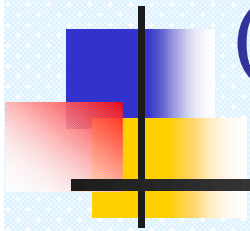


Multiobjective Optimisation of Combinatorial Libraries



Trudi Wright
University of Sheffield





Overview

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



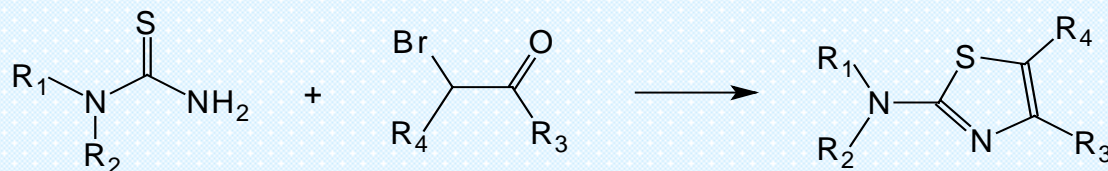
Multiobjective Problems

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



Library Design

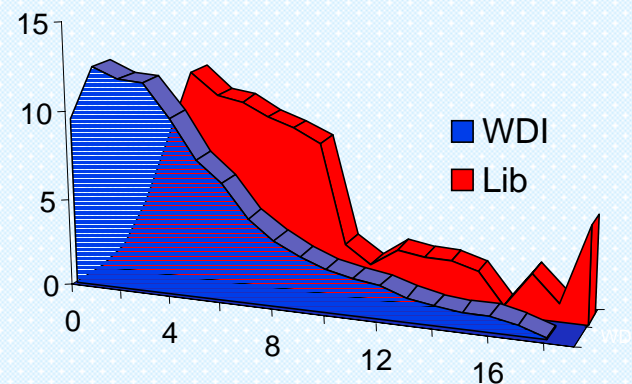
- Aminothiazole library.



- 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

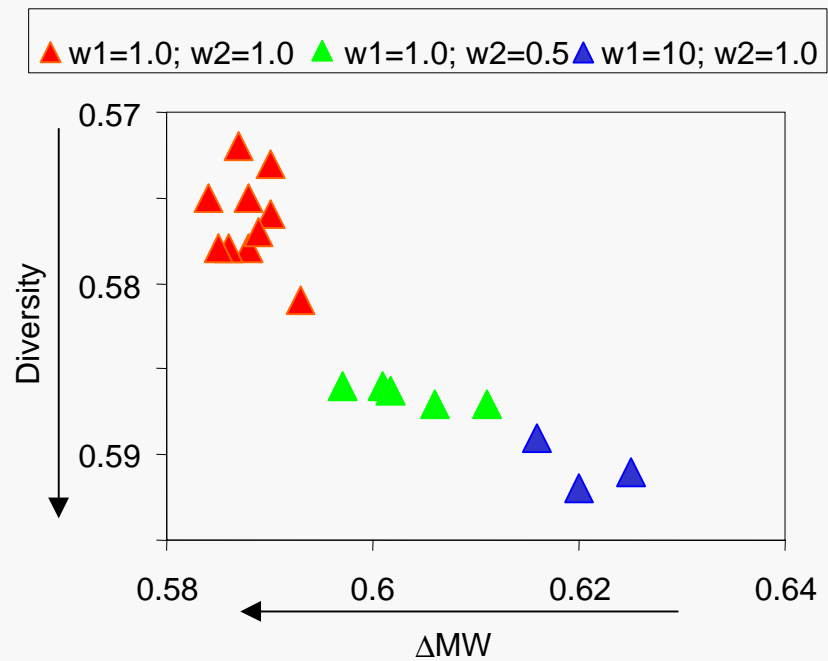
- Involves search and decision.
 - Search - identify solutions in search space.
 - Decision making – select suitable compromise solution (usually human intervention).
- Decision making before search.
 - E.g. SELECT.
- Decision making after search.
 - E.g. MoSELECT.



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





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.

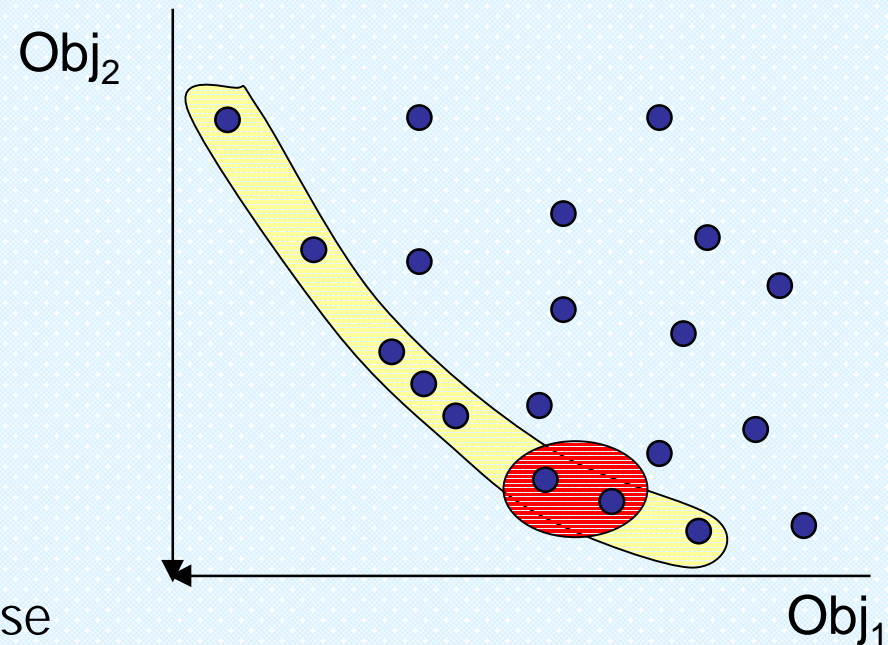


Decision After Search (MoSELECT)

