IRKGL16: Implicit RK Solver For High Precision Numerical Integration

Context

- We focus on solving **non-stiff ODEs** with **high accuracy**
- Most numerical integration of ODEs is performed with double-precision arithmetic (64-bit) but some require **extra precision**
- Examples: chaotic problems, integration over long time periods,...
- julia language supports **arbitrary precision number systems** for fast solving with high accuracy

Motivation

- Develop a new solver written in Julia
  - **Contribution** to DifferentialEquations.jl ecosystem
- Show that Implicit methods can be **more efficient** than explicit ones
- IRKGL16 implicit solver (an 8-stage IRK scheme based on Gauss-Legendre nodes) can take advantage of **modern computer technology**:
  - Parallelism
  - Mixed-precision arithmetic

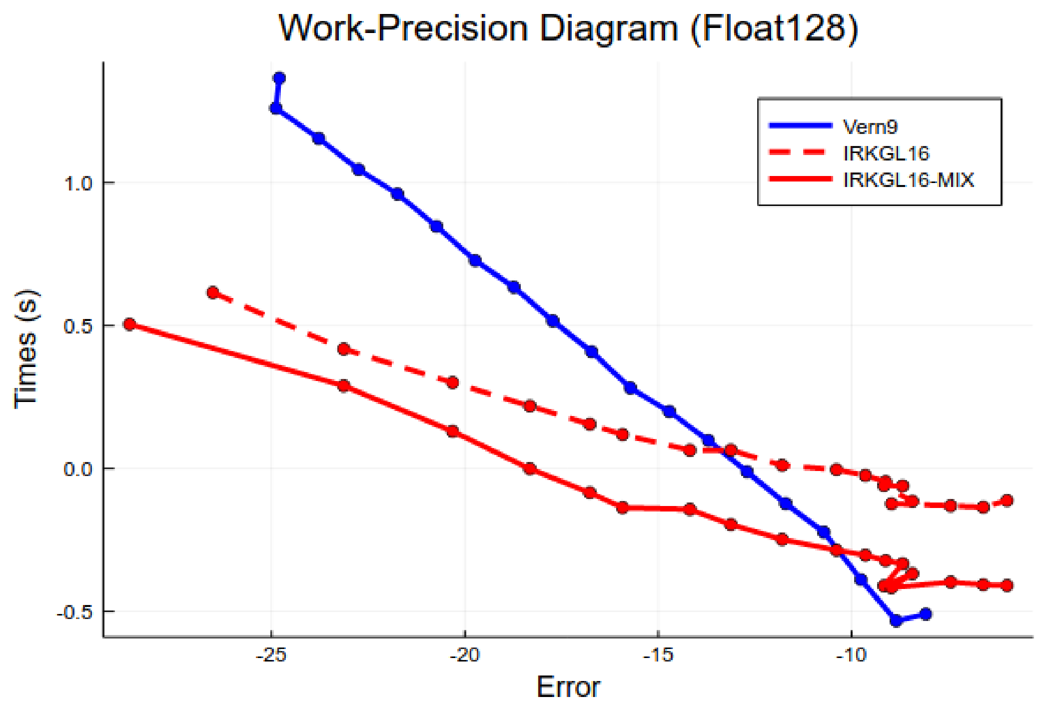
M. Antoñana, E. Alberdi, J. Makazaga and A. Murua
University of Basque Country
Applied Mathematics

eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea

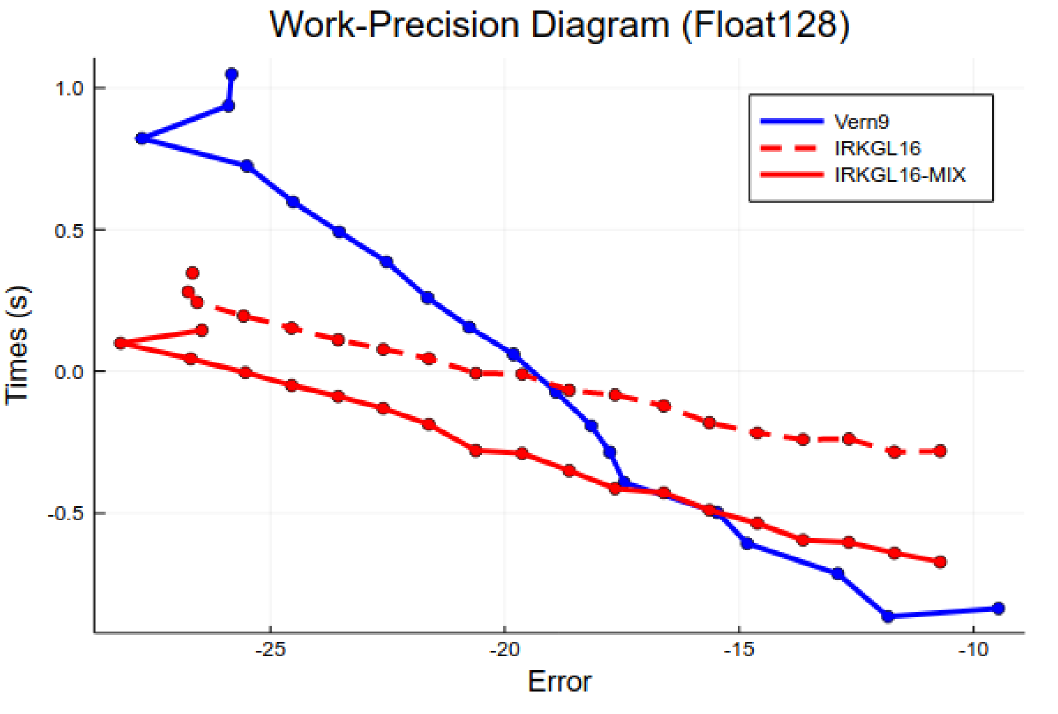
General First Order ODE Problems $\frac{du}{dt} = f(u,t)$

Challenge: improve efficiency of DifferentialEquations.jl's **Vern9** algorithm

Discrete NonLinear Schrödinger (NLS) with N=5



Restricted Three Body Problem (RTBP)



Conclusions from experiments

- **IRKGL16** is more efficient than **Vern9** for high precision computations
- Mixed-precision arithmetic improves performance of IRKGL16 solver

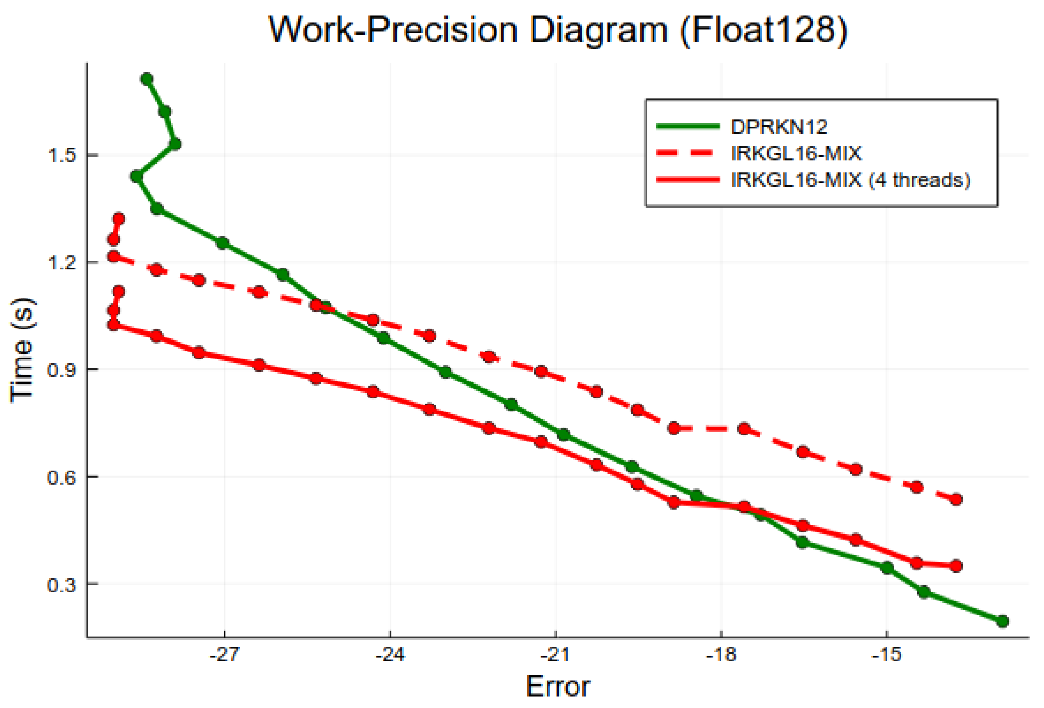
  Combining quadruple precision (128-bit) arithmetic with double precision arithmetic (64-bit)

