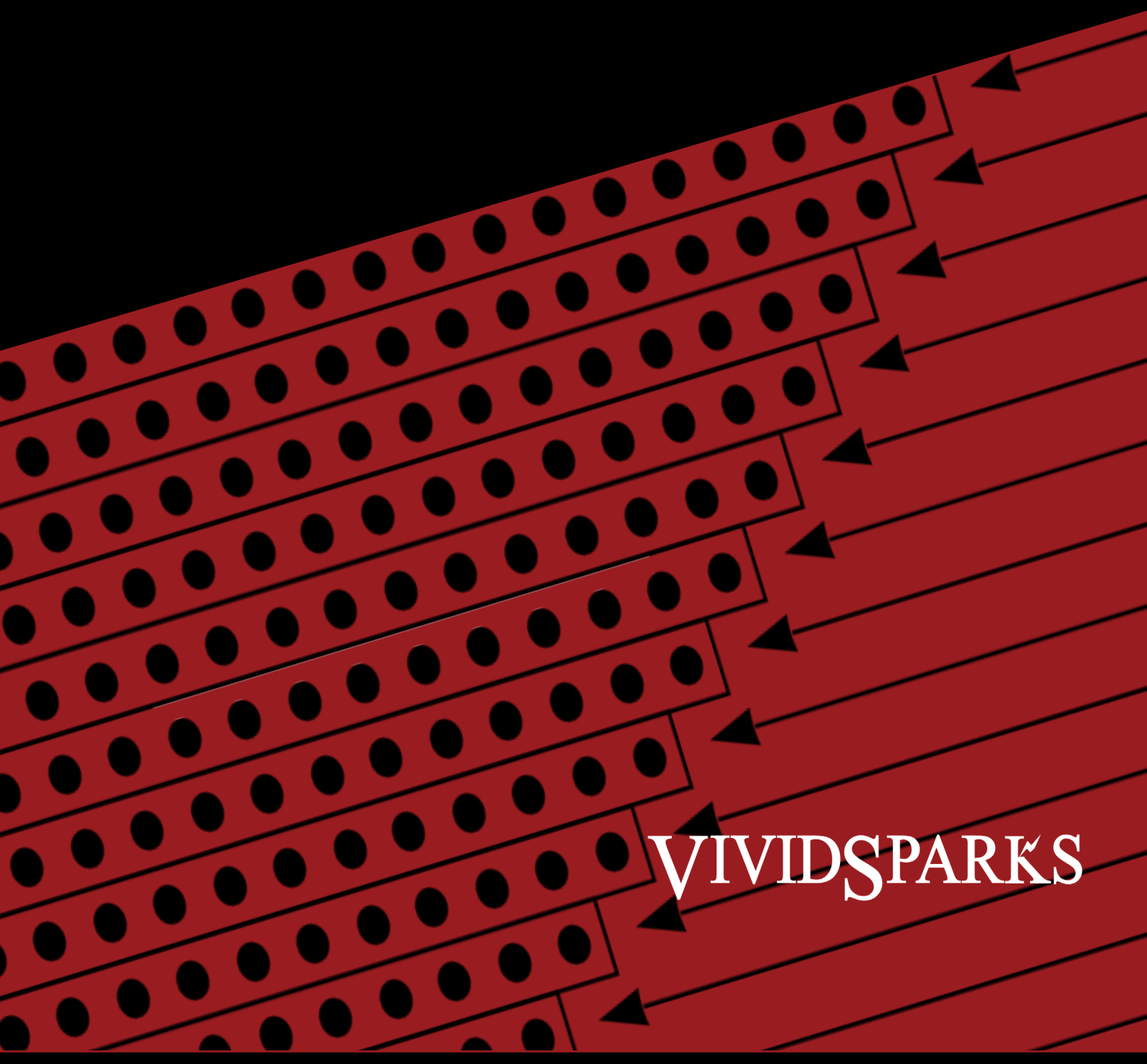
POSIT For Next Generation Computer Arithmetic



VIVIDSPARKS

Executive Summary

Everything in the world can be represented using Mathematics, from the smallest, for instance, the symmetry of a leaf, to the largest, for instance, the complexity of human brain. Math is required for the simplest day to day calculations to the more complicated 3D graphics and scientific computing. Computers compute math using IEEE-754 Floating Point (FP) number system. The FP system has limited dynamic range, less accurate results and does not properly obey mathematical laws. Due to these limitations scientists, programmers and algorithmist need to adjust their models compromising on accuracy. Posit number system on other hand provides much better accuracy with same data width as that of FP system which leads to higher performance with silicon area.

