SIP: Optimal Product Selection from Feature Models using Many-Objective Evolutionary Optimisation

Robert M. Hierons\textsuperscript{(1)}, Miqing Li\textsuperscript{(1)}, XiaoHui Liu\textsuperscript{(1)}, Sergio Segura\textsuperscript{(2)}, Wei Zheng\textsuperscript{(3)}

\textsuperscript{(1)} Brunel University London, UK
\textsuperscript{(2)} University of Seville, Spain
\textsuperscript{(3)} Northwestern Polytechnical University, China
Introduction
Software product lines

- Family of Products
- Built from a set of reusable assets
- There can be constraints
- Evidence of significant use in industry
Introduction
Feature Models

• Set of features

• There can be constraints such as:
  • Or/Alternative
  • Requires/Excludes
  • Mandatory

• A valid product is:
  • A set of features that satisfies the constraints

• An FM is typically represented as a tree
Introduction
Feature Models

\[
\text{E-Shop} \uparrow \\
\text{Catalogue} \quad \text{Payment} \quad \text{Security} \\
\text{Bank Transfer} \quad \text{Credit Card} \quad \text{High} \quad \text{Standard} \\
\text{Search} \\
\text{Public Report}
\]

= \{\text{E-shop} + \text{Catalogue} + \text{Payment} + \text{Credit Card} + \text{Security} + \text{High} + \text{Search}\}
Introduction

Automated analysis of feature models: Computer-aided extraction of information from FMs

How many products?

14
Introduction

Automated analysis of feature models: Computer-aided extraction of information from FMs

P = \{Mobile Phone + Screen + Colour + Media + Camera\}

Is P a valid product? No, two constraints are violated
Introduction
Feature models

• Much interest in the automated analysis of feature models.

• Catalogues with up to 30 analysis operations

• Tools that implement these operations
  • e.g. FaMa Framework, SPLOT, …

• Automation is via e.g. SAT solvers.
Introduction
Attributed feature models
Usually, it is not feasible to produce all valid products.

Reasons for choosing a particular product:

- What product to release first
  - many features, low cost, …

- What product to test
  - different combinations of features, low cost …
• We require a valid product:
  • We want to minimise the number of constraints failed.

• We want to optimise certain properties eg:
  • Cost (of building product)
  • Number of features
  • Historical faults
Ex: “Find a product that minimises the number of constraints violated, maximises the number of features, minimises cost, minimises the number of known defects, and minimises the number of changes since the last release.”
Problem

• Several aspects (objective functions) to optimise.
• So: a multi-objective optimisation problem.
• We could weight the objectives and optimise but ...
  • The choice of values for weights would be important.
  • How would we choose the weights?
Candidate (product) C1 dominates candidate C2 if:
- C1 is at least as good as C2 on all criteria
- C1 is better than C2 on at least one criterion.

Used by some evolutionary optimisation algorithms.
Aim is:
- get as close as possible to the Pareto front.

Returns a set of solutions:
- These are incomparable under Pareto dominance
- Provide alternative compromises.
Pareto Dominance

• Maximising – which dominate others?
Many-objective problems

- Many-objective problems
  - Those with four or more objectives
  - Typically, much more difficult to solve.

- Normally, Pareto dominance becomes ineffective

- There are algorithms that do not use Pareto dominance
  - e.g. IBEA, SPEA2+SDE, MOEA/D
Previous work

Recognised that it is a many-objective problem

Applied several evolutionary multi/many-objective optimisation algorithms (EOMs) including:
  - NSGA-II, SPEA2, IBEA

Used two feature models, up to 290 features.

Difficult to find valid products.

IBEA performed best.

Previous work
Enhancements

• Improvements by Sayad et al. included:
  • Removing core features (those that are in all valid products)
  • Removing dead features (those that are in no valid products)
  • Planting an initial seed (a valid product).


• Evaluated on more models:
  • Seven models.
  • Included one large model (6,888 features) – the Linux kernel feature (variability) model.

• Required an initial phase to find a seed
  • took approximately three hours for the large model.
Additional enhancements involved:

- Adapting mutation (cannot be applied in certain circumstances, avoiding some invalid products).
- Placing greater emphasis on the number of constraints that fail (a weight)


- New mutation and replacement operators that used a SAT solver

Previous work

Summary

• Search as a many-optimisation problem.
• Found to be difficult.
• A number of enhancements help.