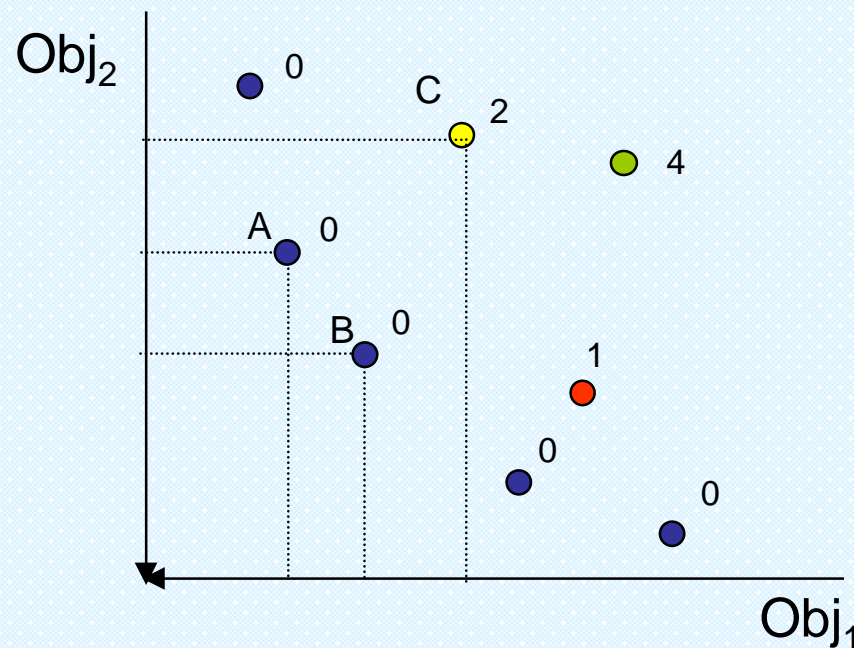
- Search identifies set of near optimal solutions.
 - Objectives handled independently to explore multiple solutions in parallel.
 - Based on Pareto method.
 - MultiObjective Genetic Algorithm (MOGA).
- Decision.
 - Select compromise solution.

Decision After Search

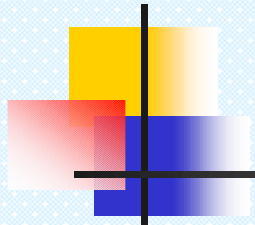
- Search
 - Pareto optimality
 - Defines set of optimal trade-offs
 - All objectives equally important
- Decision making
 - Choose best compromise
 - Include preference information



Fitness - Pareto Ranking



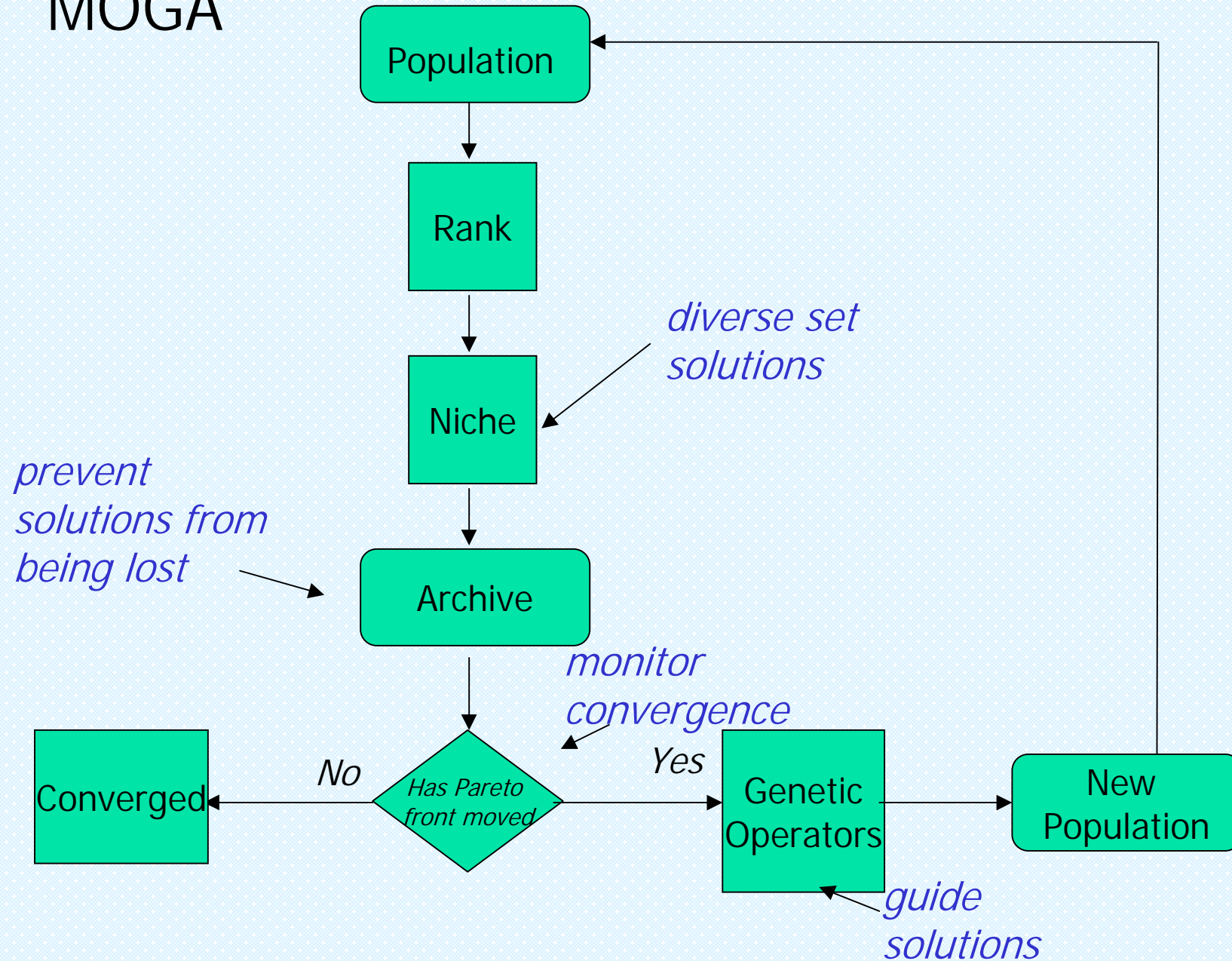
- Dominance.
 - A non-dominated soln is one for which there is no other soln better in all objectives.
- Pareto ranking.
 - An soln's rank corresponds to the number of solns in a population by which it is dominated.



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





Objectives

- Size objective.
 - Allow solution libraries variable configurations.