Introduction

FP arithmetic is widely used in many applications such as Artificial Intelligence (AI), Machine Learning (ML), High Performance Computing (HPC), communications, etc. In order to understand functions or models or equations in these applications, one has to simulate or compile or execute the equations. The response time of a processor or a accelerator to equations, functions or models is a critical factor. Faster the processing speed of the underlying hardware, quicker one can simulate or execute these models, equations or functions with better predictions. Unfortunately, underlying Floating Point Unit (FPU) in state-of-the art processors or accelerators [1] lacks speed and accuracy. In fact, in order to meet the demand of today's applications performance processors and accelerators are fabricated in higher technology node.

Posit number system on other hand [2], [5] provide much efficient encoding scheme of numbers in computer arithmetic. Posit numbers do not have overflows, underflows and no subnormals leading to large dynamic range with more accurate results with simper rounding modes.

Examples of Measured Posit Benefits

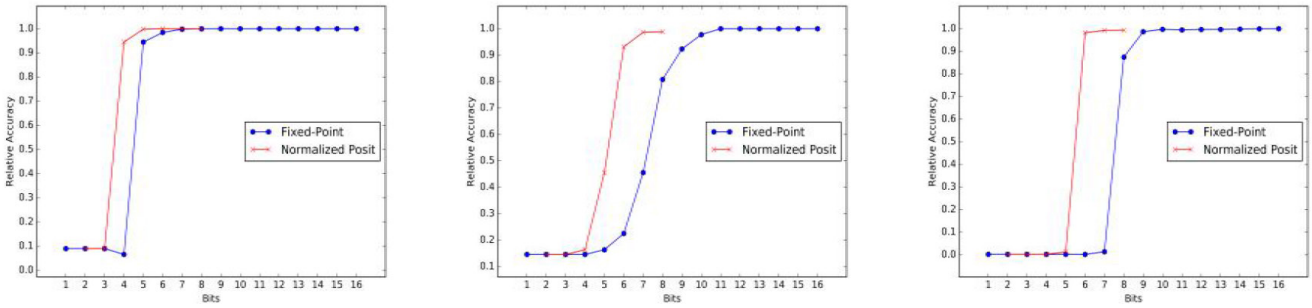
AI, HPC, and Predictive Analytics

Performing interference step of deep learning in resource constrained environments in embedded devices is challenging. Efforts require optimization at both hardware and software levels. Seyed H. F [3] et. all have conducted experiments using fixed-point number system and posit number system and shown that posits outperform fixed-point number system. The team used MNIST [6] dataset, CIFAR-10 dataset [7] and subset of the ImageNet [8]. Different Deep Convolutional Neural Networks (DCNNs) are used for each dataset and single FP number was selected for baseline implementation. Accuracy results are shown in following Table 1.

Task	Dataset	#interference set	Network	Layers	Top-1 accuracy
Digit Classification	MNIST	10000	LeNet	2 Conv and 2FC	99.03%
Image Classification	CIFA-10	10000	ConvNet	3 Conv and 2FC	68.45%
Image Classification	ImageNet	10000	AlexNet	5 Conv and 3FC	55.45%

Table 1 Top-1 accuracy of 3 different Neural Networks.

Weights are represented by a variable length fixed-point number system (with maximum of 16 bits) and Posit <8,0> number system. The relative accuracy of different tasks are shown in Figure 1.