• However:
  • Enhancements complicates the approach
  • New operators are more computationally expensive
  • Can be specific to certain optimisation algorithms

• Can we produce a simpler search-based approach?

Our initial work

• Initial motivation:
  • Try different evolutionary algorithms – those that have been found to be good at many-objective optimisation.

• We wanted approaches that return a range of solutions
  • Provide alternative products – user can choose between these.

• Initial experiments on two case studies:
  • WebPortal and E-Shop

• It was difficult to find valid products (similar to earlier work).
Finding products

- We conjectured that two aspects made this difficult:
  - The presence of many constraints.
  - As the number of objectives increases, there is less evolutionary pressure towards valid products.

- We aimed to address these two points.

- We also aimed to use case studies with realistic attribute values.
Novel encoding (Shrink)

Prioritised objectives (Prioritise)

Realistic case studies

EMO algorithms comparison

Approach

SIP: Shrink and Prioritise
Approach
SIP: Shrink and Prioritise

Novel encoding (ShrInk)

Prioritized objectives (Prioritise)

1 + n

Realistic case studies

EMO algorithms comparison
Direct

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
</table>

Approach
Standard encoding
Approach
Novel encoding

• Basic idea:
  • No need to include Core features (Sayad at al. and Henard et al. also observed this).
  • Did not remove dead features
    • Feature models should not have these.
    • They should be found and removed in advance.
  • Sometimes we don’t need to include some parents (e.g. if children are in a OR relationship).
Approach
Novel encoding

- GPS
  - Routing
  - Traffic avoiding
  - Radio
    - AM
    - FM
    - Digital
  - Interface
    - Keyboard
    - Screen
      - Touch
      - LCD

### Encoding

<table>
<thead>
<tr>
<th>Encoding</th>
<th>GPS</th>
<th>Routing</th>
<th>3D map</th>
<th>Auto-rerouting</th>
<th>Traffic avoiding</th>
<th>Radio</th>
<th>FM</th>
<th>AM</th>
<th>Digital</th>
<th>Interface</th>
<th>Keyboard</th>
<th>Screen</th>
<th>Touch</th>
<th>LCD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>SIP</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>
Approach
Novel encoding

• This helps since:

  • We have a smaller search space.

  • Some constraints can be enforced through the encoding.
    • Should make it easier to generate valid products.
Approach
SIP: Shrink and Prioritise

Novel encoding (Shrink)

Prioritized objectives (Prioritise)

1 + n

Realistic case studies

EMO algorithms comparison
Approach
Prioritised objectives: 1+n

• Traditional, (n+1)-approach:
  • All of the objectives are considered together.

• The 1+n-approach:
  • We first consider the number of constraints failed, only then the remaining (n) objectives.
  • Motivation: invalid products have no value.

• Implemented as:
  • First compare individuals using number of constraints failed.
  • If equal on this, then compare as normal (for the given algorithm).
Approach
Implementing 1+n

- Mating selection:
  - Compare individuals on number of constraints failed.
  - If equal on this, then compare as normal (for the given algorithm).
- Environmental selection (determining which survive):
  - Group individuals on number of constraints failed.
  - Keep the best groups.
  - Use normal approach (for EMO) if we need to select a subset from a group.
Approach
Prioritised objectives

Minimise complexity
Minimise changes
Minimise violations

Maximise features
Minimise size

n+1

1+n
**Approach**

SPI: Shrink and Prioritise

- **Novel encoding** (ShrInk)
- **Prioritized objectives** (Prioritise)
- **Realistic case studies**
- **EMO algorithms comparison**

![Diagram](image)
## Approach

**Realistic case studies**

<table>
<thead>
<tr>
<th></th>
<th>DRUPAL</th>
<th>Amazon Web Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Features</td>
<td>48</td>
<td>79</td>
</tr>
<tr>
<td>Attributes</td>
<td>22</td>
<td>17</td>
</tr>
<tr>
<td>Products</td>
<td>$2.09 \cdot 10^9$</td>
<td>66,528</td>
</tr>
</tbody>
</table>

- Published FMs with random attributes.
- Randomly generated FM with 10,000 features.
**Approach**

**SIP: Shrink and Prioritise**

- **Novel encoding (Shrink)**
- **Prioritised objectives (Prioritise)**
- **Realistic case studies**
- **EMO algorithms comparison**