R1

0	1	1	0	1	0	1	1	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---

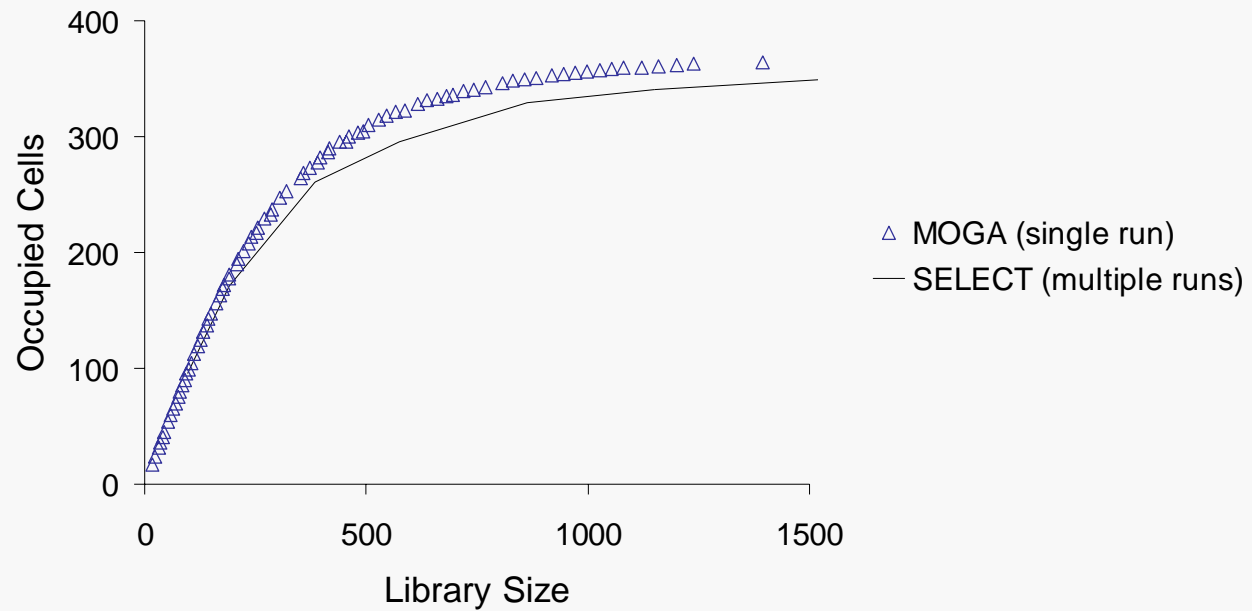
R2

1	0	1	1	0	1
---	---	---	---	---	---

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

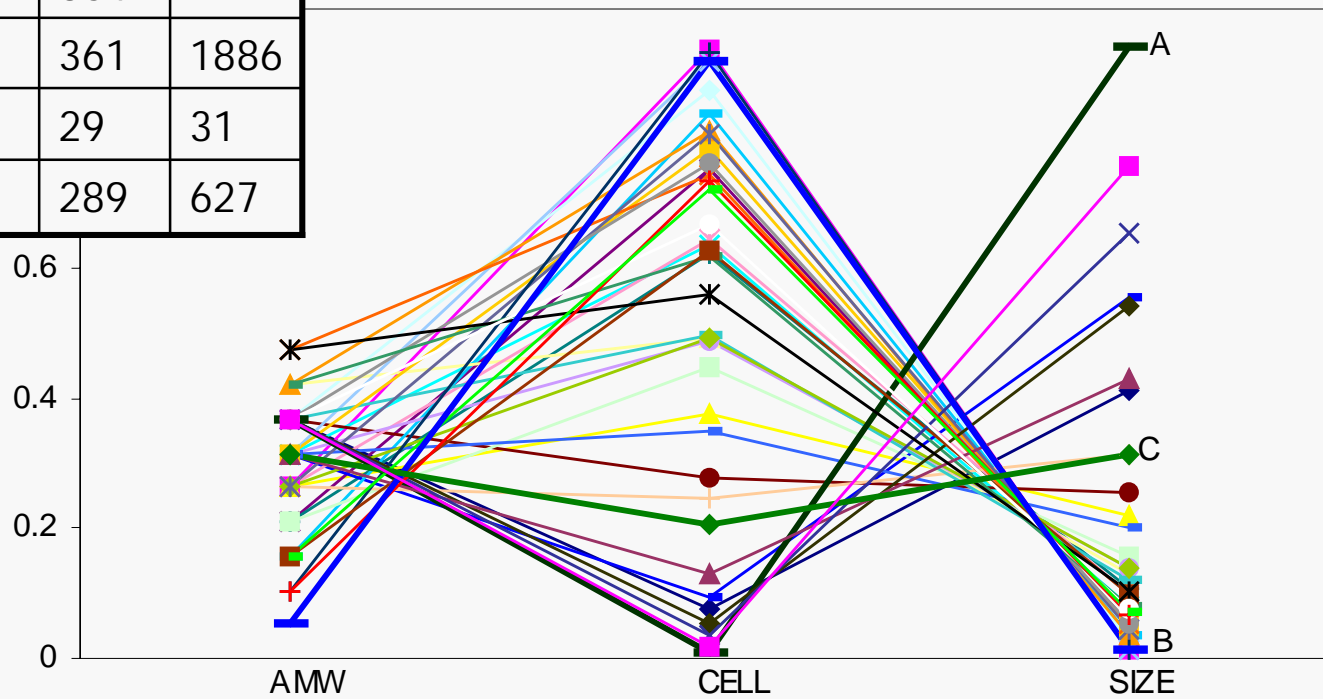


Results

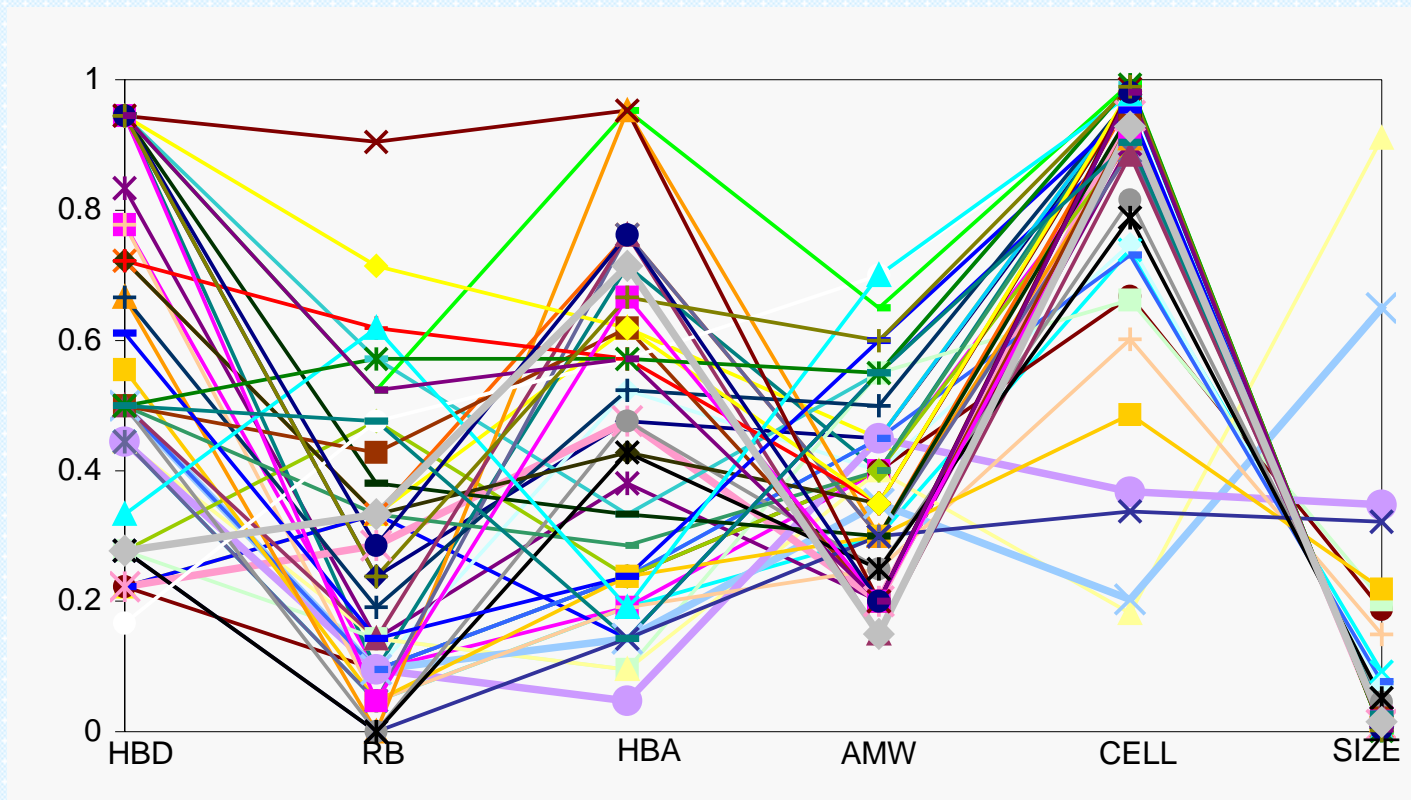


