Establishing Confidence and Understanding Uncertainty in Real-Time Systems

Iain Bate, David Griffin, Benjamin Lesage

Dept of Computer Science

University of York

iain.bate@york.ac.uk

Overview of the Seminar

- **Introducing the baseline data that has been captured**
- **Stage 1 – Black Box - "Basic" statistical analysis**
- **Stage 2 – Grey Box - Combining machine learning and statistics to give deeper knowledge**
- **Categorising what we found**
  - Known knowns
  - Known unknowns
  - Unknown knowns
  - Unknowns unknowns

TACO Framework Recap

- **With Rolls-Royce we have developed a search-based framework for large-scale testing of industrial software**
  - Simulated annealing algorithm creates test vectors applied to the Software Under Test (SUT)
- **Aim is to provide confidence of execution time data through coverage**
- **BCHLr fitness function designed to maximise localized path coverage**
  - BCHLr from S. Law, I. Bate, Achieving Appropriate Test Coverage for Reliable Measurement-Based Timing Analysis, Euromicro Conference on Real-Time Systems, 2016.
- **BCHLr shown to be more effective than other fitness function**

Data from TACO

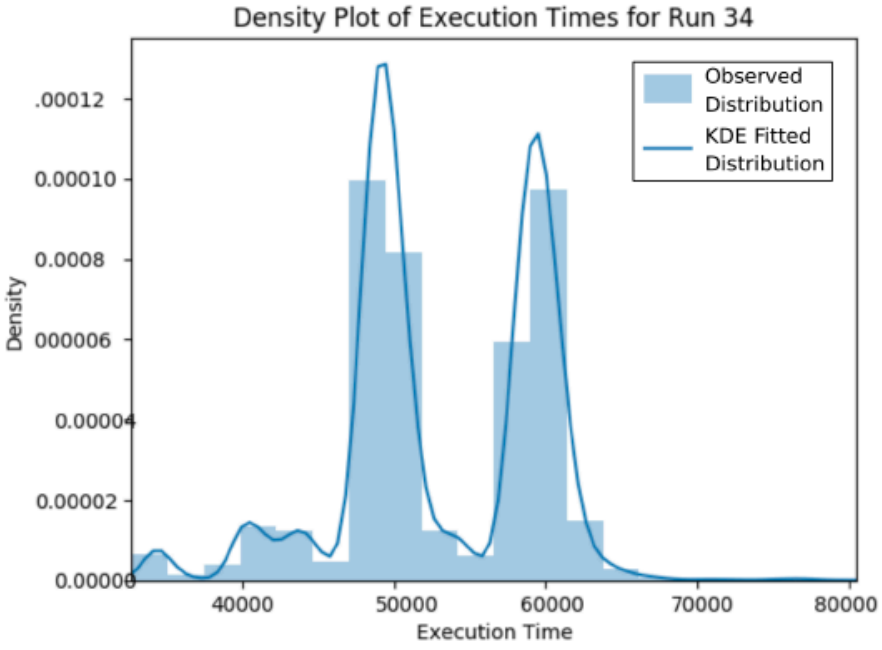
- **The result is a vast amount of data from a Full-Authority Digital Engine Controller**
- **Close to 300 tasks analysed**
- **Platform is PI3B+**
- **Block level**
  - #iterations
  - #cycle count
- **Program level**
  - Sequence of iPoints covered
  - iPoint is instrumentation point at the start of each basic block
  - End-to-end execution time

