

GPU ACCELERATION OF FUNCTIONAL SOFTWARE TESTING

Vanya Yaneva Ajitha Rajan Christophe Dubach



vanya.yaneva@ed.ac.uk
homepages.inf.ed.ac.uk/s0835905

1. MOTIVATION



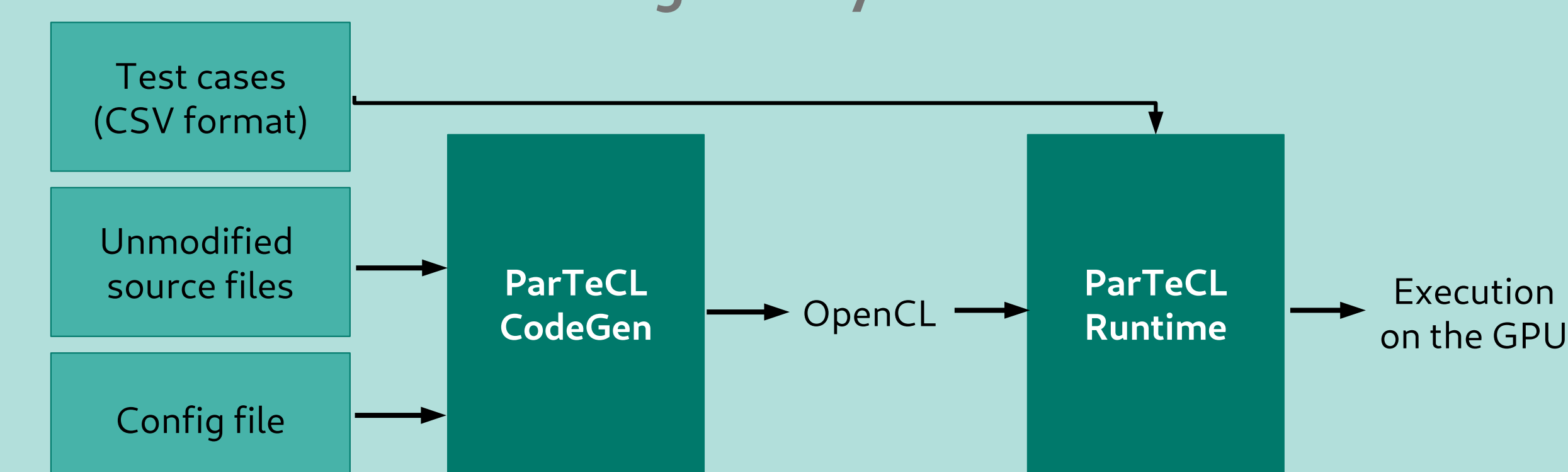
- **Background**
Software is everywhere and its correctness is critical.
- **Problem**
Functional software testing is crucial, but **extremely time consuming**.
- **Proposed solution**
Execute test cases in parallel on the GPU threads.

2. METHODS

Introducing **ParTeCL** – a two-step tool to automatically execute tests on the GPU threads.

- **ParTeCL CodeGen** generates OpenCL kernel from the tested C program so that software testers and engineers don't need specialist GPU knowledge.

