Multiobjective Optimisation of Combinatorial Libraries

Trudi Wright
University of Sheffield
Overview

- Multiobjective problems (MOP).
- Combinatorial library design as a MOP.
- Solving library design problems.
  - Applications.
  - Design issues and concepts.
Multiobjective Problems

- Optimise many objectives.
- Objectives.
  - Uncorrelated.
  - Non-commensurable.
  - Often competing.
- No one unique optimal solution exists, but set of near optimal solutions
Library Design

- Aminothiazole library.

\[
\begin{align*}
R_1 & \quad N & \quad S & \quad NH_2 \\
R_2 & & & \\
\end{align*}
\begin{align*}
+ & \quad Br & \quad O & \\
R_3 & & \quad R_4 & \\
\end{align*}
\Rightarrow \begin{align*}
R_1 & \quad N & \quad S & \quad N & \quad R_2 \\
& & & \quad R_3 & \quad R_4 & \\
\end{align*}
\]

- Virtual library 12850 molecules (74 α – bromoketones, 170 thioureas).

- Optimise selection of combinatorial subsets across multiple objectives.
Objectives in Library Design

- **Size.**
  - Minimum size/number reactants.

- **Diversity.**
  - Cell based.

- **Drug-like properties.**
  - E.g. rotatable bonds.
Solving a Library Design Problem

- Involves search and decision.
  - Search - identify solutions in search space.
  - Decision making – select suitable compromise solution (usually human intervention).
- Decision making before search.
  - E.g. SELECT.
- Decision making after search.
  - E.g. MoSELECT.
Decision Before Search

- Decision.
  - Size Objective.
    - Specify fixed library configuration.
  - Diversity & Drug-like Objectives.
    - Aggregate into single weighted sum fitness function.
    - E.g. \( f(n) = w_1 \cdot \text{diversity} + w_2 \cdot \Delta MW \)

- Search.
  - Optimise single objective \((f)\).
Results Decision Before Search

![Graph](image)

- $w_1=1.0; w_2=1.0$
- $w_1=1.0; w_2=0.5$
- $w_1=10; w_2=1.0$
Limitations Decision Before Search

- Specifying configuration difficult.
- Setting weights is difficult for different objectives.
- The use of weights obscures regions of the search space.
- A single compromise solution is found when usually a family of equally valid alternatives exist.
Decision After Search (MoSELECT)

- Search identifies set of near optimal solutions.
  - Objectives handled independently to explore multiple solutions in parallel.
  - Based on Pareto method.
  - MultiObjective Genetic Algorithm (MOGA).
- Decision.
  - Select compromise solution.
Decision After Search

- Search
  - Pareto optimality
  - Defines set of optimal trade-offs
  - All objectives equally important

- Decision making
  - Choose best compromise
  - Include preference information
Fitness - Pareto Ranking

- **Dominance.**
  - A non-dominated solution is one for which there is no other solution better in all objectives.

- **Pareto ranking.**
  - An solution’s rank corresponds to the number of solutions in a population by which it is dominated.
Issues

- How to maintain a diverse solution.
- How to prevent nondominated solutions from being lost.
- How to guide solutions towards the Pareto set.
- Monitoring convergence.
MOGA

Population

Rank

Niche

Archive

Converged

Has Pareto front moved

Genetic Operators

New Population

Yes

Has Pareto front moved

No

prevent solutions from being lost

diverse set solutions

monitor convergence

guide solutions

Yes

Converged

No
Objectives

- Size objective.
  - Allow solution libraries variable configurations.

- Maximise diversity.
- Minimise difference in drug-like profiles.
Results

Library Size

Occupied Cells

- MOGA (single run)
- SELECT (multiple runs)
Increasing the Objectives

<table>
<thead>
<tr>
<th>SOLN</th>
<th>AMW</th>
<th>CELL</th>
<th>SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>364</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>A</td>
<td>0.8</td>
<td>361</td>
<td>1886</td>
</tr>
<tr>
<td>B</td>
<td>0.2</td>
<td>29</td>
<td>31</td>
</tr>
<tr>
<td>C</td>
<td>0.7</td>
<td>289</td>
<td>627</td>
</tr>
</tbody>
</table>
Increasing the Objectives
Presenting Solutions

- Increasing objectives massively increases search space.
  - Results in many solutions.

- How much information is useful to the end user?
  - Need to determine which non-dominated solutions are useful.
Reducing the Solution Space - Niching

- Increase niche radius
- But poor solutions
Reducing the Search Space - Constraints

- Restrictions imposed by environment or resources i.e.
  - Library size.
  - Plate coverage.
  - Combinatorial efficiency.
- Formulated as hard constraints.
- Feasible solutions satisfy restrictions.
Constraint Handling

- Infeasible solutions penalised during Pareto ranking
  - Initialise chromosomes with feasible solutions
  - Form subpopulations of feasible and infeasible
  - Rank within subpopulation
  - Increase rank of infeasible subpopulations
Constraints

- Restrictions imposed by environment or resources i.e.
  - Plate coverage.
  - Library size.
  - Combinatorial efficiency.
Plate Coverage

- Libraries are stored on plates one compound per well.
- Constraint.
  - All feasible libraries must be multiple of 96.
Results - No Constraint

- Occupied Cells
  - Y-axis: 0 to 400
  - Library Size: 0 to 1500

Graph showing the relationship between library size and occupied cells.
Results - Plate Constraint

- Applying plate constraint of 100% to the Aminothiazole library.
Constraints

- Restrictions imposed by environment or resources i.e.
  - Plate coverage.
  - Library size.
  - Combinatorial efficiency.
Results - Size Constraint
Results - Size Constraint

The graph shows the relationship between library size and occupied cells for two conditions: no constraint and constraint 200-400. The x-axis represents library size, while the y-axis represents occupied cells.

- **No Constraint**: Blue triangles indicating a smooth increase in occupied cells as library size increases.
- **Constraint 200-400**: Red triangles showing a slight deviation from the no-constraint trend, indicating constraint effect.

The graph illustrates how constraining the library size affects the number of occupied cells.
Results - Size Constraint

![Graph showing occupied cells vs library size with two conditions: no constraint and constraint 400-600.](image)
Results - Size Constraint

- No constraint
- Constraint 600-800

Occupied Cells vs. Library Size

- No constraint
- Constraint 600-800
Constraints

- Restrictions imposed by environment or resources i.e.
  - Plate coverage.
  - Library size.
  - Combinatorial efficiency.
Combinatorial Efficiency

- Number R2
- Number R1

- Constraint
- No constraint

Ideal efficiency
Combinatorial Efficiency

Library Size vs Occupied Cells for UNCONSTRAINED and CONSTRAINED cases.
Conclusions

- **SELECT**
  - Specify size & configuration
  - Weighting of objectives
  - Single solution found

- **MoSELECT**
  - Size & configuration optimised as objective
  - No weighting objectives handled independently
  - Choice of solutions
  - Flexible
Acknowledgements

- Sheffield
  - Val Gillet
  - Peter Fleming
  - Robin Pursehouse

- GSK
  - Stephen Pickett
  - Darren Green