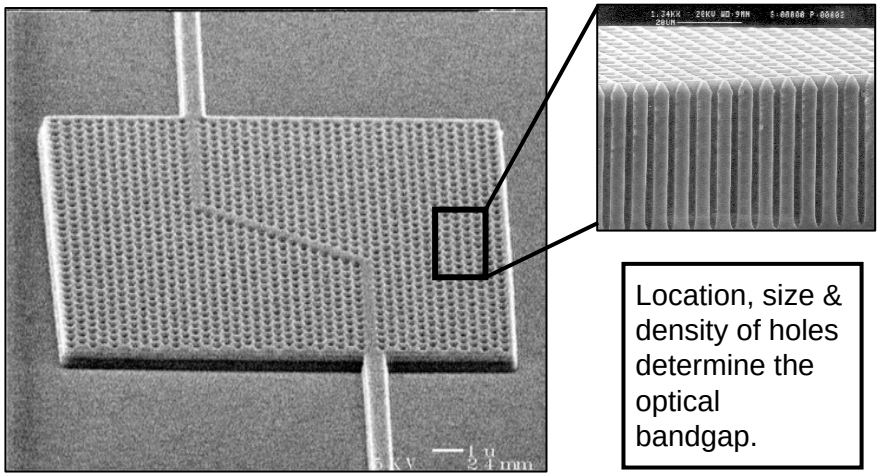
Simon Cox (University of Southampton)
Engineering Workflows (= 'Scripts')

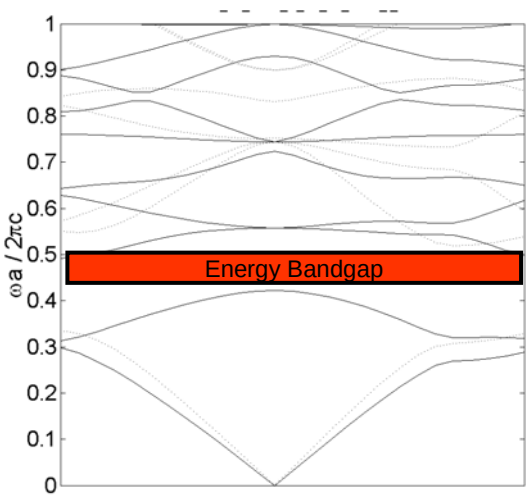
- Scenario: Design optimisation
  - Model device, discretize, solve, postprocess, optimise
- Scripting approach
  - Flexibility & High-level functionality
  - Quick application development
  - Extend user's existing PSE e.g. Matlab, Python, ...
  - … is our execution/ enactment engine too
- Favourite things to do
  - Create, Retrieve, Cut 'n' Shut (*Re-use anything and everything*)
  - Configure, Execute, Monitor (*Bring Grid to user*)
  - Share, Steer, Dynamically modify (*Semantic support*)

GEODISE

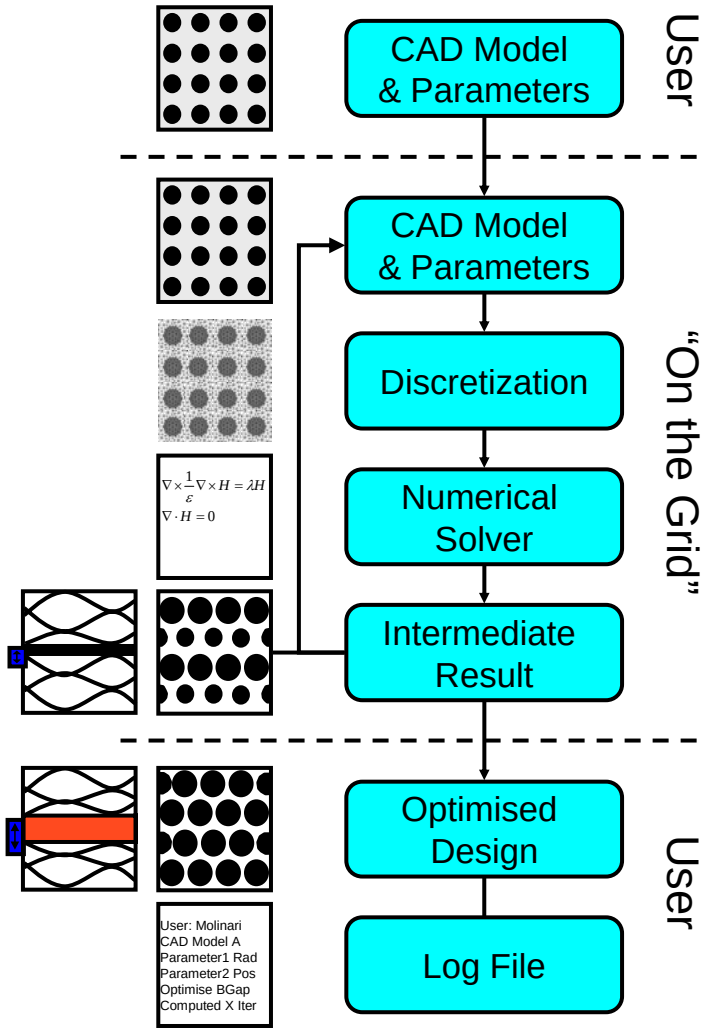
Photonic Crystal - Optimisation



Location, size & density of holes determine the optical bandgap.