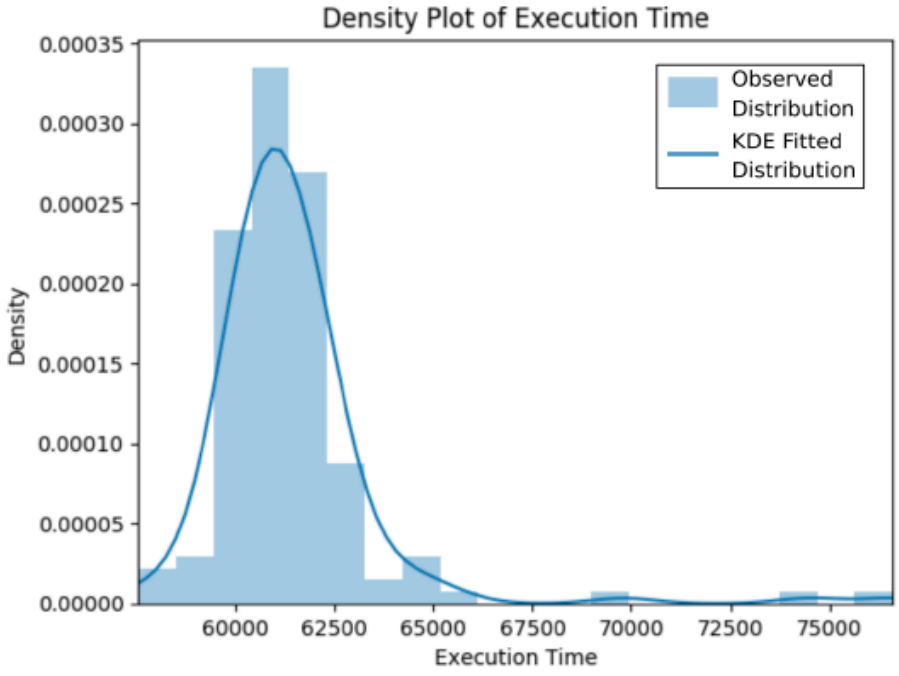
Objectives

- **To understand the sufficiency of testing**
- **Sufficiency judged by**
  - Understanding the structure of the software
  - To enable accurate hybrid analysis
  - To have confidence in High WaterMarks (HWM)
- **Three stages**
  - Pure black-box statistics
  - Machine learning – depending on how you define this
  - Statistical analysis of learned information

Stage 1 – Black Box

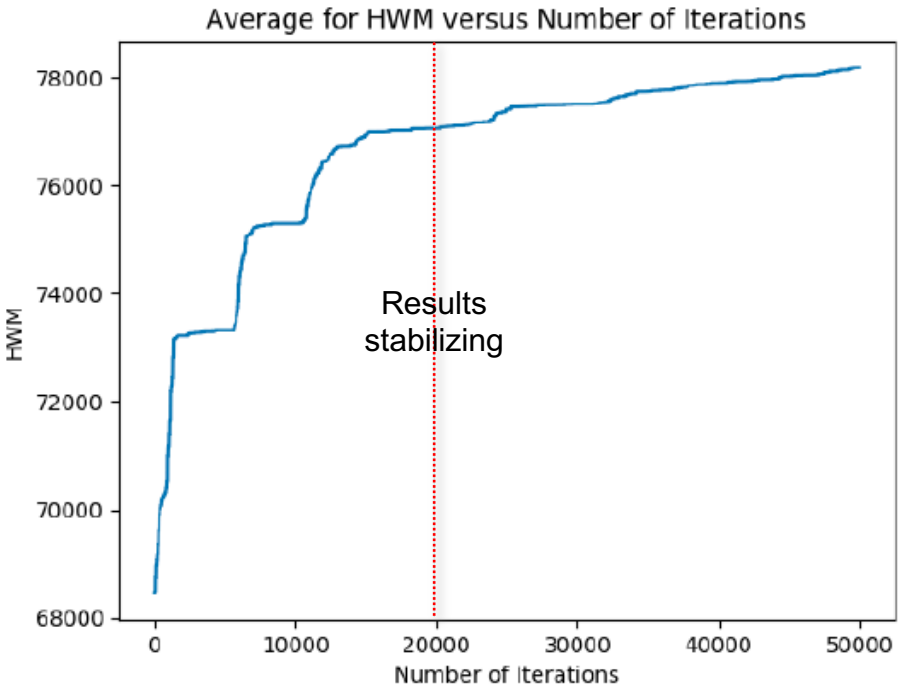
- **LHS – Understanding the SUT via the spread of execution time**
- **RHS – Checking validity of test set up via the degree of measurement noise across samples**

