/87271>

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/1887/87271>

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

Cover Page



Universiteit Leiden



The handle <http://hdl.handle.net/1887/87271> holds various files of this Leiden University dissertation.

**Author:** Bagheri, S.

**Title:** Self-adjusting surrogate-assisted optimization techniques for expensive constrained black box problems

**Issue Date:** 2020-04-08

---

# Samenvatting

Realistische optimalisatieproblemen uit de echte wereld komen overal voor. Van dagelijkse taken tot complexe wetenschappelijke, industriële of zakelijke problemen. Op veel technische gebieden heeft de ontwikkeling van nauwkeurige simulatiesoftware, die echte experimenten vervangt, het mogelijk gemaakt om een optimaal ontwerp te vinden voor zeer complexe systemen met veel parameters. Zoeken naar een optimaal ontwerp is vaak onderworpen aan meerdere restricties zoals geometrische beperkingen, stabiliteit, veiligheid, budget, enz., waardoor alleen een kleine subset van ontwerpen uitvoerbaar is.

Optimalisatie van industriële ontwerpen, ookwel simulatie-gebaseerde optimalisatie genoemd, verwijst naar een klasse problemen waarbij de doel- en / of beperkingfuncties "black boxes" zijn. Dit betekent dat ze geen expliciete algebraïsche formulering hebben.

Het feit dat de doel en beperkingfunctie van dergelijke problemen alleen geëvalueerd kunnen worden door middel van tijdrovende simulaties is waarom de efficiëntie van een optimalisatie-algoritme van cruciaal belang is op dit gebied. Daarom kunnen we stellen dat veel echte optimalisatieproblemen "black box" zijn, duur zijn om te evalueren, en onderhevig zijn aan meerdere beperkingen en hoge dimensionaliteit.

De belangrijkste focus van dit proefschrift is het ontwikkelen van geschikte optimalisatie algoritmen om de genoemde uitdagingen op een efficiënte manier aan te pakken. De pogingen om deze optimalisatieproblemen op te lossen maken vaak gebruik van de techniek waarbij het hoofdprobleem opgesplitst wordt in deelproblemen. In **hoofdstuk 2** geven we een kort overzicht van de bestaande optimalisatiemethoden die geschikt zijn voor het aanpakken van black box problemen, zowel met als zonder restricties. Slechts een aantal studies zijn gewijd aan het gelijktijdig aanpakken van alle veeleisende uitdagingen van realistische optimalisatieproblemen. In dit proefschrift introduceren we twee methoden om dure "black box" optimalisatieproblemen met beperkingen op te lossen (**C**onstrained **O**ptimization **P**roblem, i.e., COP's).

---

De geïntroduceerde methoden worden geëvalueerd aan de hand van twee COP's en een set van 24 goed bestudeerde testproblemen die bekend staan als de Gproblems-suite [107].

MOPTA08 [92] is een grootschalig massa-optimalisatieprobleem in de auto-industrie. Dit probleem, geïntroduceerd door een technicus bij General Motors, heeft een 124-dimensionale parameter ruimte met 68 niet-lineaire beperkingen die de uitvoer zijn van een crashsimulatie. In het ideale geval kunnen de beperkingfuncties in de praktijk 60 keer per dag worden berekend. Het is gewenst om binnen een maand een ontwerp te vinden met 10% tot 20% verminderde massa, wat betekent dat het maximale aantal functie-evaluaties beperkt is tot 60 keer 30 = 1800. De tweede COP die in dit proefschrift is gebruikt is een probleem gericht op het minimaliseren van de aerodynamische weerstandskracht van het ontwerp van een vleugelprofiel, die onderworpen is aan meerdere gelijkheids- en ongelijkheidsbeperkingen, zoals beschreven in **hoofdstuk 5**.

SACOBRA (**S**elf-**A**djusting **C**onstrained **O**ptimization by **R**adial Basis Function Interpolation), wat staat voor zelfregulerende optimalisatie met restricties door radiale basis functie-interpolatie, is een efficiënte techniek die RBF-interpolaties gebruikt als surrogaat modellen voor doel- en beperkingfuncties. SACOBRA wordt geïntroduceerd in **hoofdstuk 3** van dit proefschrift. Dit optimalisatieraamwerk kan sommige van zijn belangrijkste hyperparameters automatisch beheren zonder enige voorafgaande informatie over de problemen. SACOBRA kan ongeveer 80% van de G-problemen met succes oplossen met een vaste configuratie en zonder verdere parametrisering. Dit gezegd zijnde, overtreft SACOBRA andere state-of-the-art algoritmen bij het optimaliseren van MOPTA08 en G-problemen. In **hoofdstuk 4** is het SACOBRA-raamwerk uitgebreid om met gelijkheidsbeperkingen en ongelijkheidsbeperkingen om te gaan.

