VPSDK : a portability library for extended arithmetic operations
targetting a RISC-V Variable eXtended Precision accelerator.

Linear algebra kernels

Direct and itertative linear algebra kernels, such as linear solvers or eigensolvers, are ubiquitous in both scientific and industrial applications. But they both encounter issues.

Iterative : \[ x_{k+1} = x_k + \alpha_k (b - Ax_k) \]

Direct : \[ A = LU; x = U^{-1} L^{-1} b \]

Impact of numerical precision on iterative solver

Extended precision limits the impact of round-off errors and thus speed-up, and in some case enable, the convergence of iterative methods.

Variable precision solutions

Multiple solutions exists to handle variable precision in application code. Some are software only solutions and other are based on proprietary hardware accelerator. Each of them have their own advantages and limitations, but they all need specific programming models.

VPSDK provides

- A common interface for VPFloat numbers independent of VP backend choice
- Functions to manage dense and sparse matrices
- BLAS implementation for VPFloat numbers

VPSDK will be delivered in open-source

#include <VPFloat.hpp>
using namespace VPFloatPackage;

void main() {
    // VPFloat precision settings
    std::int16_t precision = 128;
    std::int16_t exponent_size = 7;
    std::int16_t stride_size = 1;
    std::int16_t n = 10;
    std::int16_t vpfloat_memory_size = precision;

    // VPFloat array declaration
    VPFloatArray r_k[16];

    // VPFloat scalar declaration
    VPFloat rs = precision;

    for (int i = 1; i < 100; i++) {
        r_k[i-1] = rs / (double)i;
    }

This work has been performed in the context of the EPI and TRISTAN projects. EPI has received funding from the European Union’s Horizon research and innovation program under Grant Agreement EPI-SGA1: 826647. TRISTAN has received funding from the Key Digital Technologies Joint Undertaking (KDT JU) under Grant Agreement nr. 101959474.