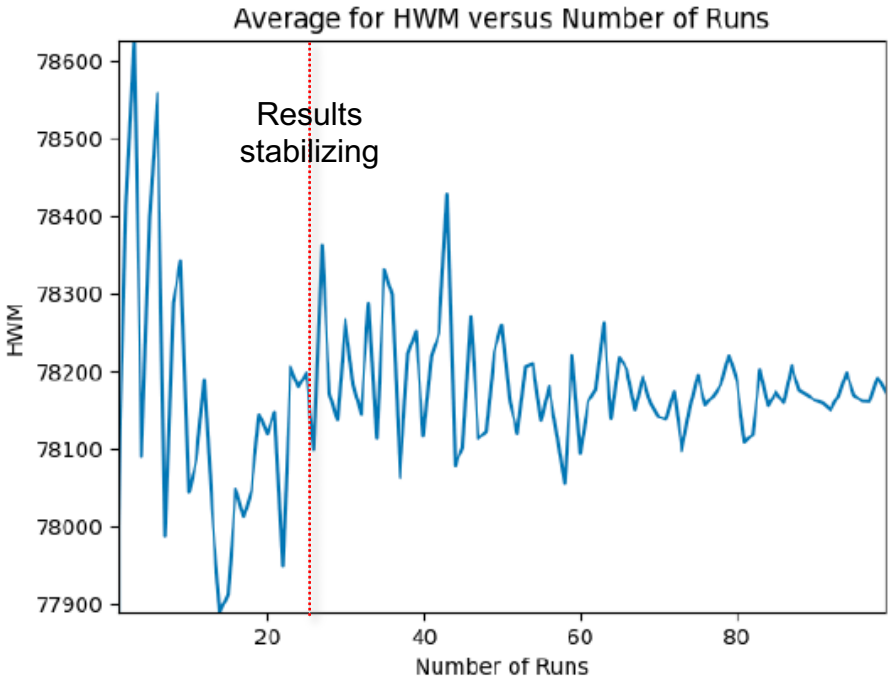




Stage 1 – Black Box

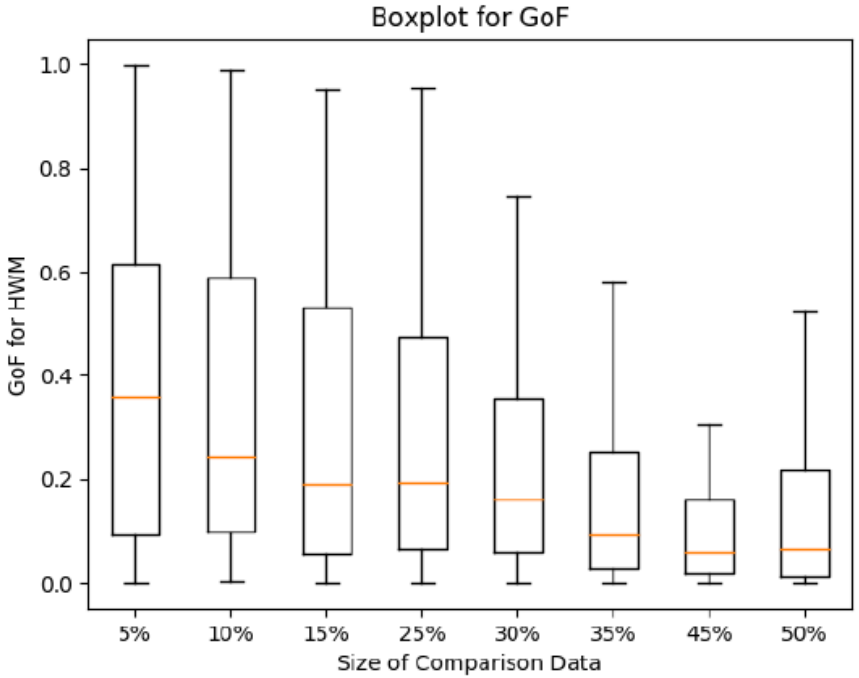
- **What is the best balance between #runs and #iterations?**
  - LHS - #iterations is the number of test vectors each time
  - RHS - #runs is the number of times search function executed
- **25 runs with 20k iterations seemed appropriate for this SUT**