(a) (b) (c)

Figure 1 Results showing the relative accuracy to the baseline DCNN implementation on various datasets with representation of weights using variable length fixed-point number system and normalized posit number system (posit<8,0>). (a) Relative accuracy results for LeNet on MNIST dataset. (b) Relative Accuracy results for ConvNet. (c) Relative accuracy results for AlexNet on ImageNet dataset.

The posit <8,0> outperformed the fixed-point number system in terms of accuracy with fewer bits. The results demonstrates that it is possible to perform LeNet, ConvNet, AlexNet with 5 bits, 7 bits and 7 bits respectively with less than 1% accuracy degradation in comparison to performance of the same networks with 7 bits, 11 bits and 9 bits respectively while using the variable length fixed-point number system. The team then claims that reduction in memory utilization by 28.6%, 36.4% and 23% as compared to state-of-the-art variable length fixed-point implementations [9], [10] and can also significantly reduce the number of memory access through memory concatenation schemes.

Milan Klöwer and his team from Oxford University [11] have conducted experiment on posits as an alternative to FP number system using shallow water equations. The shallow water equations result from a vertical integration of the Navier-Stokes equations under the assumption that horizontal length scales are much greater than vertical scales. The shallow water equations for the prognostic variables velocity u = (*u*,*v*) and sea surface elevation η are

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} + f\hat{\mathbf{z}}\times\mathbf{u} = -g\nabla\eta + \mathbf{D} + \mathbf{F} \qquad (1a)$$

$$\frac{\partial \eta}{\partial t} + \nabla\cdot(\mathbf{u}h) = 0. \qquad (1b)$$

For the atmosphere, η is interpreted as pressure. The shallow water system is forced with a zonal wind stress F. The dissipation term D removes energy on large scales (bottom friction) and on small scales (diffusion). The non-linear term (u·∇) u represents advection of momentum. The term $f\hat{z}\times$ u is the Coriolis force and $-g\nabla\eta$ is the pressure gradient force, with *g* being the gravitational acceleration. Eq. 1b is the shallow water-variant of the continuity equation, ensuring conservation of mass. The domain is a zonally periodic rectangular channel of size 2000 km ×1000 km, with a meridional mountain ridge in the middle of the domain.

The solution to the shallow water equations includes vigorous turbulence that dominates a meandering zonal current. Using either float or posit arithmetic with 16 bits the simulated fluid dynamics are very similar to a double precision reference: As shown in a snapshot of tracer concentration (Figure. 2) stirring and mixing can be well simulated with half precision floats and with 16 bit posits (2 exponent bits). This provides a first evidence that the accumulated rounding errors with posits are smaller than with floats.

