## Overview

- R is a language for statistical computing
  - Dynamic language
- Single Instruction Multiple Data (SIMD)
  - Common instructions across many elements
  - Takes advantage of register width in processor

## Motivation

- R is becoming increasingly more popular in recent years
- R is very slow at computing because of its dynamic properties
- R is purely implemented in sequential code. None of R's functions are thread-safe
- SIMD provides single thread parallelization

## Progress

- Using a Intel(R) Xeon(R) CPU E31245 @ 3.30GHz with
  - Per core: L1 32KB, L2 256KB
  - Shared: L3 8192 KB
- Intel Intrinsics AVX and SSE4.2
- Parallelized basic math functions for integer type
  - 4 lane integer registers
    - Maximum speedup of 4x
  - 2 lane double registers (division)
    - Maximum speedup of 2x
    - Unexplained over-performance of division

## Findings

- A drop in performance when amount of data used reaches 32 KB
- Swapping vector registers out (register spill) from the working set creates a large overhead

## Future Work

- Parallelization of basic math functions for real numbers
- Extension of this process into more areas of the R language implementations

## Acknowledgements

- Haichuan Wang
- PURE committee

## References