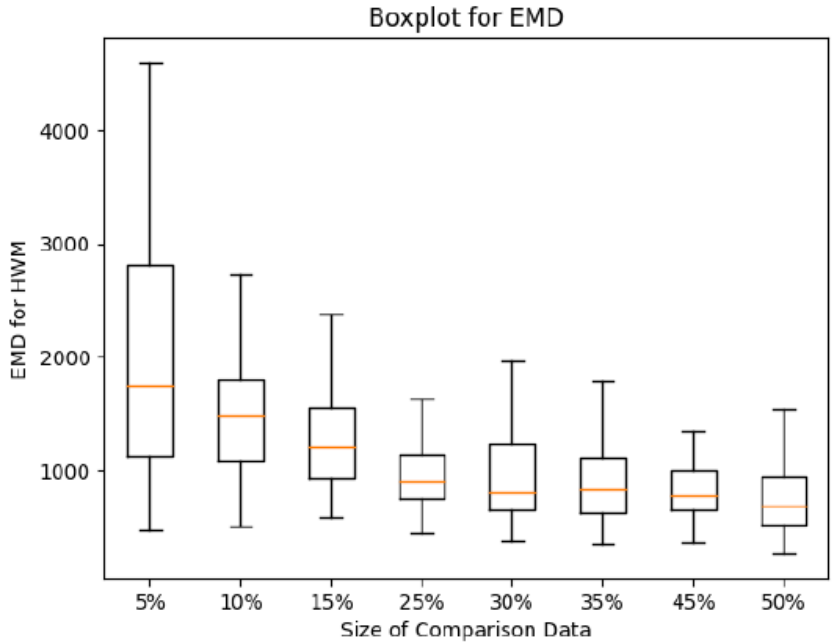




Stage 1 – Black Box

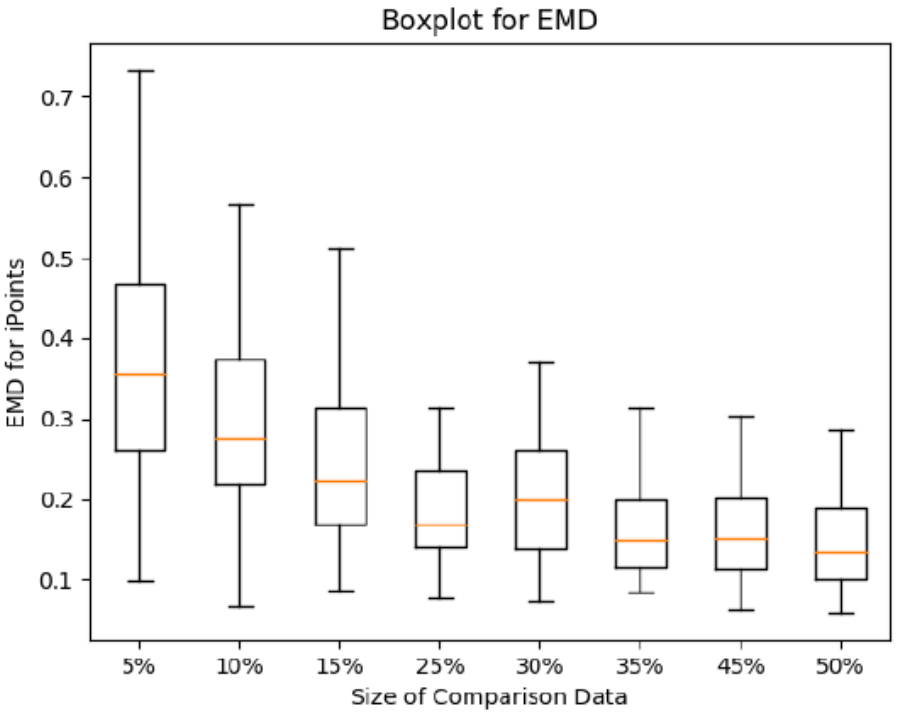
- **Used 100 trials and 50k iterations to assess convergence**
- **Performed cross fold validation 20 times to understand how much variability we would see**
- **10% means 10% of the trials compared with the other 90%**
- **Suggests Earth Movers Distance gave more stable / useful results**
  - E.g. Goodness of Fit goes up between 45% and 50%