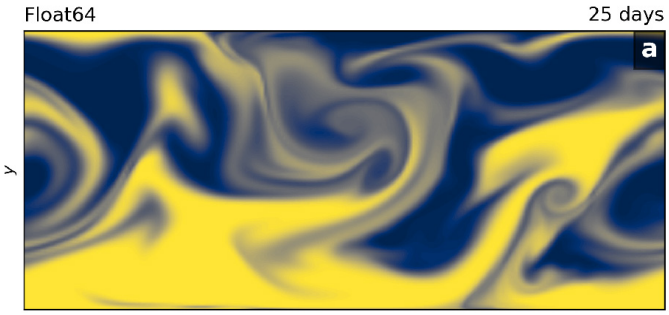


Figure 2 a

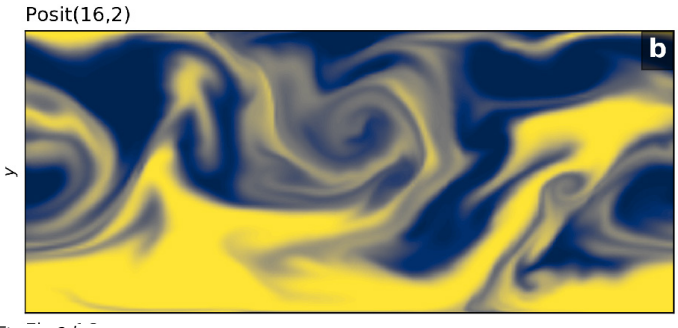


Figure 2 b

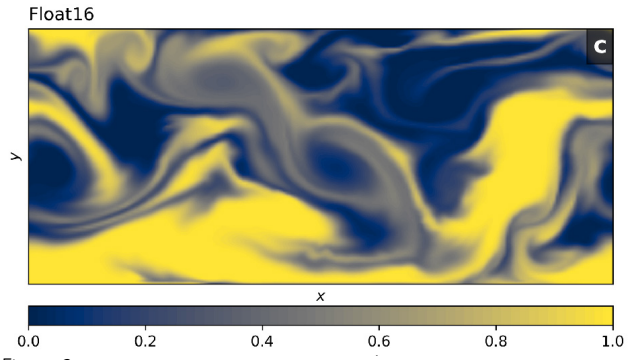


Figure 2 c

Figure 2 Snapshot of tracer concentration simulated by the shallow water model, based on (a) double precision floats and (b) posit arithmetic (16bit with 2 exponent bits) and (c) half precision floats. The tracer was injected uniformly in the left half of the domain 25 days before. This simulation was run at a resolution of $\Delta = 10$km (200x100 grid points). The corresponding video can be found at http://milank.de/videos/swm_posit_tracer.mp4

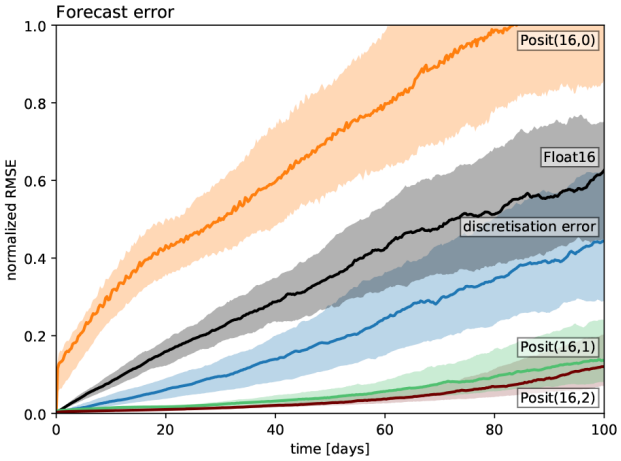


Figure 3 Forecast error measured as the root mean square error (RMSE) of sea surface height taking the double precision forecast as reference. The RMSE is normalised by a mean forecast error at very long lead times. Solid lines represent the median of 280 forecasts per number format. The shaded areas denote the interquartile range.

To quantify differences between reduced precision arithmetics the team performed model forecasts that compare rounding errors. The forecast error in the shallow water model is computed as root mean square error (RMSE) taking the model based on double precision floating point arithmetics as reference truth. They then use the sea surface height (equivalent to pressure) to compute the forecast error as this variable captures the large scale circulation. The forecasts are created based on 280 different initial conditions from random start dates of a 50 year long control simulation. Each forecast is performed several times from identical initial conditions but with the various number formats. To compare the magnitude of rounding error that are caused by a reduction in precision to a realistic level of error that is caused by model discretisation, they also perform forecasts at double precision that fall back to a 3rd-order Runge-Kutta scheme for time integration and a simpler enstrophy conserving advection scheme described in Sadourny [12].

Clearly the best forecast is obtained for posit arithmetic with 1 or 2 exponent bits (Figure 3), with a small accumulation of rounding errors even for lead times of 100 days. The forecast error for 16 bit posits without exponent bit increases quickly (Figure 3), especially for short forecast lead times, but a persistence forecast, i.e. assuming the initial conditions persist over time, is still worse (not shown).

Another most thorough study to date to quantify the benefits of posits over IEEE Floating Point has been produced by numerical analysts Peter Lindstrom, Scott Lloyd, and Jeffrey Hittinger at Lawrence Livermore National Laboratory. They have reported on an experiment to compare the new posit system against traditional IEEE floating point [4] and found the posit number system to outperform IEEE floating point across the board. The experiment modified a numerical simulation application, Euler2D, which implements an explicit, high-resolution Godunov algorithm to solve the Euler system of equations for compressible gas dynamics on an L-shaped domain. Such a solver is simple enough to instrument and comprehend while providing sufficient complexity in the numerical behavior of the solution, e.g. a nonlinear hyperbolic system with shock formations and minimal dissipation.