Increasing the Objectives

SOLN	AMW	CELL	SIZE
	0.1	364	-
A	0.8	361	1886
B	0.2	29	31
C	0.7	289	627



Increasing the Objectives



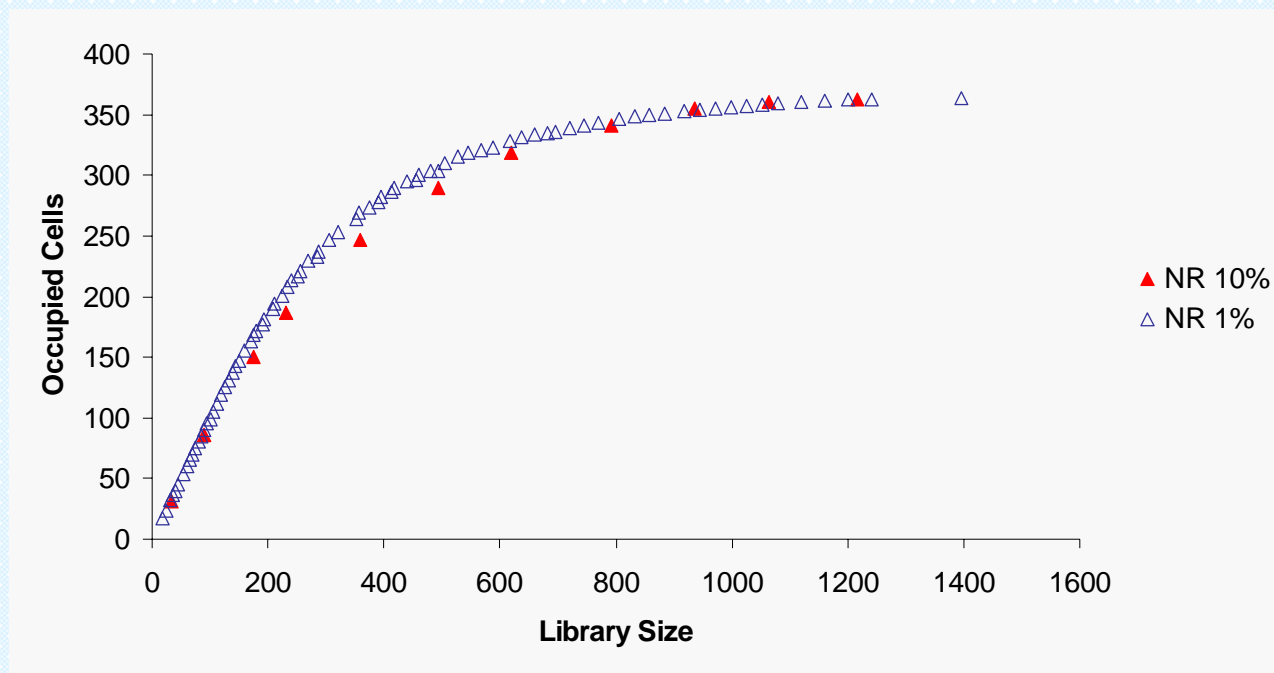


Presenting Solutions

- 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

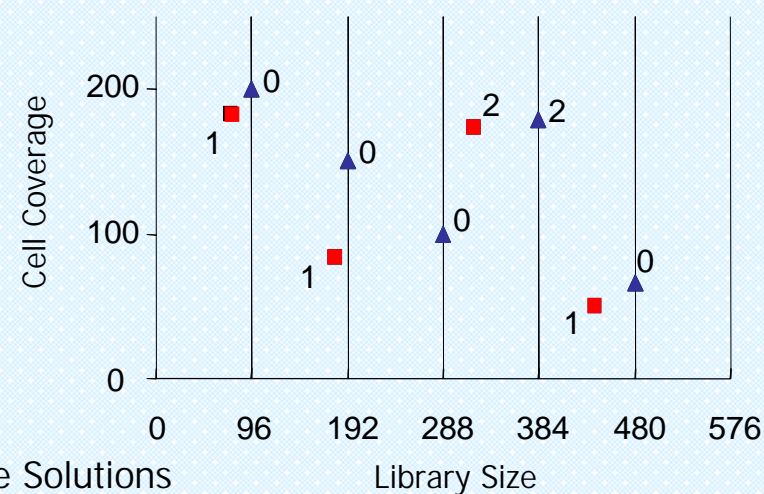
