Universidad de Investigación de Tecnología Experimental Yachay Tech

School of Mathematical and Computational Sciences

Program : Undergraduate —————— Semester : VII
Subject : HPC Professor : Saravana Prakash T
Date : May 24, 2019 Time:
Semester: Jan-May-2019
Student's Surname, Name : Zhapa, Fernando
Student Reg.No:

Neatly show all of your work. If you need to use extra paper, please indicate so and attach to this exam.
Mode of submission : Upload your answers in D2L or hard-copy at my desk
Turn down the quiz /submit : Monday (11/02/2019), at 1:00 PM.

Final Project: Free Kick Simulation: A comparison between MPI, OPENMP and Serial algorithms

1 Introduction

In this work, we make a comparison of MPI, OPENMP and Serial versions of an algorithm that simulates a free kick of a soccer game. A free kick is a case of parabolic movement in three spatial dimensions and one temporal dimension. Thus, it corresponds to a problem of differential equations. For this reason we have applied integrators like the Runge-Kutta and Beeman methods. We expected also to apply Verlet method, however, it is not applicable for this problem since Verlet is independent of velocity but free kick problem needs velocity as a initial value.

We had another options like Verlet with velocity or Euler methods. The problem is that their iterative formulas converge to the same formula as Beeman method. The reason is that those three methods iterate over acceleration, which for free kick is an irrelevant aspect since the only acceleration that free kick undergoes is the acceleration of gravity and its change is negligible.

We present a comparison in function of time of Runge-Kutta and Beeman methods both with version in serial, MPI and OpenMP.

2 Methodology

The free kick problem and the Runge-Kutta method in serial version had been taken from [2]. Information about Verlet, Euler and Beeman methods haven been retrieved from [1].

Runge-Kutta method is based on computing the next value calculating four derivatives at different points. The formula is:

$$x_{n+1} = x_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{1}$$

In this formula, x_i is any function that depends on time, h is a constant and k_i are presented as follows:

$$\begin{aligned} k_1 &= f(t_n, v_n) \\ k_2 &= f(t_n + \frac{h}{2}, v_n + \frac{h}{2}k_1) \\ k_3 &= f(t_n + \frac{h}{2}, v_n + \frac{h}{2}k_2) \\ k_4 &= f(t_n + h, v_n + hk_3) \end{aligned} \tag{2}$$

On the other hand, Beeman method is based on the formulas:

$$x(t + \Delta t) = x(t) + v(t)\Delta t + \frac{2}{3}a(t)\Delta t^2 - \frac{1}{6}a(t - \Delta t)\Delta t^2 \tag{3}$$

and

$$v(t + \Delta t) = v(t) + \frac{1}{3}a(t + \Delta t)\Delta t + \frac{5}{6}a(t)\Delta t - \frac{1}{6}a(t - \Delta t)\Delta t \tag{4}$$

However, in free kick problem in considered invariant with respect of time. That fact makes that equations 3 and 4 transform into:

$$x(t + \Delta t) = x(t) + v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2 \tag{5}$$

and

$$v(t + \Delta t) = v(t) + a(t)\Delta t \tag{6}$$

Verlet with velocity and Euler methods converge to equations like 5 and 6. That is the reason that we focus on the analysis of only Runge-Kutta and Beeman methods.

Additionally, all the methods take into account the mass, the wind force, the drag force, temperature and density. Another magnitudes are the initial position, velocities and spin rate. In this work we analyze for several spin rates.

3 Results

For the computations. a machine with the following characteristics was used:

- **Processor:** Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz
- **RAM:** 16GB
- **Endianness:** Little Endian
- **Number of cores:** 12

All the tests were done with 6 different values of spin rate. Fig. 1 and 2 show how the Runge-Kutta and Beeman methods behave with free kick problem. We have got a sorpresive behaviur because the parallel version run slower than the serial version. However, that has an exaplanation. Free kick problem does not deal with too large data. Only a vector of 6 elements corresponding to position and velocities is being computed in every iteration. That is why for the parallel version 6 cores were chosen. The consequent reason of no having large ammounts of data is that the parallel version suffers of overhead. That is, communication between processes last more than the computations on each processor.