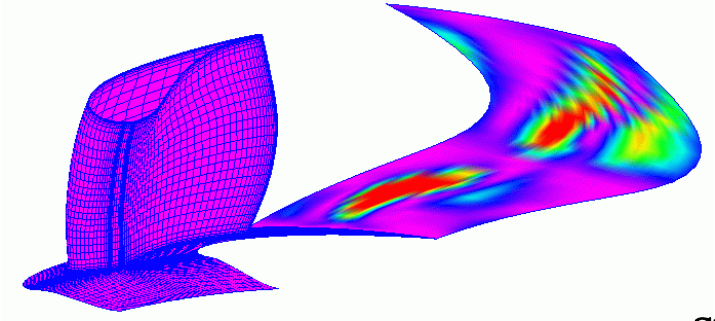


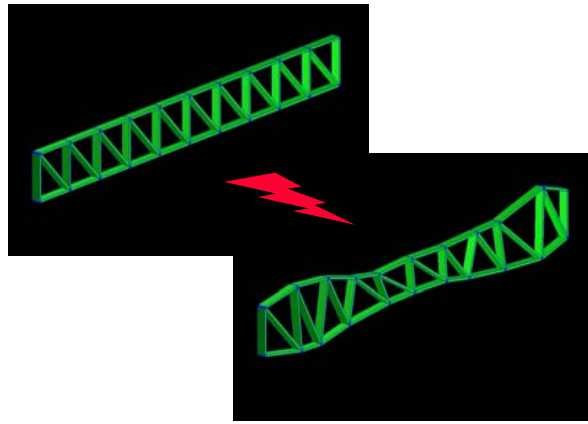
The aim of this design optimisation process is to find a configuration which maximises this bandgap and minimizes energy loss.

