Multi-objective optimisation of building designs

Sandy Brownlee
Research Associate, Loughborough University
Outline

- Me
- Evolutionary Multi-objective Optimisation
- Building design optimisation
- Improvements
  - Constraints
  - Surrogates
  - Inheritance
- Conclusions, questions etc.
Me

- Former RGU PhD, now RA at Loughborough
- TSB / EPSRC funded project – this talk
  - *A Simulation-based Optimisation Tool for the Minimisation of Building Carbon Emission and Water Usage*
  - Civil & Building @ Lboro + consortium of industrial partners
- Other interests…
  - Fitness modelling in EA
  - Deepening understanding of EA & problems
  - Applications
Evolutionary Multi-Objective Optimisation
EMO

- Single objective GA
- Moving to multi-objective
- Constraints
- Performance indicators
- NSGA-II
Single objective GA

1. Generate random population
2. Assign a *fitness* to members of the population
3. Choose the best ones and recombine them to produce *offspring*
4. Mutate the offspring
5. Repeat 1-4 until we’re done
SO GA Example

0 1 1 1 1 0 1
1 0 1 1 1 0 0
0 0 1 1 1 0 1
0 1 1 0 1 0 1
0 0 0 0 0 1 1
1 0 0 0 0 0 0
4

0 1 1 1 1 0 1
1 0 1 1 1 0 0
0 1 1 1 1 0 0
1 0 1 1 1 0 1
0 1 1 1 1 0 0
1 0 1 1 1 0 1
2

0 1 1 1 1 0 0
1 0 1 1 1 1
1 0 1 1 1 0
0 1 1 0 0 1
1 0 1 0 0 0
0 0 0 0 0 1 0
1

0 1 1 1 1 0 0
1 0 1 1 1 0 0
0 1 1 1 1 0 0
1 0 1 1 1 1 1
0 1 1 1 1 0 0
1 0 1 1 1 1 1
3

0 1 1 1 1 0 0
1 0 1 1 1 0 0
0 1 1 1 1 0 0
1 0 1 1 1 1 1
0 1 1 1 1 0 0
1 0 1 1 1 1 1
3

0 1 1 1 1 0 0
1 0 1 1 1 0 0
0 1 1 1 1 0 0
1 0 1 1 1 1 1
0 1 1 1 1 0 0
1 0 1 1 1 1 1
5

0 1 1 1 1 0 0
1 0 1 1 1 0 0
0 1 1 1 1 0 0
1 0 1 1 1 1 1
0 1 1 1 1 0 0
1 0 1 1 1 1 1
4

0 1 1 1 1 0 0
1 0 1 1 1 0 0
0 1 1 1 1 0 0
1 0 1 1 1 1 1
0 1 1 1 1 0 0
1 0 1 1 1 1 1
3

0 1 1 1 1 0 0
1 0 1 1 1 0 0
0 1 1 1 1 0 0
1 0 1 1 1 1 1
0 1 1 1 1 0 0
1 0 1 1 1 1 1
2
Multi-objective

- Multi-objective optimisation…
- In reality, most problems are multi-objective, often with conflicts – e.g. cost vs performance
- How do we define fitness for more than one objective?
- Could just add them together, but how do we weight them?
- Better to find the trade-off and make an informed decision
Dominance

- This time there are two “fitnesses” (objective values) for each solution
- One solution dominates another if it is “better” in both objectives
- Can plot the objectives of population in 2D
- Set of non-dominated solutions is the Pareto front
Constraints

- Some solutions might be fit, but are otherwise unwanted
  - Building with no ventilation is cheap and low-energy, but not very comfortable!
  - Examples: max hours over 28°C, min lighting, compliance with building regs
- Penalty functions, algorithm enhancements
  - Whole area of research in itself
  - Can be included in the concept of dominance
- Constraints can be hard to satisfy
Comparing performance