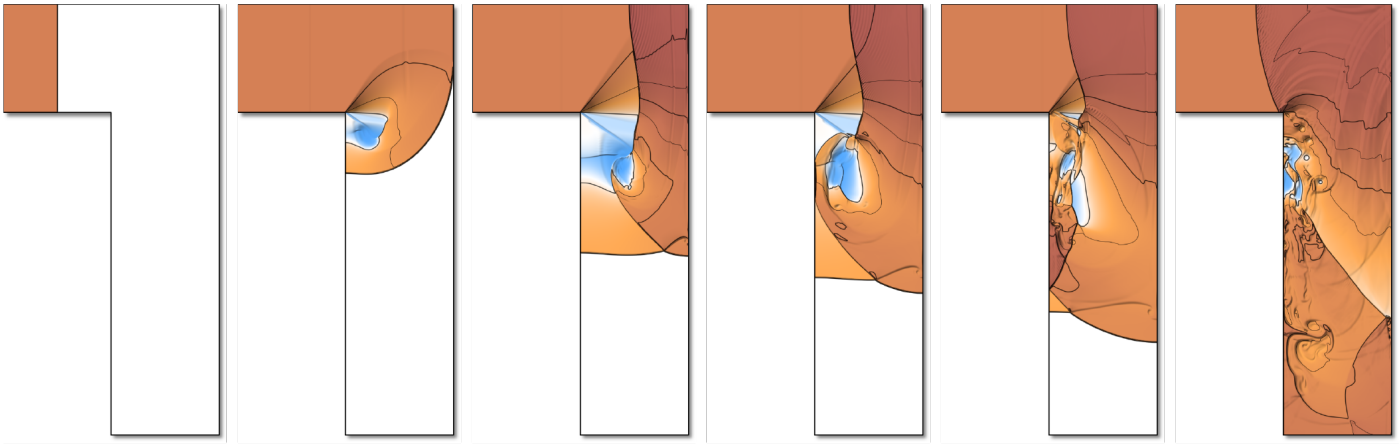


Figure 4: Snapshots in time, t, from the Euler2D mini-application showing the evolution of the density field in an L-shaped chamber. Blue color indicates density lower than the initial density (red) of the shock wave. (a) Initial state. (b) Shock reflects off of the far wall. (c) Reflected shock hits vortex. (d) Shock reflects off of near wall. (e) Second reflection hits vortex. (f) Final state.

The problem solved in the Euler2D code is the propagation of a shock wave in air through an L-shaped conduit. The domain is the union of two rectangles: [(0,3), (2,4)] U [(1,0), (2,3)]. At the initial time, a shock, moving with dimensionless speed M_s = 2.5 relative to the quiescent state of (ρ, u_x, u_y, p) = (1,0,0,1), is positioned at x= 0.5. The inlet flow at x = 0 is constant. The code is run with uniform mesh of size h = 1 ⁄ n= 1 ⁄256 using a fixed time step of $\Delta t \approx 2.8\ e^{-4}$, resulting in roughly 1.3 trillion floating-point operations over the entire run. The system dynamics are shown in Figure 4 the shock propagates into the chamber and diffracts around the corner, initiating the shedding of a vortex from the corner. At time t ≈ 0.51, the initial shock reflects off the far wall, and the reflected shock propagates back upstream, encountering the vortex around time t ≈ 0.95. The reflected shock breaks up the vortices shedding off the corner and reflects again off the near wall at several times. Eventually, the flow moves down the channel with a propagating sequence of oblique shock waves and a great deal of wave-wave interactions.

