Design Space Exploration
Flynn’s Taxonomy

- Classification of computer architectures made in 1966 by Michael Flynn (IBM)
- Based on whether instruction and data streams are parallel
- SISD – serial processor
- SIMD – array or vector processor
- MISD – for fault tolerance, systolic array
- MIMD – multicore or distributed processor
Options include
- Algorithm (most important)
- Parallelism
- Precision
- Interface
- Customisation

Within each are other options and so the actual design space is extremely large.

Key to making good designs is to have good judgment regarding the tradeoffs:
- These may be different depending on what you need to optimise
- Can be estimated using back-of-envelope techniques and reduced implementations
- Finding suitable input data to characterise your application is also a big issue
For competing factors such as speed and area efficiency (1/area)

Pareto Frontier separates infeasible from feasible designs

We want to be as close to the optimal as possible
Introduced some important principles
- Moore’s Law (tells us how IC area scales)
- Dennard’s Law (tells us how IC technology scales)
- Amdahl’s Law (tells us how to estimate speedup for parallel processing)

FPGA designs have followed technology

Design space is large (curse of dimensionality) so we need to be selective and tried to be close to Pareto Frontier

Exploration must be done right to avoid having to redesign system


Review Exercises

› Explain in your own words:
  - Moore’s Law (tells us how IC area scales)
  - Dennard’s Law (tells us how IC technology scales)
  - Amdahl’s Law

› A problem has a section of non-parallelisable code which takes 100 s to execute, and the rest of the code is parallelisable and takes 1 hour to process. If we are given the task of designing an FPGA accelerator to replace the CPU and wish to achieve a speedup of 100, what should the speedup of the FPGA accelerator core be? What if it takes a day to process?