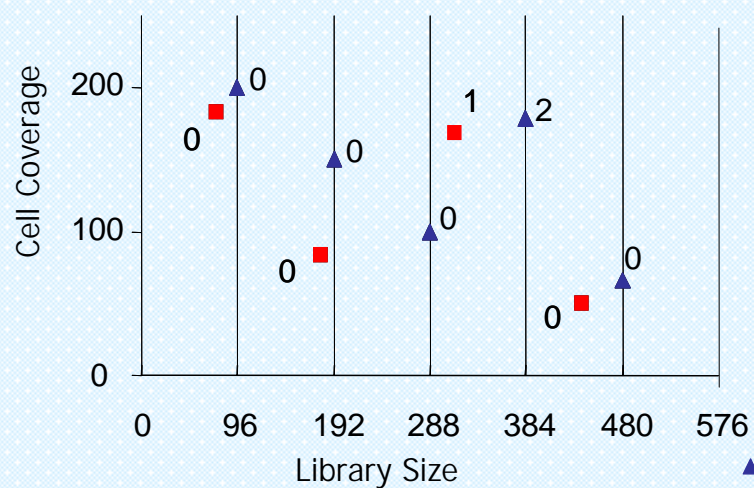


Constraints

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

Plate Coverage

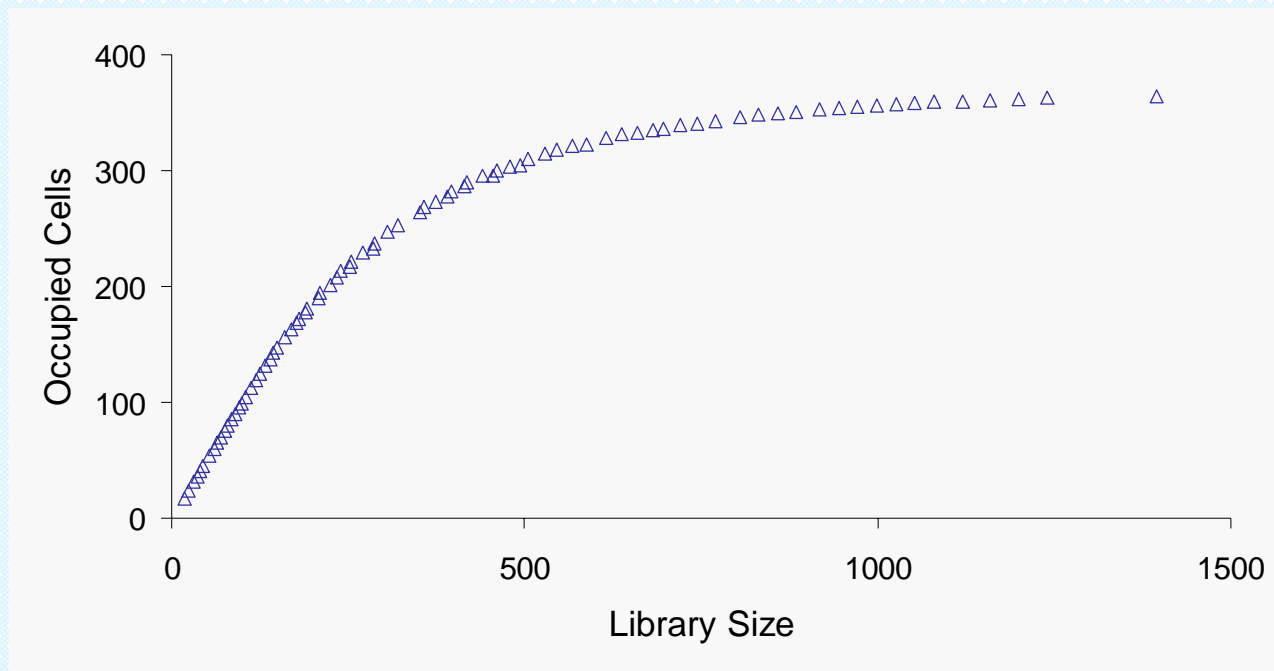
- Libraries are stored on plates one compound per well.
- Constraint.
 - All feasible libraries must be multiple of 96.