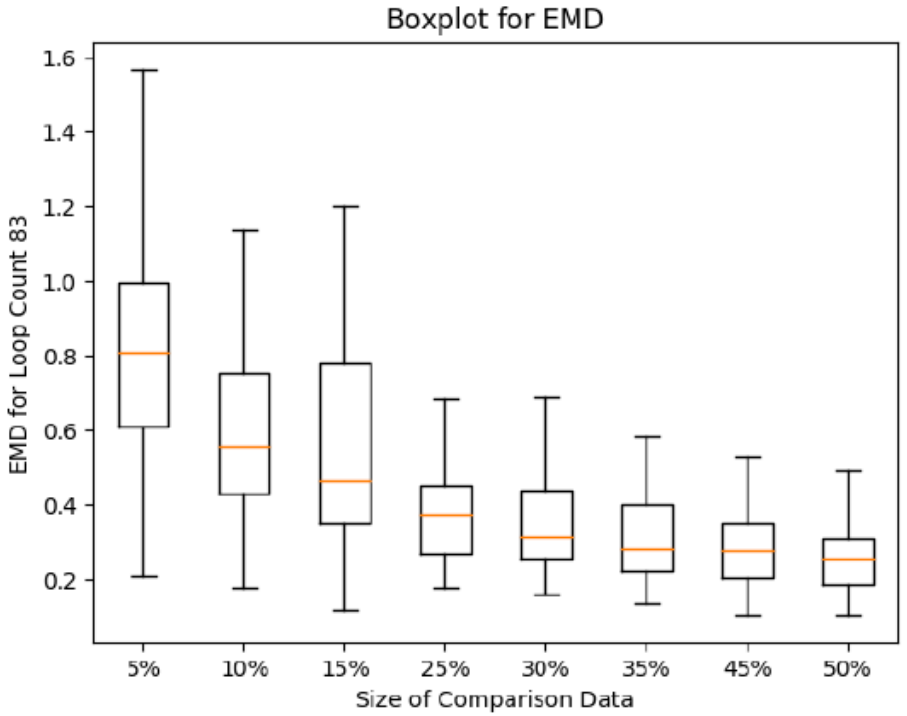




Stage 2 – Grey Box

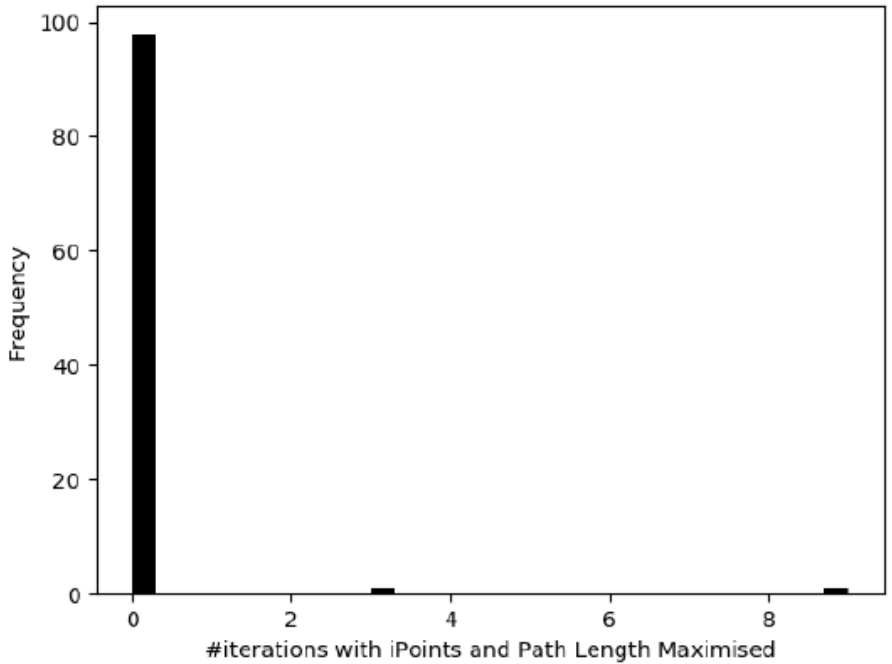
- **Execution time didn't give much confidence of convergence**
  - Positive result as fitness function targets other (significant factors) and tries to avoid convergence
- **We went deeper looking at significant factors, e.g. execution time, loop counts, #iPoints**
- **Analysis suggested results are converging**