A pointwise, closed form solution to the Euler 2D hyperbolic PDE does not exist. To establish ground truth, the LLNL team used a quadruple precision floating point type to compute a high-precision solution. The team then computed the root mean square pointwise error in the density field to establish solution accuracy. The team reported that the RMSE was expected to be dominated by round-off error associated with each numerical type due to fixed discretization parameters, i.e., fixed truncation error. Plots of the pointwise error in the density field over time for 32-bit and 64-bit representations relative to the quadruple precision are shown in Figure 5 and Figure 6 (next page). We see spikes in error that correlate with events such as shock-wall and shock-vortex impact. These spikes are more pronounced in the 64-bit plot because of the additional precision provided in 64-bit arithmetic.

IEEE floating point and related types do quite poorly in relation to posits and other tapered precision numerical types, most evident in the 64-bit precision plot, where posit<64,2> outperforms IEEE double precision by nearly three orders of magnitude.

Conclusion

Posit number system provides much better encoding of numbers in computer arithmetics with associated benefits encouraged VividSparks to introduce series of posit accelerators with our own Posit C compiler to computing market. Our products not only provides much better results and high performance but also highly silicon efficient yet consume very less power.

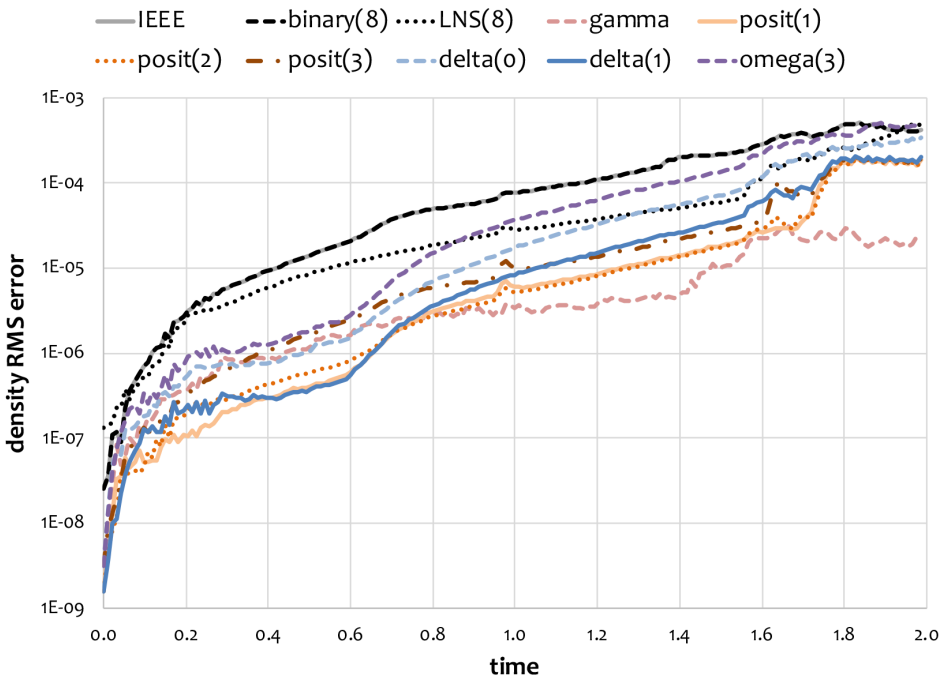


Figure 5: RMSE in Euler2D density field as a function of simulation time and 32-bit number representation.

