

Mixed Precision Runge-Kutta methods using Half Precision on the A64FX Processor

Ben Burnett ¹ Sigal Gottlieb ¹ Zachary J. Grant ²
Alfa Heryudono ¹ Gaurav Khanna ³

¹University of Massachusetts Dartmouth

²Oak Ridge National Laboratory

³The University of Rhode Island

Ookami User Group
February 10, 2022

Table of Contents

- 1 Introduction
- 2 Mixed-Precision Runge-Kutta Methods
- 3 Conclusions and Future Work

Table of Contents

- 1 Introduction
- 2 Mixed-Precision Runge-Kutta Methods
- 3 Conclusions and Future Work

No Free Lunch

“Nothing is acquired for free, and necessarily must cost us some thing”
-Epictetus

No Free Lunch

“Nothing is acquired for free, and necessarily must cost us some thing”
-Epictetus

IEEE 754 Floating Point Standard

Name	Significand Bits	Exponent Bits	Exponent Min	Exponent Max
Half	11	5	-14	+15
Single	24	8	-126	+127
Double	53	11	-1022	+1023
Quadruple	113	15	-16382	+16383

No Free Lunch

“Nothing is acquired for free, and necessarily must cost us some thing”
-Epictetus

IEEE 754 Floating Point Standard				
Name	Significant Bits	Exponent Bits	Exponent Min	Exponent Max
Half	11	5	-14	+15
Single	24	8	-126	+127
Double	53	11	-1022	+1023
Quadruple	113	15	-16382	+16383

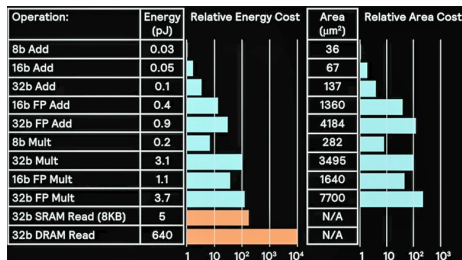


Figure: Mark Horowitz, ISSCC presentation 2014

Table of Contents

- 1 Introduction
- 2 Mixed-Precision Runge-Kutta Methods**
- 3 Conclusions and Future Work

Motivating Example: The Van der Pol Equation

For the General ODE

$$\frac{d}{dt}u = F(u)$$

solve the Van der Pol system

$$u_1' = u_2$$

$$u_2' = u_2(1 - u_1^2) - u_1$$

Motivating Example: The Van der Pol Equation

For the General ODE

$$\frac{d}{dt}u = F(u)$$

solve the Van der Pol system

$$\begin{aligned}u_1' &= u_2 \\u_2' &= u_2(1 - u_1^2) - u_1\end{aligned}$$

using the Implicit Midpoint

$$y^{(1)} = u^n + \frac{\Delta t}{2}F(y^{(1)})$$

$$u^{n+1} = u^n + \Delta t F(y^{(1)})$$

Motivating Example: The Van der Pol Equation

For the General ODE

$$\frac{d}{dt}u = F(u)$$

solve the Van der Pol system

$$u_1' = u_2$$

$$u_2' = u_2(1 - u_1^2) - u_1$$

using the Implicit Midpoint

$$y^{(1)} = u^n + \frac{\Delta t}{2}F(y^{(1)})$$

$$u^{n+1} = u^n + \Delta t F(y^{(1)})$$

Newton-Raphson Method

$$g(y^{(1)}) = 0 = u^n + \frac{\Delta t}{2}F(y^{(1)}) - y^{(1)}$$

To solve, use an expensive iteration until a tolerance is reached

$$y_{i+1}^{(1)} = y_i^{(1)} - J_i^{-1}g(y_i^{(1)})$$

Using Mixed Precision

Mixed-Precision Implicit Midpoint Method

$$y_{\epsilon}^{(1)} = u^n + \frac{\Delta t}{2} F^{\epsilon}(y_{\epsilon}^{(1)})$$

$$u^{n+1} = u^n + \Delta t F(y_{\epsilon}^{(1)})$$

Using Mixed Precision

Mixed-Precision Implicit Midpoint Method

$$y_{\epsilon}^{(1)} = u^n + \frac{\Delta t}{2} F^{\epsilon}(y_{\epsilon}^{(1)})$$

$$u^{n+1} = u^n + \Delta t F(y_{\epsilon}^{(1)})$$

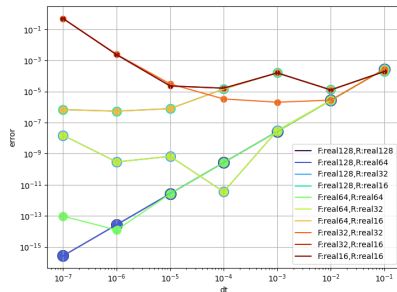


Figure: Errors from mixed-precision implicit midpoint rule

Errors calculated in extended precision regardless of computations base precision from a reference solution done with Real128 precision using a 4th order RK method with smaller Δt