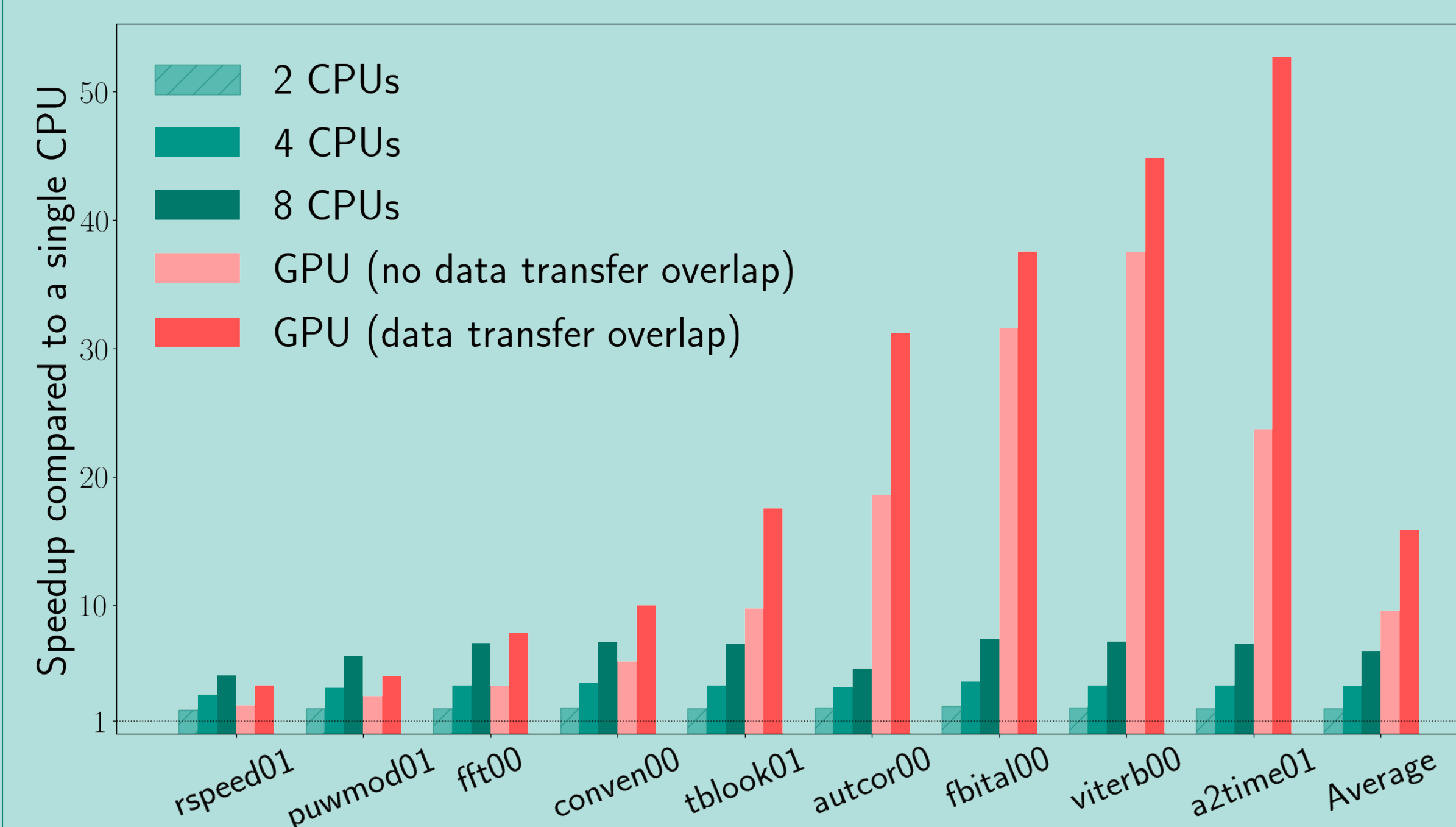
- **ParTeCL Runtime** launches the tests on the GPU threads so testing is fully automated.



github.com/wyaneva/partocl-codegen
github.com/wyaneva/partocl-runtime

3. RESULTS

Testing Embedded C Software
(experiments with 9 benchmarks from the EEMBC industry benchmark suite)



- Speedup **up to 53x** (average **16x**)
- Testing results on the GPU **are the same** as those on the CPU.

GPU: Nvidia Tesla K40m
CPU: Intel® Xeon®

4. CONCLUSIONS

We can automatically accelerate the functional testing of embedded C programs on the GPU.

FUTURE WORK

- Extend to other application domains. Currently working on accelerating testing of finite state machines.
- Extend the scope for C applications. Add support for recursion, dynamic memory allocation and file IO.