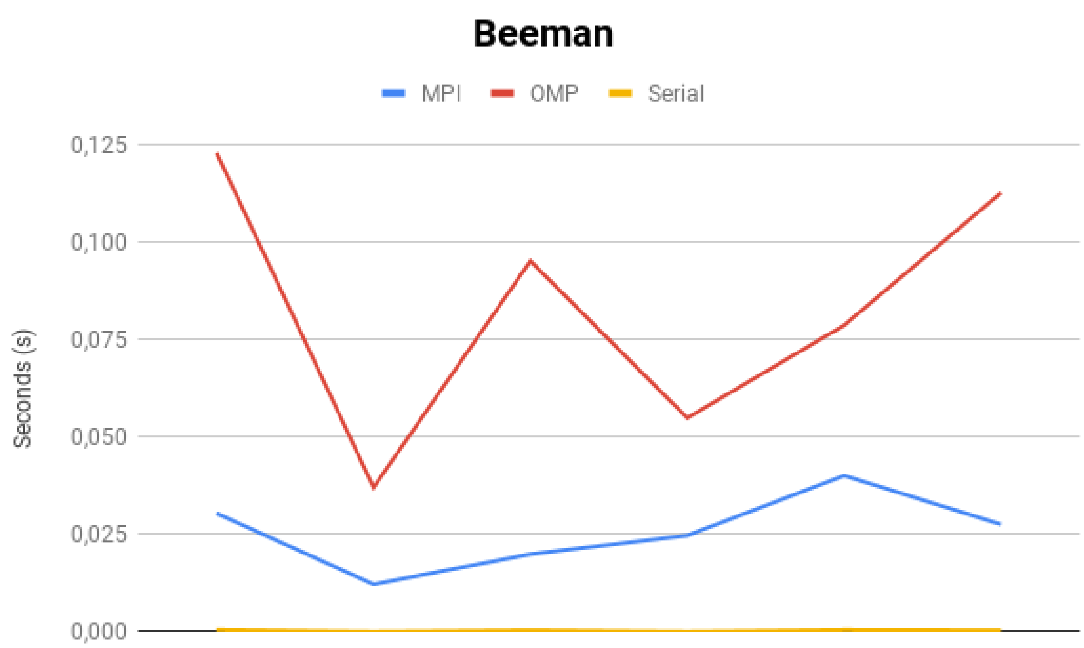

Figure 1: Beeman method with MPI, OpenMP and serial algorithms.