1 + n
• Four used:
  • *Direct encoding* (one bit for each feature). Used in initial work by Sayyard et al.
  • *Core encoding* – do not include core features. Used by Sayyad et al. and Henard et al.
  • *Hierarchical encoding* – do not include parents unless necessary
  • The (proposed) *novel encoding*
    • Combines the above enhancements
    • (core + hierarchical)
Evaluation
EMO algorithms

NSGA-II
[Deb et al. 2002]

IBEA
[Zitzler et al. 2004]

SPEA2+SDE
[Li et al. 2014]

MOEA/D-WS
[Zhang and Li 2007]

MOEA/D-TCH
[Zhang and Li 2007]

MOEA/D-PBI
[Zhang and Li 2007]
Evaluation
Initial experiments

- We initially used:
  - (n+1)-approach
  - Direct encoding
  - Two feature models (E-shop and WebPortal).

- Found that:
  - All EMOs returned relatively few valid products.

- Motivated the enhancements.
<table>
<thead>
<tr>
<th>SPL</th>
<th># Features</th>
<th># CTC</th>
<th># Attributes Per feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>BerkleyDB</td>
<td>13</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>ERS</td>
<td>36</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>WebPortal</td>
<td>43</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>E-shop</td>
<td>290</td>
<td>21</td>
<td>4</td>
</tr>
<tr>
<td>Drupal</td>
<td>48</td>
<td>21</td>
<td>22</td>
</tr>
<tr>
<td>AmazonEC2</td>
<td>79</td>
<td>0</td>
<td>17</td>
</tr>
<tr>
<td>Random (generated by SPLAR)</td>
<td>10,000</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

- All feature models were written in the SXFM format.

First four previously used
For the four previously used models:
  • We used the same five objectives as previous experiments
For Amazon and Drupal, two experiments for each:
  • We used the same five objectives (synthetic values).
  • We had eight objectives for each (realistic values).
For the large randomly generated model:
  • The same five objectives as previous work (for comparison).

All sets included correctness (number of constraints not satisfied).
Subjects with realistic attributes

• Two case studies.

• Drupal:
  • Previously, 22 non-functional attributes extracted from GIT repository.

• Amazon EC2:
  • Configuration space of Amazon Elastic Computing Service.
  • Attributes were not fixed – there are constraints.
  • We randomly assigned values that satisfied the constraints.
Evaluation
Experimental setup

- Each experiment repeated 30 times.
- Termination criterion: 50,000 fitness evaluations
- Population size:
  - 100 for most
  - For MOEA/D – closest possible integer to 100 (depends on number of objectives).

- Parameter values used:
  - Those recommended in the literature.
Evaluation
Performance metrics

Three metrics:
- Hypervolume (HV): the volume of the objective space between the solutions and a reference point.
  - Only used the valid solutions returned.
- The number of executions that returned at least one valid product (VN).
- Mean rate of valid individuals in the final population (VR).
- Note that HV and VR were averaged over populations that had at least one valid solution.
Results
### Results

Results with E-shop

- An example, direct encoding and (n+1)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>HV</th>
<th>VN (/30)</th>
<th>VR</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSGA-II</td>
<td>0.0000</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>IBEA</td>
<td>0.0000</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>MOEA/D-WS</td>
<td>0.016184</td>
<td>26</td>
<td>21.42%</td>
</tr>
<tr>
<td>MOEA/D-TCH</td>
<td>0.0000</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>MOEA/D-PBI</td>
<td>0.18815</td>
<td>4</td>
<td>10.50%</td>
</tr>
<tr>
<td>SPEA2+SDE</td>
<td>0.0000</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>
### Results

Results with E-shop

- An example, novel encoding and (n+1)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>HV</th>
<th>VN (/30)</th>
<th>VR</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSGA-II</td>
<td>0.136545</td>
<td>13</td>
<td>100%</td>
</tr>
<tr>
<td>IBEA</td>
<td>0.169191</td>
<td>16</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-WS</td>
<td>0.184810</td>
<td>5</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-TCH</td>
<td>0.199697</td>
<td>1</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-PBI</td>
<td>0.166157</td>
<td>5</td>
<td>100%</td>
</tr>
<tr>
<td>SPEA2+SDE</td>
<td>0.144341</td>
<td>15</td>
<td>100%</td>
</tr>
</tbody>
</table>
Results with E-shop

