1st R-CCS International Symposium

Piecewise polynomial approximation algorithm for short-range intermolecular interaction on wide SIMD architectures

Kentaro Nomura[1], Yutaka Maruyama[1], Keigo Nitadori[1], Jun Makino[1,2]

1. RIKEN Center for Computational Science 2. Kobe University

Background

- Interaction in molecular dynamics simulations, which can divided into short- and long-range part in Ewald sum method, dominates the calculation time.
- To accelerate, the short-range force calculation has been replaced with rational approximation, table lookup interpolation, polynomial approximation.
- Piecewise polynomial approximation is a powerful technique to approximate functions with low order polynomials keeping accuracy for small sections.

  Pro : multiply and add operation rich, division free.

  Con : Random memory access to table of polynomial coefficients.

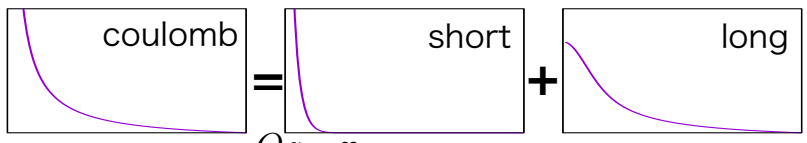
512-bit-width SIMD registers of novel high-performance CPUs can serve as an efficient table for piecewise polynomial approximation.

Objective

To demonstrate a new implementation of piecewise polynomial approximation algorithm using wide SIMD registers as a table to accelerate the calculation of short-range intermolecular interaction.

Algorithm

Short-range interaction in Ewald sum method



$$\boldsymbol{F}(r) = \frac{Qq}{4\pi\epsilon_0 r^2}\frac{\boldsymbol{r}}{r} = \boldsymbol{F}_{\mathrm{short}}(r) + \boldsymbol{F}_{\mathrm{long}}(r)$$

$$\boldsymbol{F}_{\mathrm{short}}(r) = \frac{Qq}{4\pi\epsilon_0}\left\{\frac{\mathrm{erfc}(\alpha r)}{r^2} + \frac{2\alpha}{\sqrt{\pi}}\frac{\exp(-\alpha^2 r^2)}{r}\right\}\frac{\boldsymbol{r}}{r}$$

Q,q : charges, r : distance between charges, ε_0 : permittivity of vacuum

Coulomb interaction is divided into two part, short- and long-range part. The short-range part includes complementary error function and exponential function which results in highly compute intensive.

Existing method: Rational function approximation (RFA)

GROMACS*, one of the fastest MD simulation package, adopts RFA.

$$\boldsymbol{F}_{\mathrm{short}}(r) = \frac{Qq}{4\pi\epsilon_0}\left\{\frac{1}{r^3} - \frac{\mathrm{erf}(\alpha r)}{r^3} + \frac{2\alpha}{\sqrt{\pi}}\frac{\exp(-\alpha^2 r^2)}{r^2}\right\}\boldsymbol{r}$$

$$\frac{\sum_{m=0}^{10} b_m(\alpha^2 r^2)^m}{\sum_{m=0}^{5} c_m(\alpha^2 r^2)^m}$$

$b_m^{(l)}$,$c_m^{(l)}$: mth coefficient of polynomial function

The correlation part of short-range force is approximated by rational function. For details, see pmeForceCorrectionSingleAccuracy on http://manual.gromacs.org/documentation/current/doxygen/html-lib/group__module__simd.xhtml.

*Abraham, et al., Software X, 1-2, 2015.

New method: Piecewise polynomial approximation (PPA)

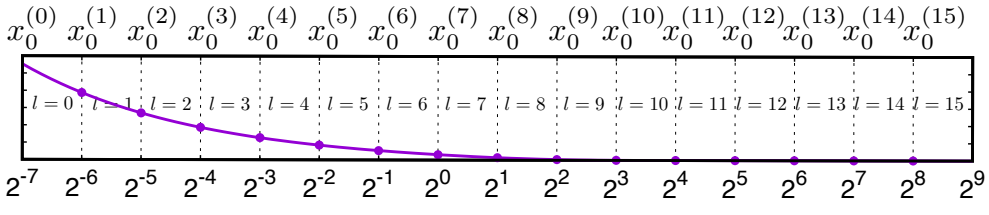
$$F_{\mathrm{short}}(r) = \frac{Qq}{4\pi\epsilon_0}\left\{\mathrm{erfc}\left(\sqrt{(\alpha r)^2}\right) + \frac{2}{\sqrt{\pi}}\exp(-\alpha^2 r^2)\sqrt{(\alpha r)^2}\right\}\frac{\boldsymbol{r}}{r^3}$$

$$x = (\alpha r)^2 - x_0^{(l)}$$

$$F_{\mathrm{approx}}^{(l)}(x) = \left(\sum_{m=0}^{n} a_m^{(l)} x^m\right)^2$$

α : constant of Ewald sum method.
l : index of regions where function is approximated.
$x_0^{(l)}$: lower bound of $(\alpha r)^2$ in region l.
$a_m^{(l)}$: mth coefficient of polynomial function in region l.