Using Mixed Precision

Mixed-Precision Implicit Midpoint Method

$$y_{\epsilon}^{(1)} = u^n + \frac{\Delta t}{2} F^{\epsilon}(y_{\epsilon}^{(1)})$$

$$u^{n+1} = u^n + \Delta t F(y_{\epsilon}^{(1)})$$

Mixed-Precision Implicit Midpoint Method

$$y_{\epsilon}^{(1)} = u^n + \frac{\Delta t}{2} F^{\epsilon}(y_{\epsilon}^{(1)})$$

$$u^{n+1} = u^n + \Delta t F(y_{\epsilon}^{(1)})$$

Corrected Mixed-Precision Implicit Midpoint Method

for $k = 1, \dots, p - 1$

$$y_{[0]}^{(1)} = u^n + \frac{\Delta t}{2} F^{\epsilon}(y_{[0]}^{(1)})$$

$$y_{[k]}^{(1)} = u^n + \frac{\Delta t}{2} F(y_{[k-1]}^{(1)})$$

$$u^{n+1} = u^n + \Delta t F(y_{[p-1]}^{(1)})$$

Performance of Mixed-Precision Methods

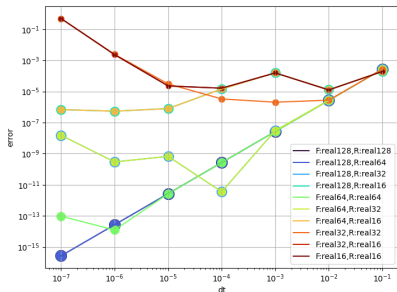


Figure: No corrections

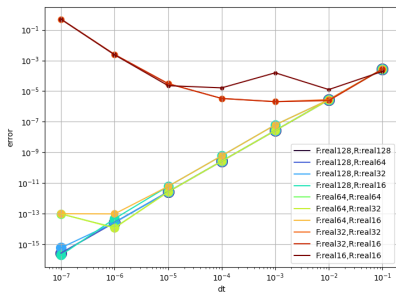


Figure: One correction

Performance of Mixed-Precision Methods

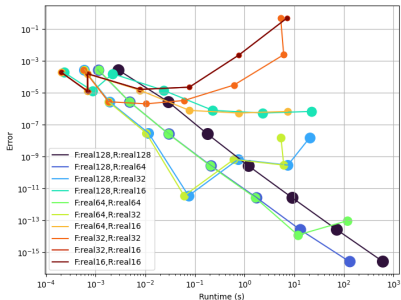


Figure: No corrections

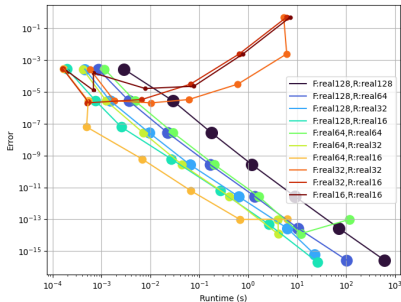


Figure: One correction

Power Consumption of Mixed-Precision Methods

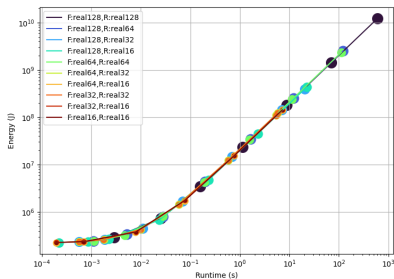


Figure: Energy (J) Consumption vs Run Time of Solver

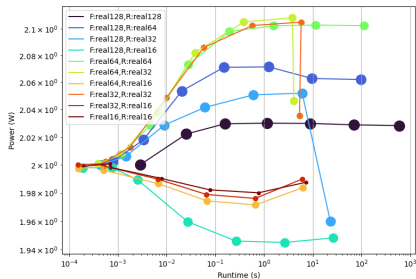


Figure: Power (W) Consumption vs Run Time of Solver

Power and energy consumption was collected from the A64FX processors energy counters using `perf`

Table of Contents

- 1 Introduction
- 2 Mixed-Precision Runge-Kutta Methods
- 3 Conclusions and Future Work**

Conclusions

Key Takeaway:

Potential to save an appreciable amount of time while still being able to maintain high accuracy.

Conclusions

Key Takeaway:

Potential to save an appreciable amount of time while still being able to maintain high accuracy.

Further Work:

- More problems (Are problems that require other methods to solve the implicit stage more likely to benefit?)
- More methods (Do solvers with more stages benefit more/less?)

Conclusions

Key Takeaway:

Potential to save an appreciable amount of time while still being able to maintain high accuracy.

Further Work:

- More problems (Are problems that require other methods to solve the implicit stage more likely to benefit?)
- More methods (Do solvers with more stages benefit more/less?)

Thank you for attending! Any questions?