- An example, core encoding and (n+1)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>HV</th>
<th>VN (/30)</th>
<th>VR</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSGA-II</td>
<td>0.003343</td>
<td>28</td>
<td>2.07%</td>
</tr>
<tr>
<td>IBEA</td>
<td>0.26741</td>
<td>30</td>
<td>33.91%</td>
</tr>
<tr>
<td>MOEA/D-WS</td>
<td>0.074223</td>
<td>30</td>
<td>26.74%</td>
</tr>
<tr>
<td>MOEA/D-TCH</td>
<td>0.0000</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>MOEA/D-PBI</td>
<td>0.070765</td>
<td>30</td>
<td>30.62%</td>
</tr>
<tr>
<td>SPEA2+SDE</td>
<td>0.0000</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>
Results

Results with E-shop

- An example, novel encoding and 1+n

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>HV</th>
<th>VN (/30)</th>
<th>VR</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSGA-II</td>
<td>0.162943</td>
<td>30</td>
<td>100%</td>
</tr>
<tr>
<td>IBEA</td>
<td>0.190496</td>
<td>30</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-WS</td>
<td>0.222875</td>
<td>30</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-TCH</td>
<td>0.226257</td>
<td>30</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-PBI</td>
<td>0.192485</td>
<td>30</td>
<td>100%</td>
</tr>
<tr>
<td>SPEA2+SDE</td>
<td>0.159930</td>
<td>30</td>
<td>100%</td>
</tr>
</tbody>
</table>
Results
Results with E-shop

• Direct encoding:
  • With, \( (n+1) \), only two algorithms returned valid solutions (VR of 4/30 and 26/30).
  • With \( 1+n \), all (sometimes) produced valid solutions.

• Hierarchical and core encodings:
  • More effective, core better than hierarchical.

• Novel encoding:
  • Almost all EMOs had 100% VR.
Results
Results with E-shop

- HV for novel:
  - With 1+n, the best results were with some variants of MOEA/D.
  - With (n+1), IBEA best (similar to previous work).

- We tried E-shop with 500,000 evaluations:
  - Better results.
  - However, only 1+n (with core/novel) gave 100% VN for all algorithms.
Results
Results with Webportal

• Results similar to E-shop but better.
  • Seems to be an easier problem.

• 1+n approach always gave 100% VN and VR.

• (n+1) approach gave 100% VN, VR for core and novel encodings.

• IBEA tended to give the best HV.
Results

Other results (five objectives)

• Similar results.

• 1+n approach and novel always gave 100% VN and VR.

• The only combination that did this.

• HV comparisons were quite variable. Highest values for:
  • SPEA2+SDE (Amazon, Berkley)
  • IBEA (Drupal)
  • NSGA-II (ERS).
Results
Other results (real attributes)

• Drupal:
  • Relatively easy to find valid products.
  • Only \((n+1)/\text{direct}\) failed to have 100% VR,VN for all EMOs.

• Amazon:
  • Results were much poorer.
  • Almost no experiments produced valid products with \(\text{direct}/(n+1)\).
  • Most were effective with novel/1+n.
Results
Amazon (real attributes)

- Results for 1+n with the novel encoding.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>HV</th>
<th>VN (/30)</th>
<th>VR</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSGA-II</td>
<td>0.001844</td>
<td>30</td>
<td>100%</td>
</tr>
<tr>
<td>IBEA</td>
<td>0.001897</td>
<td>17</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-WS</td>
<td>0.001877</td>
<td>30</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-TCH</td>
<td>0.001688</td>
<td>30</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-PBI</td>
<td>0</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>SPEA2+SDE</td>
<td>0.002001</td>
<td>30</td>
<td>100%</td>
</tr>
</tbody>
</table>
Results
Large, randomly generated model

- Search found this difficult.
- Only novel/1+n returned valid solutions
  - All others had 0% for VN, VR.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>HV</th>
<th>VN (/30)</th>
<th>VR</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSGA-II</td>
<td>0.014588</td>
<td>24</td>
<td>100%</td>
</tr>
<tr>
<td>IBEA</td>
<td>0.020762</td>
<td>25</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-WS</td>
<td>0.042142</td>
<td>15</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-TCH</td>
<td>0.025037</td>
<td>19</td>
<td>100%</td>
</tr>
<tr>
<td>MOEA/D-PBI</td>
<td>0.042513</td>
<td>18</td>
<td>100%</td>
</tr>
<tr>
<td>SPEA2+SDE</td>
<td>0.018173</td>
<td>28</td>
<td>100%</td>
</tr>
</tbody>
</table>
### Results
Best performing (novel, 1+n)

- Comparing HV values.
- Not all differences are statistically significant (many were).