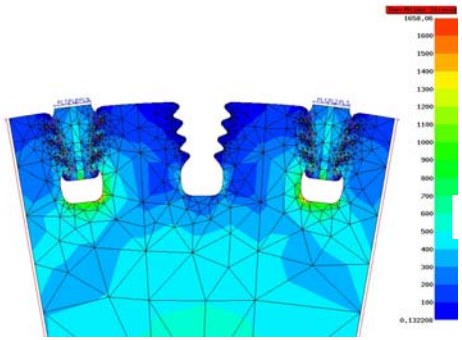


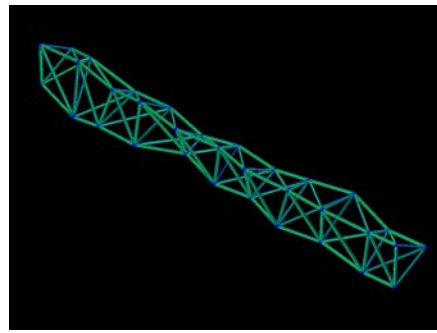
GEODISE

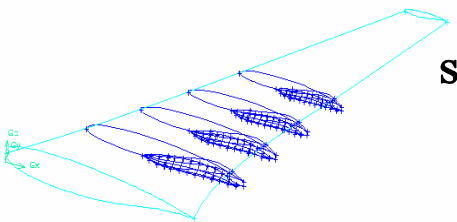
Design

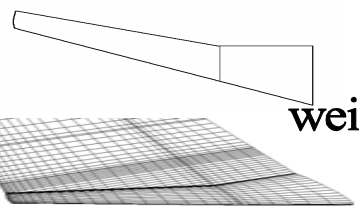


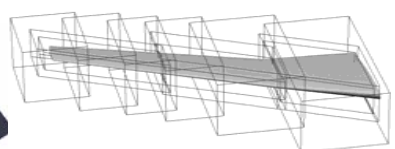


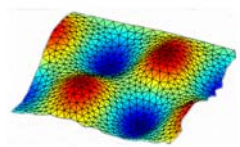


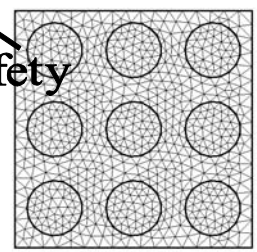


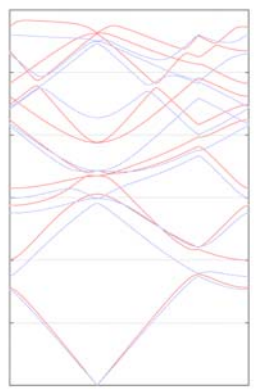


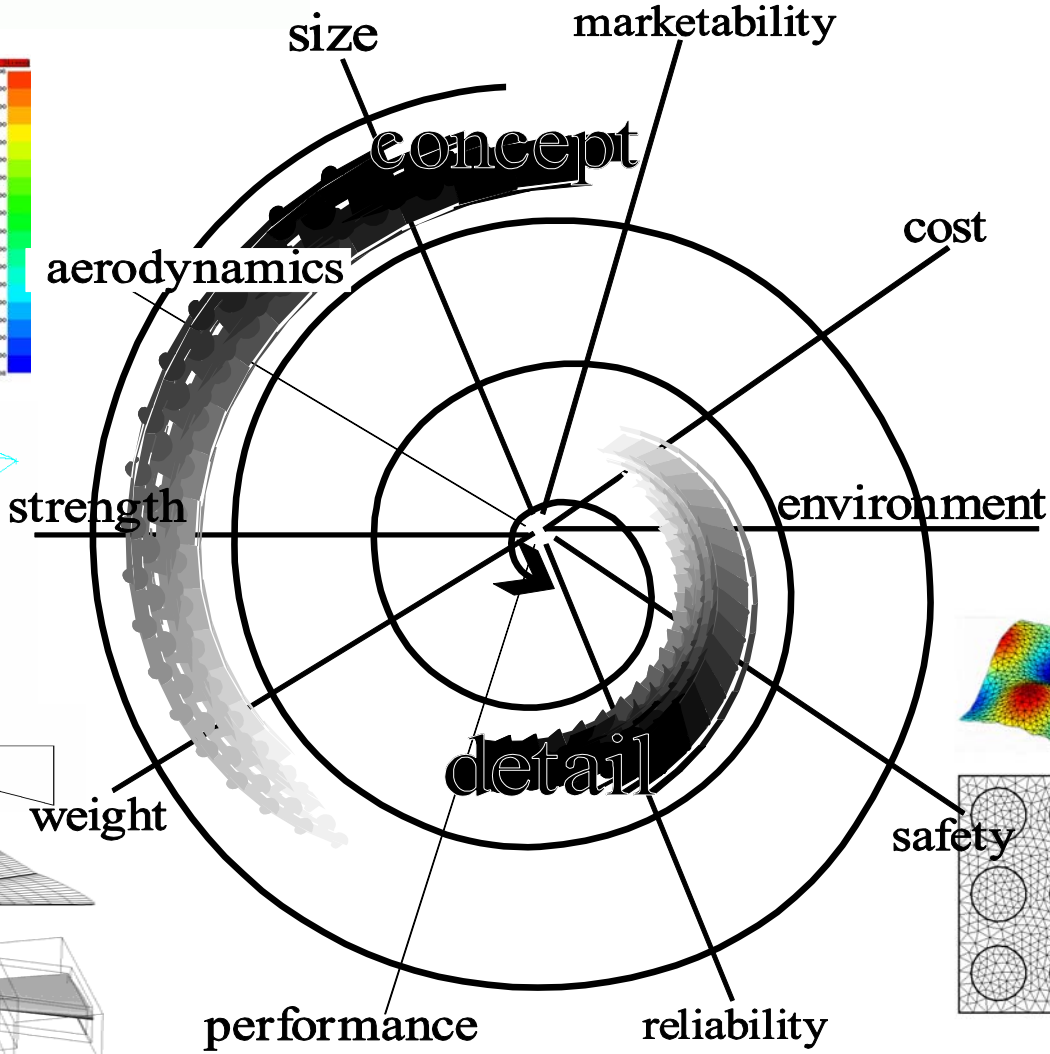












Grid-Enabled Matlab Scripting Environment

- Motivations:
  - Flexible, transparent access to computational resources
  - Easy to use for engineers (and in widespread use)
- Our Approach
  - Matlab chosen as the hosting environment
    - Extend the user's existing PSE
    - High-level functionality
    - Quick application development
    - … is our execution/ enactment engine too
  - Computational resources exposed in the form of Matlab functions
    - Job submission to Globus server using Java Cog
    - Job submission to Condor pool via Web services interface
  - Integration of CAD, Mesh generation, and Fluent solver via the use of intermediate data format, often standard-based, or package-neutral
  - Hybrid search strategies to make the best use of different search methods
- Can also use Python, Jython, etc.