- Hard to compare fronts
- What are we measuring?
  - Closeness to “true” Pareto front
  - Spread along the front
  - Extents of front
- Several measures; hypervolume used here
Hypervolume

- The area / volume between the PF and a nadir point (the global minimum)
- General measure; includes extent, spread and optimality of PF
- Prefers convex regions of PF
- Expensive if many objectives
A popular GA for MO optimisation

Selection biases search towards:
- Feasible solutions
- Non-dominated solutions (low rank)
- Non-crowded solutions

Basis for the experiments here
Building design optimisation
Building Designs

- Broad concepts
- 3 example problems, with results
- Variable sensitivity
Building design optimisation

- Buildings are complex!
- Many variables
  - Dimensions, materials, layout, systems (heat / light etc), control configuration
- Many objectives / constraints
  - Energy use, Construction cost, Comfort
  - Compliance
- Highly suitable for EA
Building design optimisation

- Different design stages
  - Conceptual
  - Scheme
  - Detailed
- Change at concept stage can be big
  - But also dependent on getting things right later
- Project blurring lines between stages; optimise across stages (e.g. orientation, envelope, controls) but more to be done
Building design optimisation

Evolutionary algorithm

Simulation (energy, cost modelling, comfort prediction…)

Fitness

Optimal building(s)
Example 1: Cellular Windows

- Optimise glazing for an atrium in a building
- Switch on glazing and shades in 120 cells
  - 240 bits encoding
- Minimise energy use, or energy and cost
  - Energy for lighting, heating and cooling
- Constraints: number or aspect ratio of “windows” (mutually neighbouring cells)
Example 1: Cellular Windows

Window 1  Window 2  Window 3
Single Objective

- With “number” constraint, glazing falls in central area
  - Where the light sensors are located
- With aspect ratio constraint, glazing tends to be spread out, still usually 3 windows
  - Better coverage of facade
Multi-objective

- Trade-off for energy vs cost
  - Simple linear cost per glazed cells & shades
- Larger window still tends to centre
- Hard to meet constraints
- Seeding the population helps
Example 2: Office block

- Small 5 zone office; a single floor of a larger building
- Variables
  - Orientation, glazing area, type, wall/floor types, HVAC set points and times
- Objectives
  - Energy use, cap cost
- Constraints
  - Thermal comfort, air quality (CO₂ levels)
Results

- Example building with glazing altered
Results
Example 3: Risk of mould growth

- Optimise HVAC config to identify high risk conditions
- Risk related to long, warm, damp periods
- Hospital ward in Kuala Lumpur
Variable sensitivity

- Aid to decision making
  - What does sensitivity tell us about the problem?
- Observe which variables impact the most
  - Can we ignore some of them to simplify the search?
  - What do we learn about the underlying problem? Can this aid decision making?
- Some are fixed, some vary, both have an impact
Variable sensitivity

![Variable sensitivity graph]

- **Cost (€)**: Y-axis
- **Energy (kWh)**: X-axis
- Infeasible: +
- Feasible: ×
- GlobalPF: *

The graph illustrates the relationship between cost and energy usage, highlighting the feasible and infeasible regions.
Variable Sensitivity

- Jump to IES EP comparison spread sheet
- Energy vs cost for different models
  - Ceiling construction type for IES
  - North glazing area for E+
  - Other glazing areas less important
Algorithm Improvements
Improvements

- Constraint handling
- Fitness inheritance
- Surrogate model
- Experiments / results
Constraint Handling

- Constraints can be hard to satisfy, and can limit the extent of the trade-off found
- Relaxation – ignore constraints to start with
- Normalise / weighting
  - Constraints weighted equally, or with a bias to meeting harder constraints first
- Include infeasibles in population
  - Allow some infeasible solutions in population
  - Either keep “least infeasible” or “fittest” infeasibles
A problem!