Kubieke radiale basisfunctie-interpolatie heeft sterke prestaties getoond als surrogaat in het SACOBRA-optimalisatieraamwerk. Er zijn echter geen theoretische of praktische aanwijzingen dat kubieke RBF's de beste keuze zijn. We hebben een online modelselectieprocedure voor SACOBRA ontwikkeld om automatisch het beste type radiale basisfunctie te kiezen tijdens de optimalisatieprocedure in **hoofdstuk 8**. Deze aanpak verhoogt de prestaties van SACOBRA met 10% op de hele set van 24 G-problemen. Het online modelselectiemechanisme kan worden toegepast op elk ander sequentiële, surrogaat ondersteunde, optimalisatie algoritme.

Een tweede optimalisatiebenadering, de zogenaamde SOCU, wordt geïntroduceerd in **hoofdstuk 5**. De naam SOCU (**S**urrogate-Assisted **O**ptimization encompassing **C**onstraints and **U**ncertainties) staat voor surrogaat ondersteunde optimalisatie met ondersteuning voor beperkingen en onzekerheden. SOCU, die een

---

probabilistische modelleringstechniek (Kriging ofwel Gaussian Processes) gebruikt als surrogaten, is een uitbreiding op het efficiënte globale optimalisatie-algoritme [90] (EGO), voor het omgaan met beperkingen in optimalisatieproblemen.

Voor zover wij weten, is het de eerste keer dat een op EGO gebaseerd algoritme wordt geëvalueerd op de uitdagende G-probleem-COP's. SOCU presteert beter dan de op Kriging gebaseerde methoden van Schonlau [159] vanwege zijn plugin-regelvermogen. Het kon echter alleen met SACOBRA concurreren op de laag-dimensionale problemen en in de vroege iteraties.

RBF-interpolatie gebruikt in SACOBRA, en Kriging gebruikt in SOCU, zijn beide algemeen gebruikte keuzes van modelleringstechnieken voor surrogaat ondersteunde optimalisatiemethoden. Hoewel beide methoden van zeer verschillende oorsprong zijn, hebben ze niet te ontkennen gelijkenissen. Enkele overeenkomsten en verschillen tussen RBF en Kriging worden vermeld in **hoofdstuk 7** Kriging geeft in tegenstelling tot de RBF-interpolatie een onzekerheidsmaat, die aangeeft hoe onzeker het model op elk punt is. Deze eigenschap maakt Kriging tot een populaire keuze voor veel optimalisatie algoritmen, vooral efficiënte methoden (zonder ondersteuning voor beperkingen) zoals SOCU die probabilistische concepten gebruikt.

Hoewel de RBF-interpolatie van nature geen onzekerheidsmaat biedt, hebben we in **hoofdstuk 7** een onzekerheidsmaat voor elke willekeurige RBF-kernel geformuleerd door middel van een analogie met Kriging.

Het bieden van een redelijk goed surrogaat model voor functies met hoge conditionering is een grote uitdaging. Dergelijke functies hebben een hoge verhouding van de steilste helling in de ene richting tot de vlakste helling in een andere richting. In **hoofdstuk 9** proberen we die uitdagingen aan te pakken waar SACOBRA mee wordt geconfronteert bij het oplossen van optimalisatieproblemen met doelfuncties met hoge conditionering.

We ontwikkelen een online whitening-aanpak (OW) voor SACOBRA, waarmee we functies met hoge conditionering transformeren naar eenvoudiger te modelleren functies. SACOBRA met de online whitening-aanpak (OW) is in staat oplossingen te vinden met aanzienlijk betere prestaties. Hoewel een groot aantal functie-evaluaties nodig zijn voor het voorgestelde OW-mechanisme, zijn de meeste van deze evaluaties parallel te doen. SACOBRA + OW overtreft SACOBRA bij het aanpakken van de BBOB-problemen zonder ruis [58] met hoge conditionering.