A MATLAB Toolbox for Surrogate-Assisted Multi-Objective Optimization: A Preliminary Study

Abdullah Al-Dujaili, S. Suresh

Nanyang Technological University

aldujail001@e.ntu.edu.sg

July 18, 2016
Multi-objective Optimization Problems (MOPs) involve a set of conflicting objectives that are to be optimized simultaneously.

It is common that derivatives of the objectives $f$ are neither symbolically nor numerically available.

Evaluating $f$ is typically expensive requiring some computational resources (e.g., a computer code or a laboratory experiment).

Solve using a finite budget of function evaluations.
Motivation

**Surrogate modeling**: a powerful ingredient for **computationally-expensive** Single-objective Optimization Problems (SOPs) (Jones et al., JOPT, 1998).

Readily **available well-benchmarked** software libraries for surrogate-assisted SOPs (Mueller, arXiv, 2014).

MOPs: growing community efforts towards consolidating—e.g., the recent SAMCO workshop¹. benchmarking surrogate-assisted algorithms (on different problems independently (Akhtar & Shoemaker, JOPT, 2015)).

¹http://samco.gforge.inria.fr/doku.php
Objective

Add a brick to the ongoing efforts

**Multi-objectifying** MATSuMoTo: a surrogate-assisted library for SOPs *(Mueller, arXiv, 2014)*.

**Validate** its performance on the Bi-objective Black Box Optimization Benchmarking *(Tusar et al., arXiv, 2016)*.
Surrogate-Assisted Optimization

For MOPs: **exploration-exploitation-diversification** is sought.

Two approaches for Step 4:

A1 Using the surrogate model indirectly to generate a set of candidate points: the selected points for evaluation are the optimizers of a measure derived from the surrogate model (e.g., Emmerich et al., IEEE CEC, 2011).

A2 Using the surrogate model directly to generate a set of candidate points: a subset of these points are then selected for evaluation based on a set of rules (e.g., Akhtar & Shoemaker, JOPT, 2015).

Figure: Surrogate-assisted optimization framework.
Surrogate-Assisted Optimization

○ The first approach has been the focus of several optimization software packages (e.g., Binois & Picheny, GPareto, 2016).

○ The second approach lends itself naturally to the framework of the MATSuMoTo library for SOPs (Mueller, arXiv, 2014).

○ In this paper:
  * incorporate a variant of Approach A2 (GOMORS by Akhtar & Shoemaker, JOPT, 2015) into the MATSuMoTo library.
Table: Possible feature choices for the individual steps of MATSuMoTo. Highlighted choices: new features supporting multi-objective optimization problems.

<table>
<thead>
<tr>
<th>Algorithm Step</th>
<th>Choice Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Initial design</td>
<td>CORNER</td>
<td>Corner points of the hypercube</td>
</tr>
<tr>
<td></td>
<td>SLHD</td>
<td>Symmetric Latin hypercube</td>
</tr>
<tr>
<td></td>
<td>lhd</td>
<td>Latin hypercube</td>
</tr>
<tr>
<td>(3) Surrogate model</td>
<td>RBFcub</td>
<td>Cubic RBF</td>
</tr>
<tr>
<td></td>
<td>RBFgauss</td>
<td>Gaussian RBF</td>
</tr>
<tr>
<td></td>
<td>RBFtps</td>
<td>Thin-plate spline RBF</td>
</tr>
<tr>
<td></td>
<td>RBFlin</td>
<td>Linear RBF</td>
</tr>
<tr>
<td></td>
<td>MARS</td>
<td>Multivariate adaptive regression spline</td>
</tr>
<tr>
<td></td>
<td>POLYlin</td>
<td>Linear regression polynomial</td>
</tr>
<tr>
<td></td>
<td>POLYquad</td>
<td>Quadratic regression polynomial</td>
</tr>
<tr>
<td></td>
<td>POLYquadr</td>
<td>Reduced quadratic regression polynomial</td>
</tr>
<tr>
<td></td>
<td>POLYcub</td>
<td>Cubic regression polynomial</td>
</tr>
<tr>
<td></td>
<td>POLYcubr</td>
<td>Reduced cubic regression polynomial</td>
</tr>
<tr>
<td></td>
<td>MIX_RcM</td>
<td>Mixture of RBFcub and MARS</td>
</tr>
<tr>
<td></td>
<td>MIX_RcPc</td>
<td>Mixture of RBFcub and POLYcub</td>
</tr>
<tr>
<td></td>
<td>MIX_RcPcM</td>
<td>Mixture of RBFcub, POLYcub, and MARS</td>
</tr>
<tr>
<td>(4) Sampling strategy</td>
<td>CANDloc</td>
<td>Local candidate point search</td>
</tr>
<tr>
<td></td>
<td>CANDglob</td>
<td>Global candidate point search</td>
</tr>
<tr>
<td></td>
<td>SurfMin</td>
<td>Minimum point of surrogate model</td>
</tr>
<tr>
<td></td>
<td>SurfPareto</td>
<td>Pareto front of surrogate model (currently employs GOMORS)</td>
</tr>
</tbody>
</table>
Assessment

