Towards a Multi-Objective Modularization Approach for Entity-Relationship Models

Dominik Bork, Antonio Garmendia, Manuel Wimmer

November, 4th 2020
Research Context

• Problem
  – Data models often evolve into large monolithic artefacts
  – Modularization enables handling complexity, but how?

• Objective
  – Intelligent approach to automatically modularize large ER models
  – Formulation as Many-objective problem to be solved by Search-based Software Engineering using Genetic Algorithms

• Solution Overview
  – We use Conceptual Modeling and Model-driven Development

Focus

We realize a Generic Encoding

– We implement the ModulER Tool
The ModulER Approach: Overview

1. Initialize Population
2. Encoding
3. Fitness Functions
4. End
5. NSGAII
6. Crossover
7. Mutation
8. Constraints

ER Model

- Evaluation
- Yes → Results
- No → Selection
- Constraints

- Selection
- Partially-matched Crossover
- Multi-point Crossover
- Mutations
- Swap Mutation
- Flip Mutation

- Pareto set of optimal solutions

References:

The ModulER Metamodel

- **ModuleERModel**
  - modules 1..*

- **NamedElement**
  - name : String

- **ModularizableElement**
  - modElements 1..*

- **Module**

- **Element**
  - isWeak:Boolean

- **EntityType**
  - entity 1

- **RelationshipType**

- **Link**
  - type:CardinalityType

- **<enumeration>>**
  - CardinalityType
    - Zero_To_One
    - Zero_To_Many
    - One_To_One
    - One_To_Many
ModulER Encoding

Module 1

- 9: MemberOf
- 1: Student
- 10: Family
- 4: StudentCustomer

Module 2

- 2: Lives
- 3: Address
- 5: Makes
- 6: Order
- 7: Contains
- 8: Item

Position | Chromosome |
---------|------------|
1        | 1          |
2        | 0          |
3        | 4          |
4        | 0          |
5        | 9          |
6        | 0          |
7        | 10         |
8        | 1          |
9        | 2          |
10       | 0          |
11       | 3          |
12       | 0          |
13       | 5          |
14       | 0          |
15       | 6          |
16       | 0          |
17       | 7          |
18       | 0          |
19       | 8          |

Transform into

Position | Chromosome |
---------|------------|
1        | 1          |
2        | 4          |
3        | 9          |
4        | 10         |
5        | 2          |
6        | 3          |
7        | 5          |
8        | 6          |
9        | 7          |
10       | 8          |
11       | 0          |
12       | 0          |
13       | 0          |
14       | 1          |
15       | 0          |
16       | 0          |
17       | 0          |
18       | 0          |
19       | 0          |

Enumerable Chromosome

Bit Chromosome
ModulER Fitness Functions & Alterers

<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>COH</td>
<td>Cohesion: the sum of links within modules.</td>
<td>MAXIMIZE</td>
</tr>
<tr>
<td>COP</td>
<td>Coupling: the sum of links between modules.</td>
<td>MINIMIZE</td>
</tr>
<tr>
<td>NMOD</td>
<td>Number of modules.</td>
<td>MINIMIZE</td>
</tr>
<tr>
<td>AVGMODEL</td>
<td>Average number of modularizable elements per module.</td>
<td>MINIMIZE</td>
</tr>
<tr>
<td>BAL</td>
<td>The standard deviation of module size of all modules.</td>
<td>MINIMIZE</td>
</tr>
</tbody>
</table>

Alterers

• Crossover and Mutation used to alter
  – the assignment of modularizable elements to modules
    > Partially-matched Crossover and Swap Mutator
  – the number of modules
    > Multi-point Crossover and Flip Mutator
ModulER Alterers: Crossover

<table>
<thead>
<tr>
<th>Position</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chromosome</td>
<td>1</td>
<td>4</td>
<td>9</td>
<td>10</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Enumerable Chromosome**

**Bit Chromosome**

**Crossover**

**Partially-matched Crossover (PMX)**

**Multi-point Crossover (MX)**

**Mutators**

<table>
<thead>
<tr>
<th>Mutator</th>
<th>SwapMutator</th>
<th>FlipMutator</th>
</tr>
</thead>
</table>

**Initialize Population**

**Evaluation**

**Selection**

**Constraints**

**Crossover**

**Mutation**

**Constraints**

**Selection**

**Crossover**

**Mutation**

**Partially-matched Crossover (PMX)**

**Multi-point Crossover (MX)**
# ModulER Alterers: Mutators

<table>
<thead>
<tr>
<th>Position</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chromosome</td>
<td>1</td>
<td>4</td>
<td>9</td>
<td>10</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
</tbody>
</table>

**Enumerable Chromosome**

**Bit Chromosome**

### Crossover

- Partially-matched Crossover (PMX)
- Multi-point Crossover (MX)

### Mutators

- SwapMutator
- FlipMutator

---

**Initialize Population**

**Evaluation**

**Selection**

**Constraints**

**Crossover**

**Mutation**

**End**
ModulER Constraints

Two constraints to steer the GA toward valid solutions

- **RelationshipToEntity**
  A RelationshipType should always be in one of the modules the two entities it relates are in.

- **NumberOfModules**
  The number of modules and the module size should be comprehensible for humans.
Selected Evaluation Results I

- **RQ1 Search Validation ✓**
  Is the approach capable of efficiently modularizing ER models?

<table>
<thead>
<tr>
<th>Case</th>
<th>Entity Types</th>
<th>Relationship Types</th>
<th>Inheritance</th>
<th>Attributes</th>
<th>Avg. Computation time [Seconds]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Karate</td>
<td>12</td>
<td>13</td>
<td>1</td>
<td>29</td>
<td>1.8</td>
</tr>
<tr>
<td>Finance</td>
<td>116</td>
<td>17</td>
<td>97</td>
<td>67</td>
<td>6.8</td>
</tr>
</tbody>
</table>

✓ Efficient computation of the Pareto set
✓ Only valid solutions w.r.t. the ModulER metamodel and the ModulER constraints
Selected Evaluation Results II

- **RQ2 Solution Correctness**

  How good is the quality of the produced solutions and how good are the solutions compared to ones conceptual modelers create?

  **Pareto set (n = 70)**
  - Heterogeneous solutions
  - Fulfilling the different objectives to different extends

  **Modeler solutions (n = 7)**
  - Homogeneous solutions
  - All with 3 modules
  - Focus on cohesion

  \[ \text{Avg. Edits: 5} \]

  Outperform in all objectives
Conclusions and Future Research

• The algorithm efficiently produces good modularizations
• More empirical research required
  – Larger sample size
  – More complex examples
  – Thematic clustering
• Next steps: Integrating the Modeler in-the-loop\(^1\)
• Extending the approach to
  – Consider Extended Entity Relationship model concepts
  – Consider hierarchical modularizations

\(^1\) Kessentini, Wimmer, Sahraoui (2018): Integrating the designer in-the-loop for metamodel/model co-evolution via interactive computational search. In MODELS’18 (pp. 101-111)
Thank you!