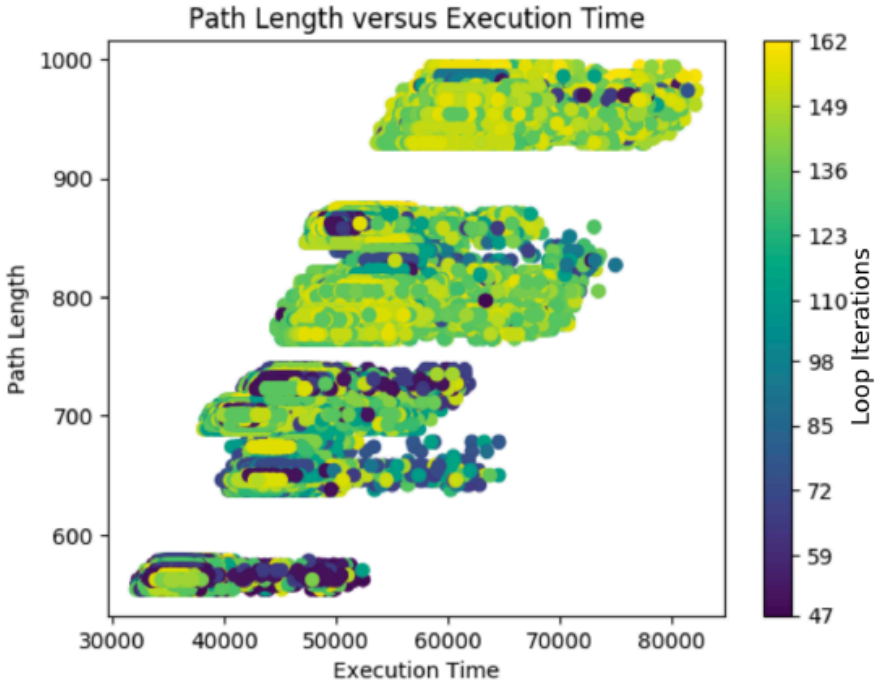




Stage 2 – Grey Box

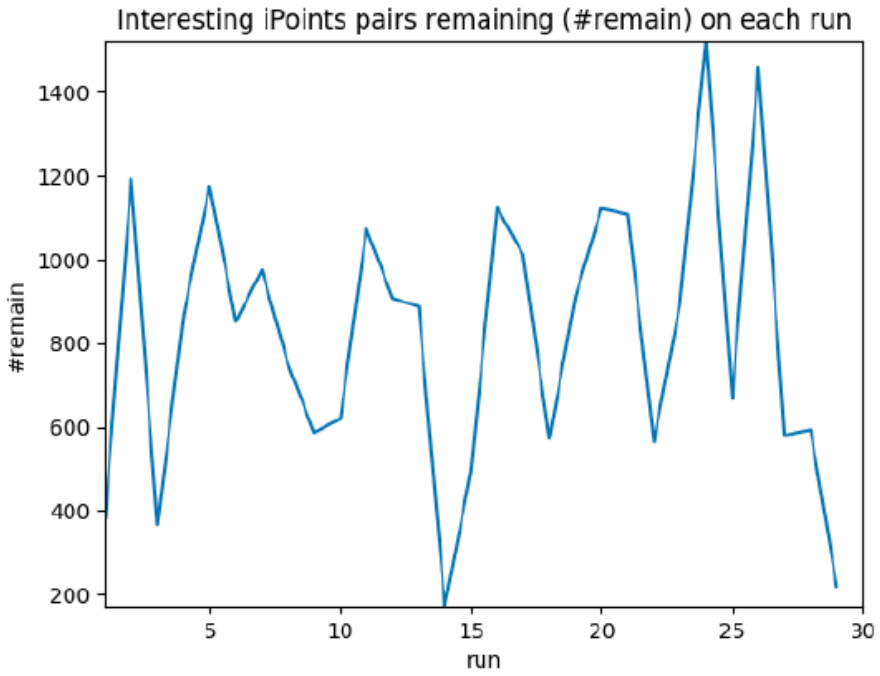
- **LHS - How often an iteration had both #iPoints and path length maximized**
- **RHS –Relationship between path length, execution time and loop iterations**
- **Supports why convergence is challenging**