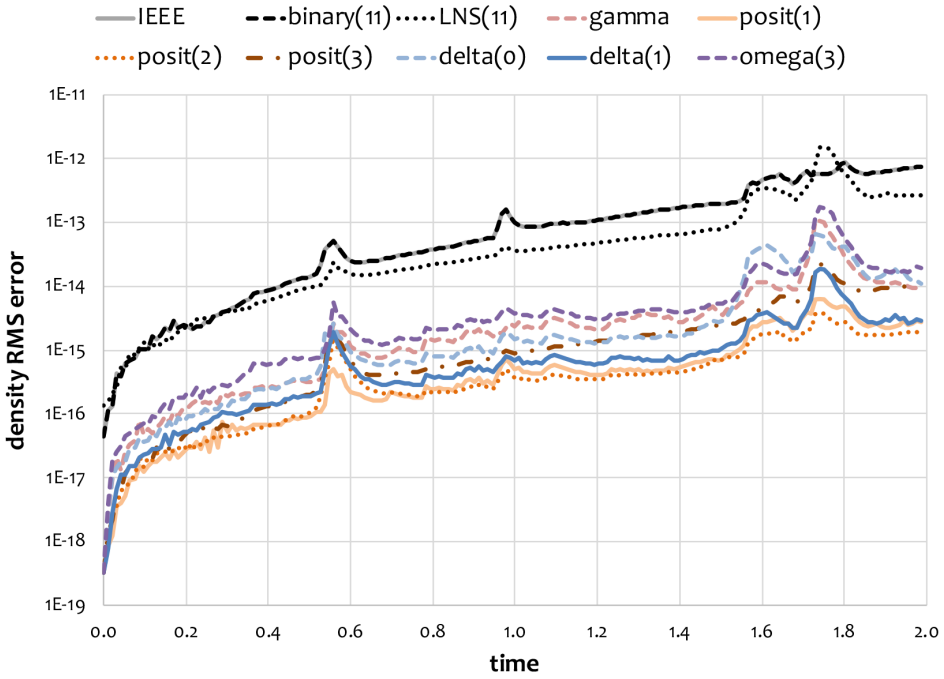


Figure 6: RMSE in Euler2D density field as a function of simulation time and 64-bit number representation.

References

[1] www.intel.com, www.amd.com, www.arm.com, www.nvidia.com

[2] John L. Gustafson, "Beyond Floating Point: Next Generation Computer Arithmetic," Stanford University Seminar: https://www.youtube.com/watch?v=aP0Y1uAA-2Y

[3] H. F. Seyed. Langroudi, Tej Pandit, and Dhireesha Kudithpudi, "Deep Learning Interference on Embedded Devices: Fixed vs Posit", https://arxiv.org/pdf/1805.08624.pdf

[4] P. Lindstrom, S. Lloyd, J. Hittinger, "Universal Coding of the Reals: Alternatives to IEEE Floating Point," CoNGA 2018, March 28, 2018, Singapore. Association for Computing Machinery. ACM ISBN 978-1-4503-6414-0/18/03. DOI:https://doi. org/10.1145/3190339.3190344

[5] Adaptive posit tensor processing for error-free linear algebra, Theodore Omtzigt, http://www.stillwater-sc.com/

[6] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner "Gradient Based Learning Applied to Document Recogniton", Proc. of the IEEE, Vol. 86, no. 11, pp:2278-2324, 1998.

[7] B. Graham, "Fractional Max-pooling", arXiv preprint arXiv: 1412.6071.

[8] O. Russakousky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, M. Bernstein, A. C. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge", Int. J. Comput, Vis, vol.115, no.3, pp 211-252, 2015.

[9] P. Judd, J. Albercio, T. Hetherington, T. Aamodt, N. E. Jerger, R. Urtasun, and A. Moshovos, "Reduced Precision Strategies for bounded memory in deep neural nets" arXiv preprint arXiv:1511.05236, 2015.

[10] P. Judd, J. Albercio, T. Hetherington, T. Aamodt, N. E. Jerger, R. Urtasun, and A. Moshovos,"Proteus: Exploiting precision variability in deep neural networks", parallel computing, 2017.

[11] M. Klöwer, P. D. Düben, T N. Palmer, Posits as an alternative to floats for weather and climate models, CoNGA 2019, March 13-14, 2019, Singapore.

[12] R. Sadourny. 1975. The Dynamics of Finite-Difference Models of the Shallow- Water Equations. , 680–689 pages. https://doi.org/10.1175/1520-0469(1975) 032<0680:TDOFDM>2.0.CO:2.

VividSparks IT Solutions Pvt. Ltd.
License no: U72200K2014OPC077975
#38 BSK Layout, Hubli-580031, India.
www.vivid-sparks.com
inquiry@vivid-sparks.com