Deformation of original short-range force formula to make target function not change drastically brings higher accuracy.

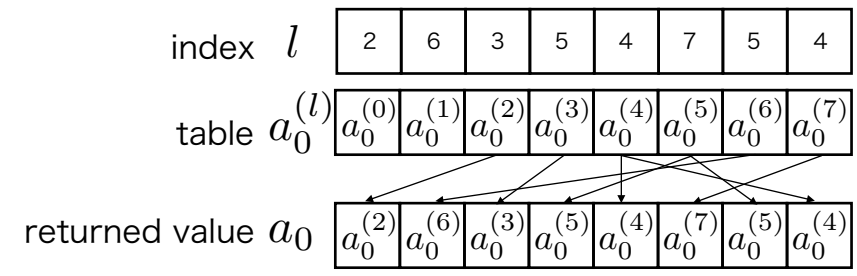


To make approximated functions accurate,
- Intervals ($x_0^{(l)}$) are set to power of 2.
- Sollya* is used for computing minimax polynomials of each l.

*S. Chevillard, M. Joldeș and C. Lauter. "Sollya: an environment for the development of numerical codes," In *Mathematical Software - ICMS 2010*, pages 28-31, Heidelberg, Germany, September 2010. Springer.

Implementation

Table lookup/permute instruction for 512-bit SIMD



512-bit SIMD registers can be used as a table with instructions below:
For Intel architecture
- __m512 _mm512_permutexvar_ps(__m512i idx, __m512 a)
- __m512 _mm512_permutex2var_ps(__m512 a,__m512i idx, __m512 b)

For ARM v8 architecture
- svfloat32_t svtbl_f32(svfloat32_t a, svuint32_t idx)

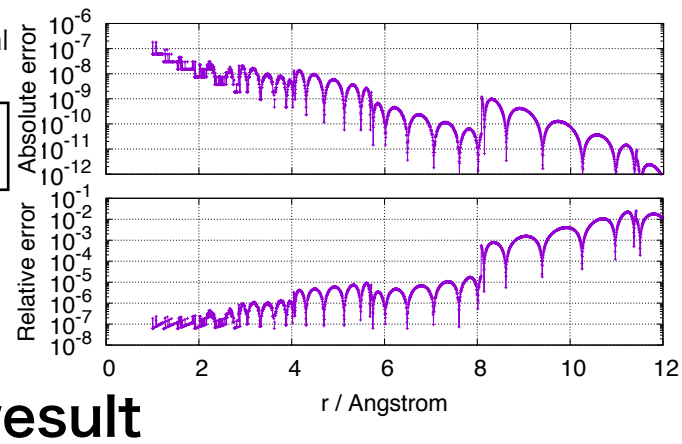
Pseudo code of PPA

Require:
  $a_m^{(l)}$: polynomial coefficients.
  X.f : X as an floating point value w/o changing any bit.
  X.s : X as an integer value w/o changing any bit.
  offset : offset binary of exponent bit.
function *n*th-polynomial approximation{
  $r^2 = |r_i - r_j|^2$;
  $R^2 = \alpha^2 * r^2$;
  $l = (((R^2).\mathrm{s} - \mathrm{offset}) >> 23)\ \&\ 0\mathrm{x}0\mathrm{f}$;
  $x_0^{(l)} = ((R^2).\mathrm{s}\ \&\ 0\mathrm{xff800000}).\mathrm{f}$;
  $x = R^2 - x_0^{(l)}$;
  return $\left(\sum_{m=0}^{n} a_m^{(l)} x^m\right)^2$;
}

Accuracy

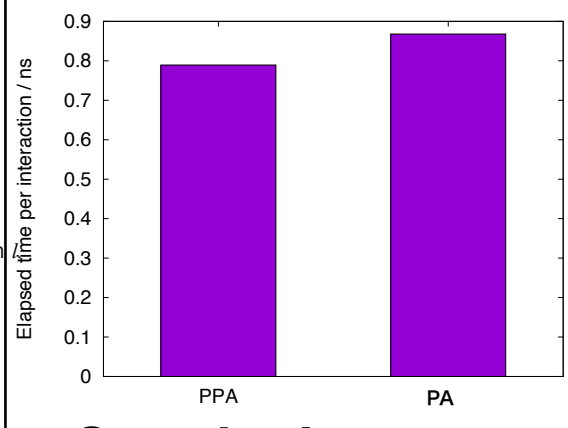
Comparison with the original force in DP and PPA in SP.

Enough accuracy for SP calculation.



Benchmark result

N^2 calculation is applied for 32768 point charge distributed on cubic lattice.



CPU: Intel Xeon Gold 6140@2.30GHz
Compiler: icc 17.0.0 20160721
Option: -xCORE-AVX512 -ip
  -unroll-loops
  -use-fast-math

PPA is10% faster than the previously known polynomial-approximation implementation.

Conclusion

We proposed a piecewise polynomial approximation algorithm which uses SIMD registers as a table of polynomial coefficients for efficient calculation of short-range intermolecular interaction.
- Enough accurate for for the calculation in single-precision.
- 10% faster than the best previously known SIMD implementation on Intel Skylake Xeon.