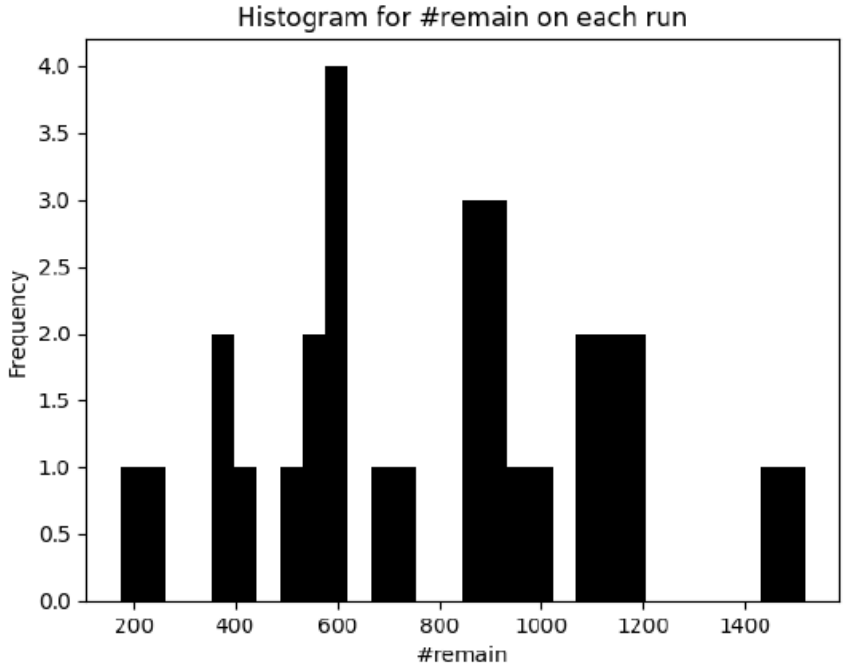




Stage 2 – Grey Box

- **Tried to establish whether infeasible paths exist**
  - i.e. for an iteration, if iPoint X visited then iPoint Y not visited
- **LHS – Took 28 runs before total #remain was zero**
  - #remain is numbers of pairs of iPoints not seen on an individual run
- **RHS – #remain on each run**
  - Shows an individual run tends to have significant infeasible pairs left





What Can be Summarised

- **The work showed**
  - Execution times had not converged much but other significant factors had to a greater degree
  - Understanding confidence and convergence is challenging
  - Even large-scale testing with the state of the art testing left significant questions
  - Applying statistical tests in a black box manner told us very little
  - Going deeper learning details about significant factors and applying statistics was more useful
  - A long way away from fully understanding confidences and uncertainties

What Can be Summarised

Can be thought of in terms Rumsfeld's / Nasa's popularised

- **Known-knowns**
  - Only what we have observed, e.g. maximum observed loop bounds
- **Known-unknowns**
  - What we know that we probably don't know, e.g. the actual maximum loop bounds
- **Unknown-knowns**
  - We haven't seen something but don't know if it could happen, e.g. a single iteration maximizing all loop bounds
- **Unknown-unknowns**
  - Events judged implausible, e.g. an error changing the loop bounds

Summary and Future Directions

- **Determining WCETs for predictable software on a less complex platform shown to be difficult**
- **Hard to determine when enough testing is done**
  - As Low As Reasonably Practicable (ALARP)
  - Globalement Au Moins Aussi Bon (GAMAB)
- **There are significant unknowns**
- **Multi-core is an even more formidable problem**
- **Other forms of machine learning may help but similar issues will arise**
- **Resilient designs must be the answer but still need to know resilience to what**
- **This type of work would inform where the errors more likely to exist**

Acknowledgements

- **Members of the RTS group**
- **Rolls-Royce for years of support, inspiration and access to real systems**
- **Innovate UK for funding the Hi-Class project**
- **Innovate UK for funding the ATICS project**
- **Huawei for funding the MOCHA project**
- **Members of WashU for stimulating discussions and space to think**