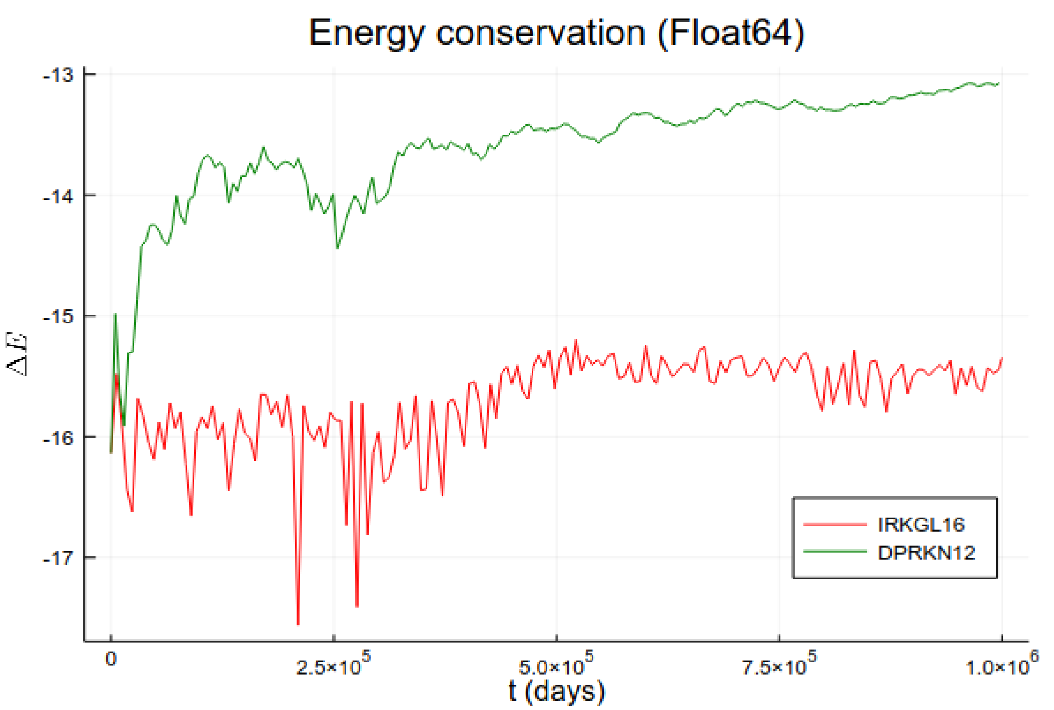
Second Order ODE Problems $u = (q, v), \quad \frac{dq}{dt} = f(v), \quad \frac{dv}{dt} = g(q,t)$

Challenge: improve efficiency of DifferentialEquations.jl's **DPRKN12** algorithm

Pleiades Problem (PLEI)

Solar-System Problem (SSP)





Conclusions from experiments

- PLEI: **IRKGL16-MIX (4 threads)** is more efficient than **DPRKN12** for high precision computations

  Multithreading version of IRKGL16 solver based on stage-wise parallelization
- SSP: **IRKGL16 fixed-step size** computation conserves energy better than **DPRKN12** (for comparable execution time) for long time integration

Summary

https://github.com/mikelehu/IRKGaussLegendre.jl

Conclusions

About julia

- Julia's high-level syntax allows good **productivity** and **well organized code**
- DifferentialEquations.jl ecosystem offers functionality for easy integration of **new methods**

About IRKGL16

- It is **competitive for high-accuracy computations** that exceeds double precision arithmetic (tol<1e-16)
- **Mixed-precision** arithmetic and **multihreading** enhance efficiency of the solver

Future work

- **Fully Integrate** as an algorithm on the common interface of DifferentialEquations.jl
- Implementation of parallelized implicit ODE solvers for **large ODE systems**

Acknowledgments

- This work has received funding from Department of Education of the Basque Government through **MATHMODE Research Group** (IT1294-19)
- Thanks to **Chris Rackauckas** for his help throughout this project