▲ Feasible Solutions
■ Infeasible Solutions

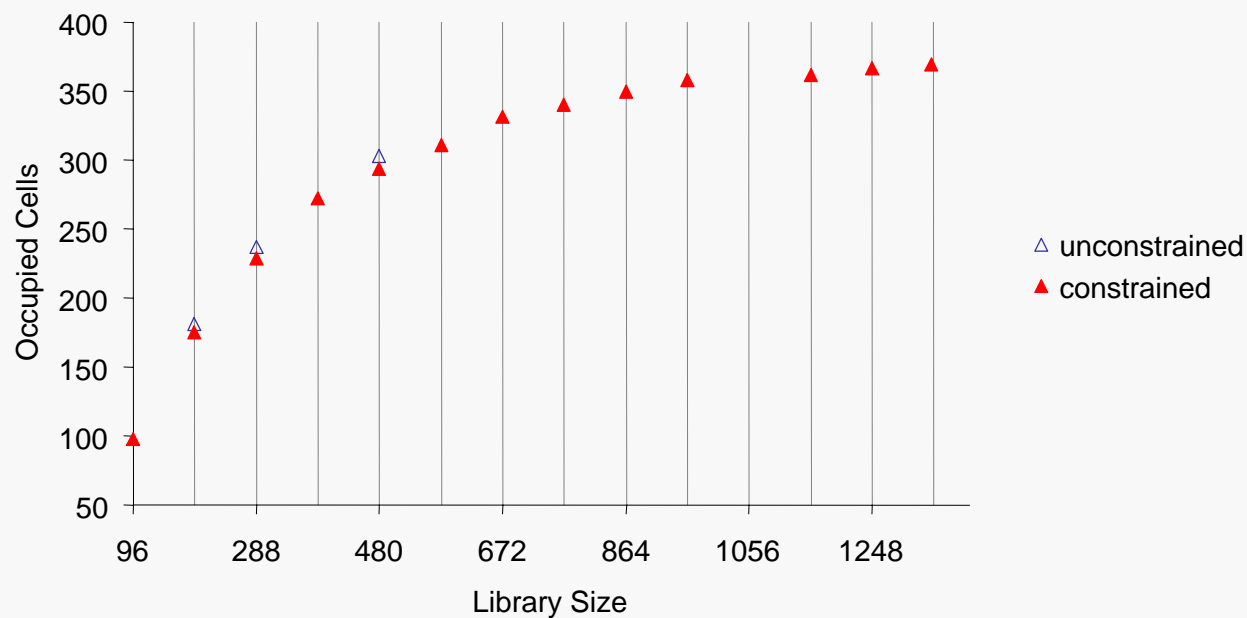


Results - No Constraint



Results - Plate Constraint

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



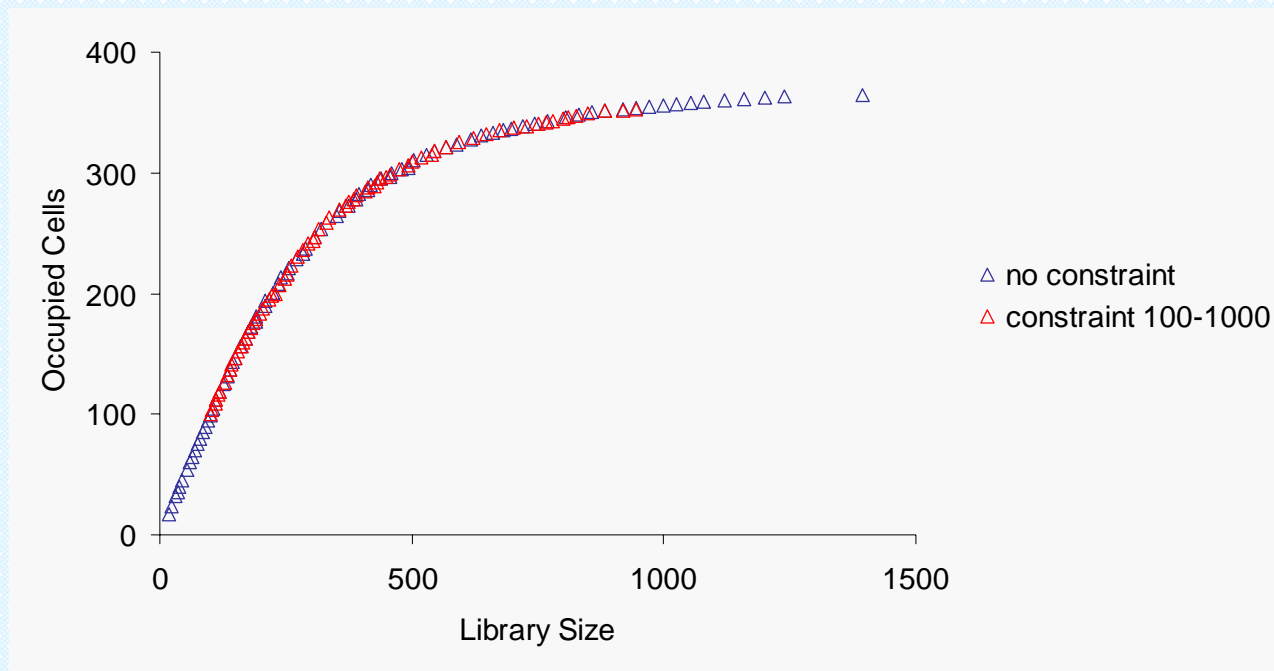


Constraints