<table>
<thead>
<tr>
<th>Model</th>
<th>Best performing MOEA</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-shop. 50k</td>
<td>MOEA/D-TCH</td>
</tr>
<tr>
<td>E-shop, 500k</td>
<td>MOEA/D-WS</td>
</tr>
<tr>
<td>WebPortal</td>
<td>IBEA</td>
</tr>
<tr>
<td>Amazon</td>
<td>SPEA2+SDE</td>
</tr>
<tr>
<td>Berkley</td>
<td>SPEA2+SDE</td>
</tr>
<tr>
<td>Drupal</td>
<td>IBEA</td>
</tr>
<tr>
<td>ERS</td>
<td>NSGA-II</td>
</tr>
<tr>
<td>Drupal, real</td>
<td>SPEA2+SDE</td>
</tr>
<tr>
<td>Amazon, real</td>
<td>SPEA2+SDE</td>
</tr>
<tr>
<td>Large, random</td>
<td>SPEA2+SDE</td>
</tr>
</tbody>
</table>
Statistical tests used

• First tested the hypothesis that all EMO algorithms perform equally (different techniques, same approach):
  • Kruskal-Wallis test. Always rejected.
• Then used pairwise comparisons:
  • post-hoc Kruskal-Wallis test
  • with Bonferroni adjustment
• Effect size using:
  • Mann-Whitney U test
  • many of the differences had large effect sizes (between 0.8 and 0.9).
Time in seconds.
For novel encoding; results similar (but usually slower) for other encodings.

<table>
<thead>
<tr>
<th>EMO</th>
<th>(n+1)</th>
<th>1+n</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSGA-II</td>
<td>101.76</td>
<td>98.758</td>
</tr>
<tr>
<td>IBEA</td>
<td>154.399</td>
<td>131.465</td>
</tr>
<tr>
<td>MOEA/D-WS</td>
<td>135.98</td>
<td>104.44</td>
</tr>
<tr>
<td>MOEA/D-TCH</td>
<td>132.9</td>
<td>99.7</td>
</tr>
<tr>
<td>MOEA/D-PBI</td>
<td>131.9</td>
<td>103</td>
</tr>
<tr>
<td>SPEA2+SDE</td>
<td>247.786</td>
<td>223.777</td>
</tr>
</tbody>
</table>
Results

The choice of the encoding and approach have more impact on performance than the choice of EMO algorithm.

There is not clear “best” EMO algorithm.

The 1+n approach, where one objective is given priority over the rest, is more effective that treating all the objectives equally.

The encoding does affect performance, with the proposed encoding proving to be most effective.
Future Work

• Incorporate novel operators from e.g. Henard et al.

• Further enhance the representation:
  • Aim to encode more constraints.

• Consider approaches that use repair.

• Additional studies:
  • Others with realistic attribute values – do the approaches perform less well on these?
  • Varying the number of objectives.
Summary

- The encoding affects performance, with the proposed encoding providing to be most effective.

- The 1+n approach is more effective than the (n+1) approach.

- Both enhancements help.

- There is no clear “best” EMO algorithm.

- SPEA2+SDE performs best most often (and for the most difficult) but takes more time.
SIP: Optimal Product Selection from Feature Models using Many-Objective Evolutionary Optimisation

Robert M. Hierons\textsuperscript{(1)}, Miqing Li\textsuperscript{(1)}, XiaoHui Liu\textsuperscript{(1)}, Sergio Segura\textsuperscript{(2)}, Wei Zheng\textsuperscript{(3)}

\textsuperscript{(1)} Brunel University London, UK
\textsuperscript{(2)} University of Seville, Spain
\textsuperscript{(3)} Northwestern Polytechnical University, China

Thanks!
Hypervolume

- Influenced by:
  - Choice of reference point.
  - Scaling of space (dimensions).

- Reference point used:
  - Nadir point of the problem’s range (the point constructed with the worst value on each objective).

- Scaling:
  - we normalised the objective values according to the range of values in the objective space.
Hypervolume

- Computed exact values when fewer than eight objectives.

- Literature suggests this is infeasible with seven or more objectives.

- For the large example (eight objectives)
  - We estimated the HV result.
  - We used Monte Carlo sampling with 10,000,000 sampling points.