- Typical EA needs thousands of simulations
- Building energy simulation takes 1-2 minutes for example problems
- Larger building or more detailed sim takes longer; also larger search space
Possible solutions

- Reduce model complexity
- Reduce weather data extent
- Parallel execution / caching solutions
- Fitness inheritance
- Surrogate
Fitness Inheritance

- Based on the idea that two “similar” solutions will have similar fitnesses / objective values
- After crossover, guess that offspring’s fitness is somewhere between that of parents
- Only inherit sometimes – typically about 50%
- Can weight towards one parent
- How do we deal with constraints?
  - Predict values for each and keep inequality
  - Not ideal!
## Fitness Inheritance

<table>
<thead>
<tr>
<th>Individual</th>
<th>Energy Use kWh</th>
<th>Cost £</th>
<th>Overheating hours (max 30)</th>
<th>Max CO2 conc. (max 1500)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent A</td>
<td>54200</td>
<td>370000</td>
<td>40</td>
<td>430</td>
</tr>
<tr>
<td>Offspring</td>
<td>57200</td>
<td>365000</td>
<td>25</td>
<td>330</td>
</tr>
<tr>
<td>Parent B</td>
<td>60200</td>
<td>360000</td>
<td>10</td>
<td>230</td>
</tr>
</tbody>
</table>
Surrogate Model

- Train a model of the fitness function
- Use the model in place of the FF

Evolutionary algorithm

Simulation (energy, cost modelling, comfort prediction…)

Optimal building(s)
Surrogate Model

Plain EA

1. Generate random population
2. Assign a *fitness* to members of the population
3. Choose the best ones and recombine them to produce *offspring*
4. Mutate the offspring
5. Repeat 1-4 until we’re done
Surrogate Model

1. Generate random population
2. Assign a *fitness* to members of the population
3. Choose the best ones and recombine them to produce *too many offspring*
4. Mutate the offspring
5. Use surrogate to filter out promising offspring
6. Repeat 1-5 until we’re done

EA with surrogate
Surrogate Model

- Limited work done with mixture of continuous and discrete variables, and with constraints
- Approach to constraints same as for FI
  - i.e. predict value then do cut-off
- Using a radial basis function network (RBFN)
- Initially tried a single network
  - Had to retrain whole network if part of it poor
  - Now one network per objective or constraint
RBFN

- Feed-forward network
- Input layer: problem vars
- Hidden layer:
  - radial basis functions
  - output similarity to centre
- Output layer:
  - linear weighted sum per objective / constraint
- Distances
  - Euclidian (cont), Manhattan (int), Hamming (bits)
Experiment

- The 5 zone building problem (energy/cost)
- Run each algorithm config, limit to 5000 evals
- NSGA-II is base-case; calc:
  - mean hypervolume for final sets
  - evals to reach hypervolume target (i.e. the HV reached by NSGA-II in 5000 evals)
  - final archive size (the detail in the trade-off) – this is linked to population diversity
Results

- Speedup & larger PF size
- Constraints need relaxed in some way

<table>
<thead>
<tr>
<th></th>
<th>NSGA-II</th>
<th>+FI</th>
<th>+FI +infeas</th>
<th>+surr +infeas</th>
<th>+Fl+surr +infeas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evals to mean HV</td>
<td>4191</td>
<td>4080</td>
<td>3015</td>
<td>3998</td>
<td>3662</td>
</tr>
<tr>
<td>Success Rate</td>
<td>50</td>
<td>60</td>
<td>70</td>
<td>90</td>
<td>100</td>
</tr>
<tr>
<td>HV after 5000 evals</td>
<td>0.214</td>
<td>0.216</td>
<td>0.224</td>
<td>0.218</td>
<td>0.220</td>
</tr>
<tr>
<td>Final archive size</td>
<td>23</td>
<td>34</td>
<td>34</td>
<td>23</td>
<td>26</td>
</tr>
</tbody>
</table>
Conclusions

- Optimisation (particularly with EA) a growing area in building design community
- Currently maturing
- Room for improvement
  - Move to concept stage (form / shape)
  - Simulation time a big issue
Questions