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

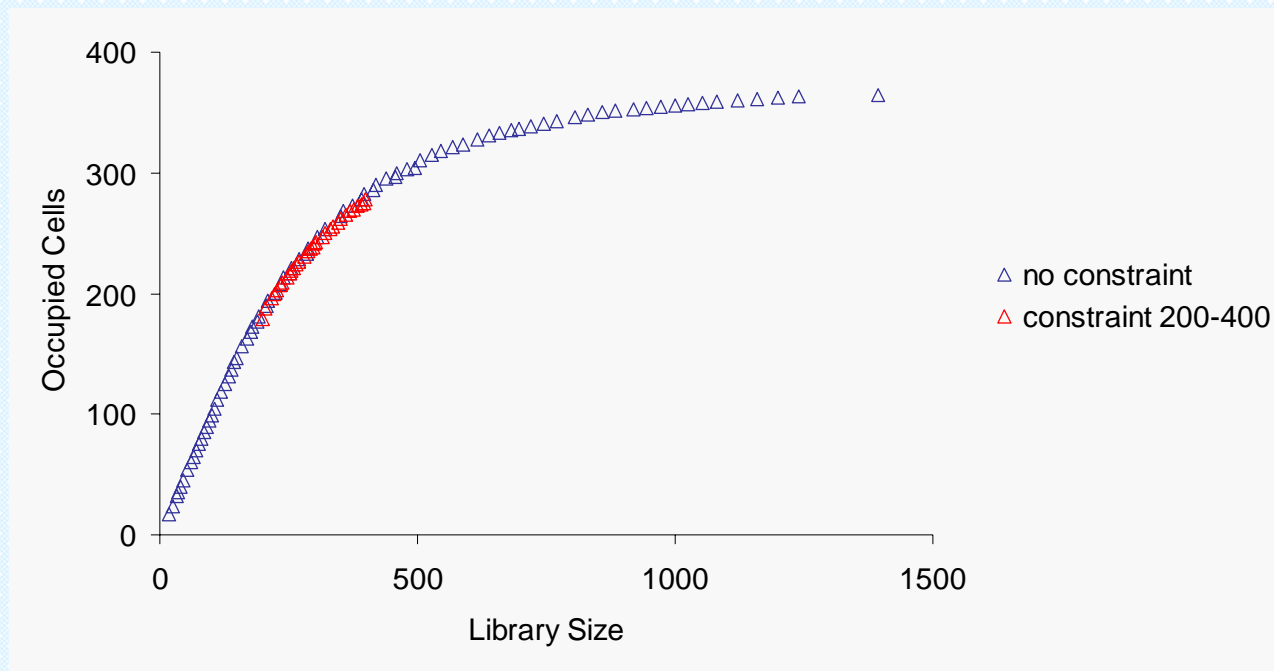


Results - Size Constraint



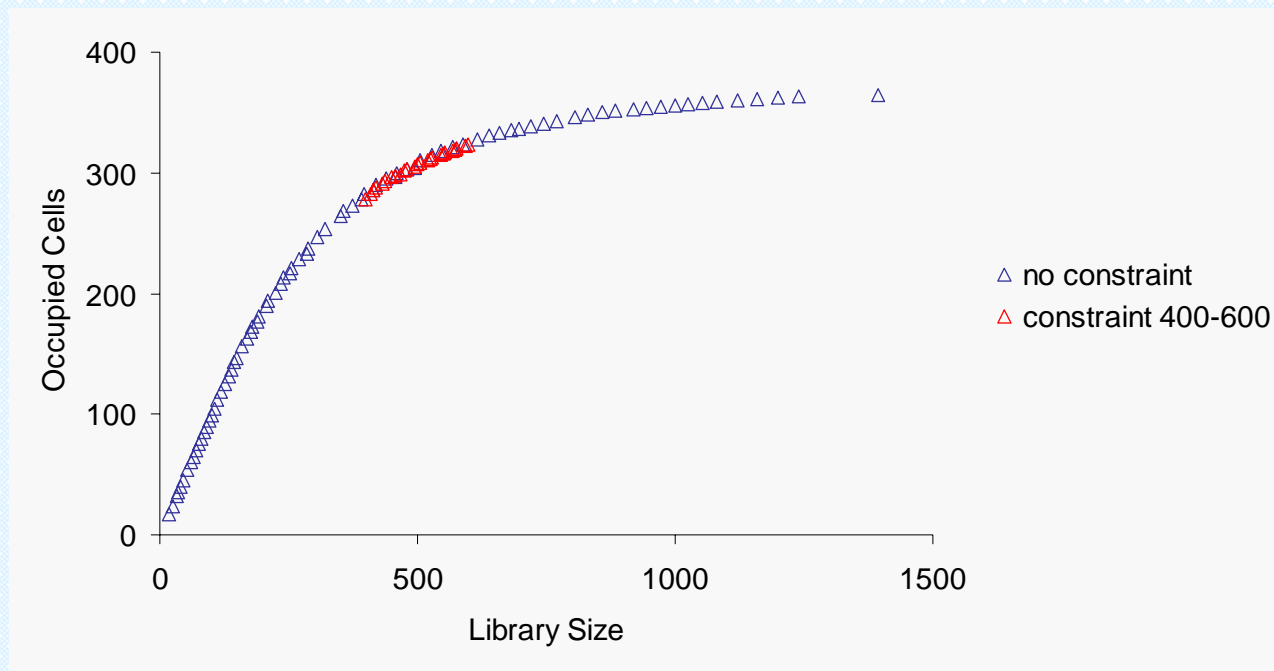


Results - Size Constraint

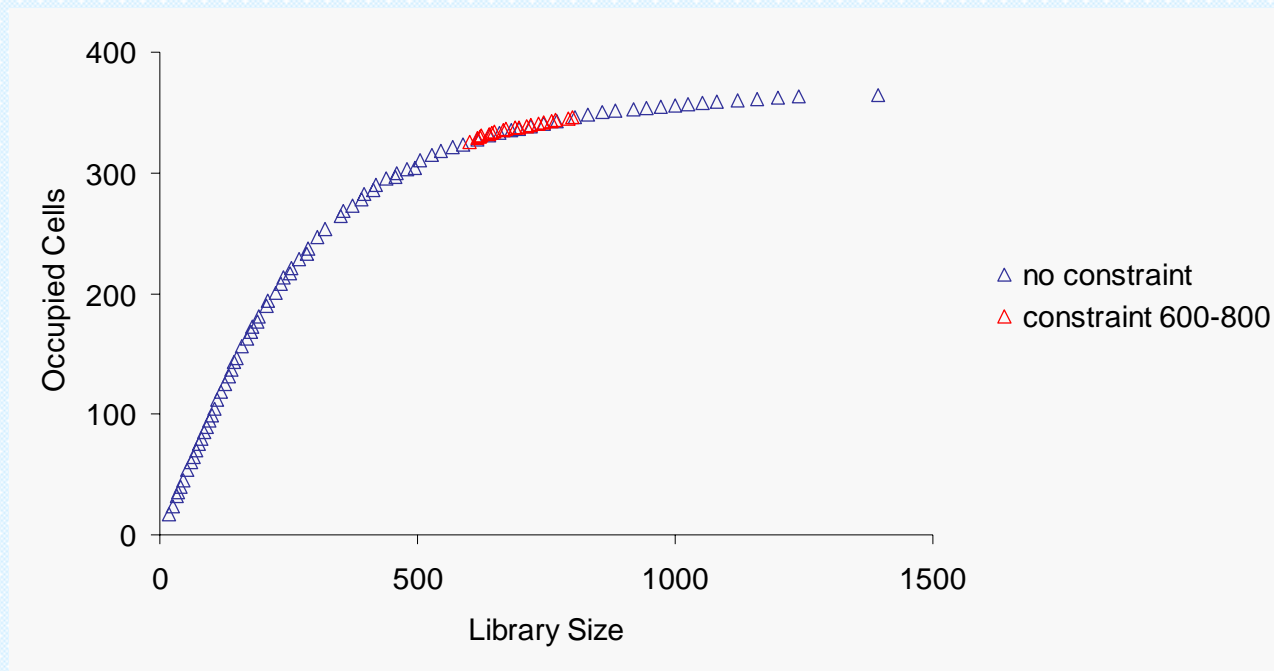




Results - Size Constraint



Results - Size Constraint