- Interfacing the Comparing Continuous Optimizer (COCO) platform with GPareto R package (too slow, weeks for $n = 20$!).

- Preliminary results qualified SMS-EGO

- Within MO-MATSuMoTo, SMS-EMOA (Beume et al., EJOR, 2007) and MO-DIRECT (Al-Dujaili & Suresh, CEC, 2016) used.$^2$

- SMS-EGO vs. MAT-SMS vs. MAT-DIRECT.

---

$^2$Available at [http://ash-aldujaili.github.io/projects.html](http://ash-aldujaili.github.io/projects.html)
Experimental Setup

COCO guidelines: data profiles and statistical test

55 problems based on bi-combinations of 24 noiseless functions:

- \( f_1 - f_5 \): separable functions
- \( f_6 - f_9 \): functions with low or moderate conditioning
- \( f_{10} - f_{14} \): functions with high conditioning and unimodal
- \( f_{15} - f_{19} \): multi-modal functions with adequate global structure
- \( f_{20} - f_{24} \): multi-modal functions with weak global structure

Dimensionality: 5-D, 10-D, 20-D, 40-D

Evaluation budget: \( 75 \cdot n \) (time limitation & slow GPareto)
Figure: Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for 121 targets with target precision in \( \{0, 10^{-0.19}, 10^{-0.18}, \ldots, 10^{0.98}, 10^{0.99}, 10^1\} \) over all the problems in \( n \in \{2, 3, 5, 10, 20\} \).
Performance Results

- Given this expensive budget setting, MAT-DIRCT and MAT-SMS show a comparable performance, outperforming SMS-EGO.
- With more function evaluations, SMS-EGO’s performance stagnates.
- On the other hand, MAT-DIRCT and MAT-SMS exhibit a gradual progress with more evaluations.
Insights & Issues

- Limited evaluation budget \((75 \cdot n)\) makes it difficult to reach a conclusive statement.

- **GPareto** R package:
  1. Extremely slow in higher dimension: R–MATLAB communication.
  2. Several run instances exited with run-time errors (error in optim function)

- Multi-objectifying **MATSuMoTo** with GOMORS (Akhtar & Shoemaker, JOPT, 2015):
  1. Ill-condition behavior after a batch of sampled points.
  2. Re-think about what kind of points are used to build the models.
BMOBench

- Inspired by COCO, we built BMOBench
- a platform with 100 MOPs.
- data profiles generated in terms of 4 quality indicators.
- Special session at SSCI'2016, Greece.\(^3\) (Deadline: 15-August-2016)
- We invite the multi-objective community to test their published/novel algorithms on these problems.

\(^3\)http://ash-aldujaili.github.io/BMOBench/
Thank you