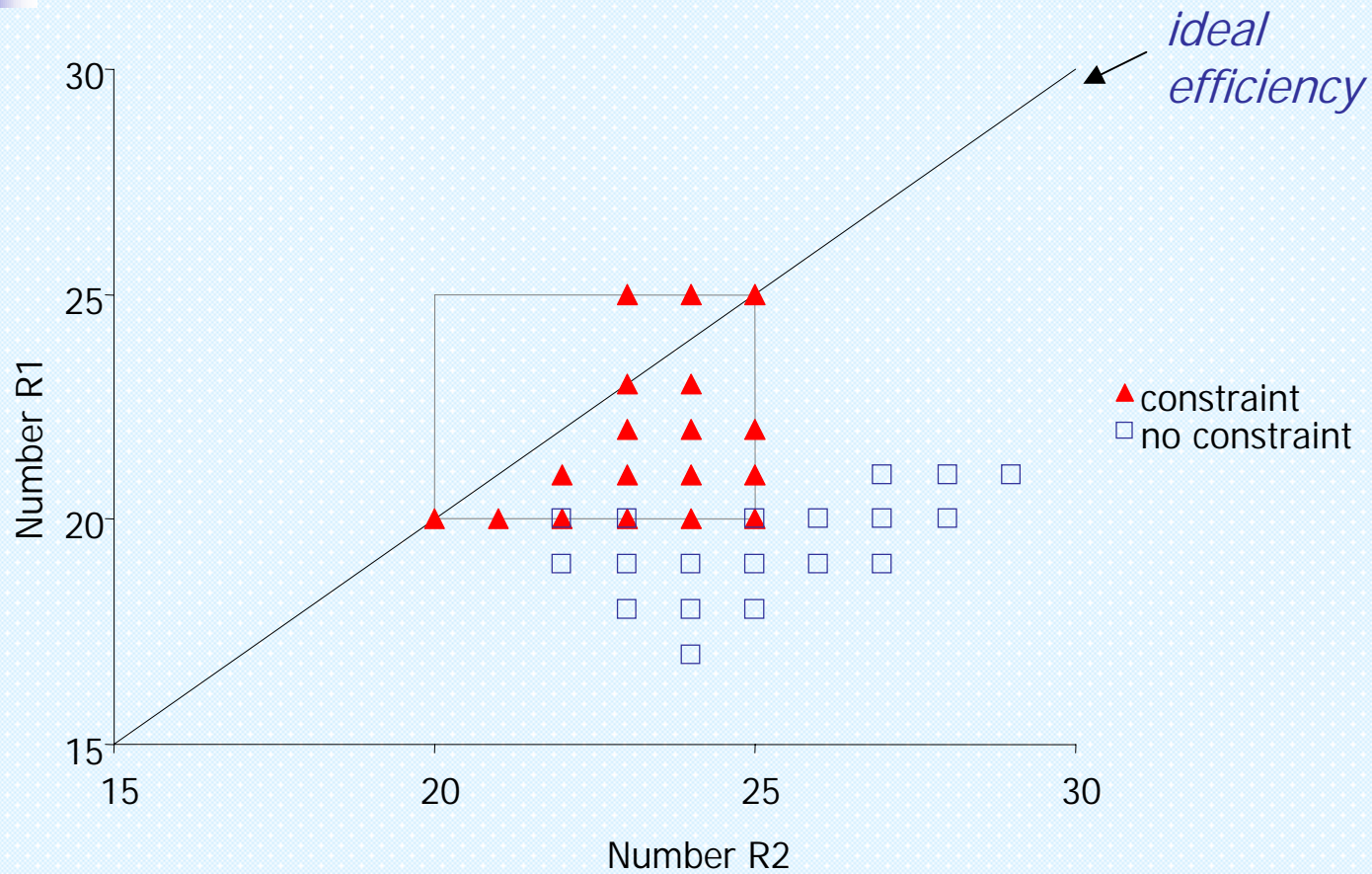




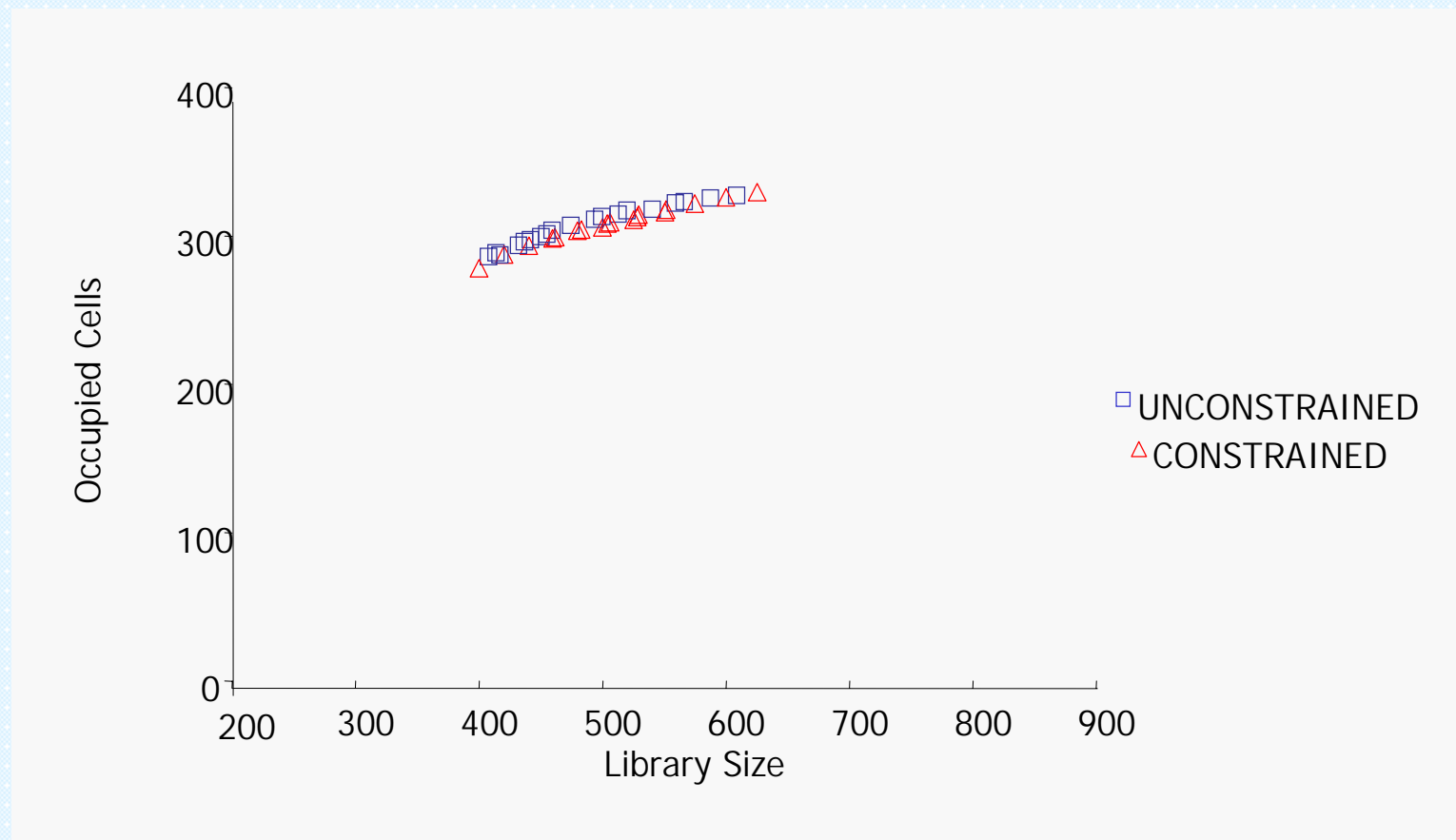
Constraints

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

Combinatorial Efficiency



Combinatorial Efficiency





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