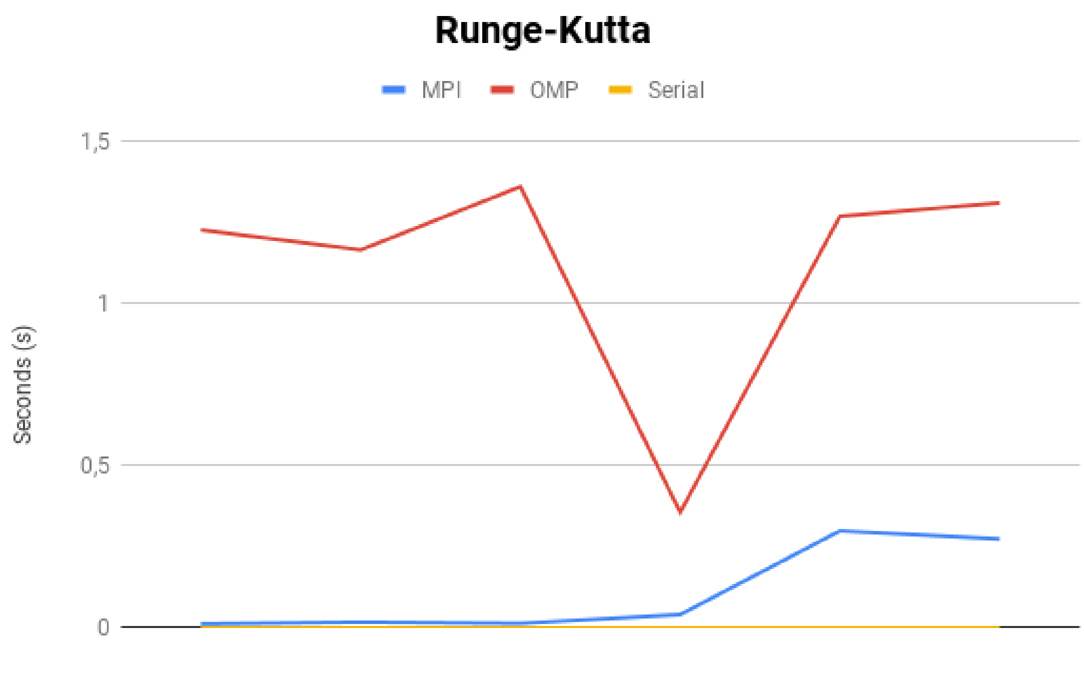

Figure 2: Runge-Kutta method with MPI, OpenMP and serial algorithms.

On the other hand, comparing Runge-Kutta with respect to Beeman, fig. 3, 4 and 5 show that Runge-Kutta method is faster than Beeman method in some points of the serial version. That behaviour is reversed on the parallel versions, in which for MPI and OpenMP, Beeman method is much faster than Runge-Kutta. It makes sense because for Runge-Kutta four components are computed on every iteration, meanwhile for Beeman only one component is computed.

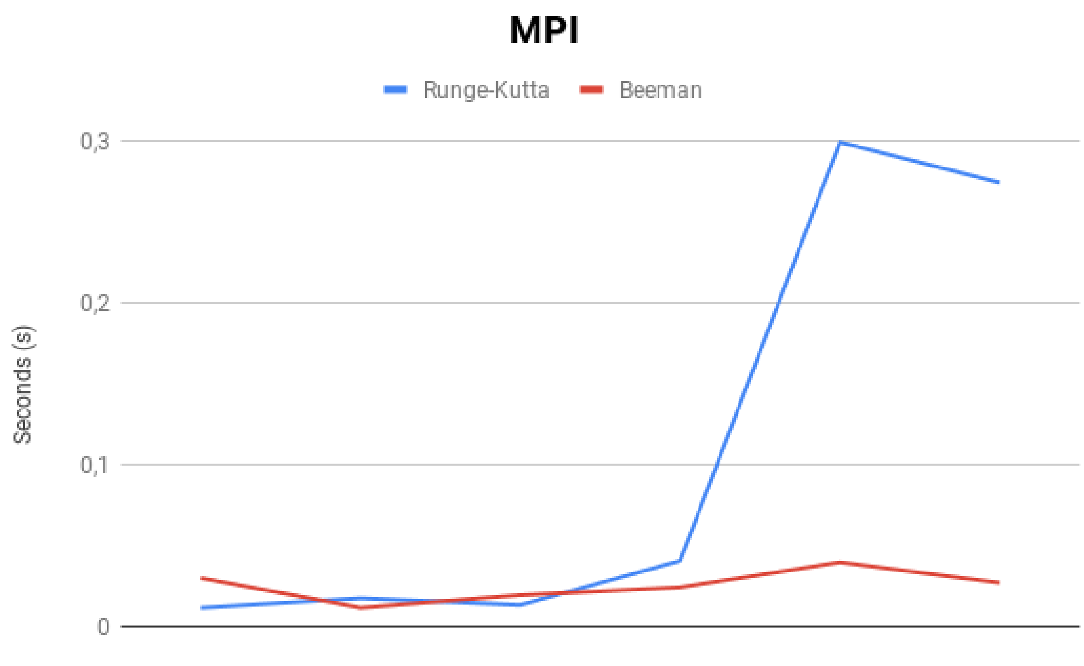

Figure 3: Comparison in time between Runge-Kutta and Beeman methods using MPI.

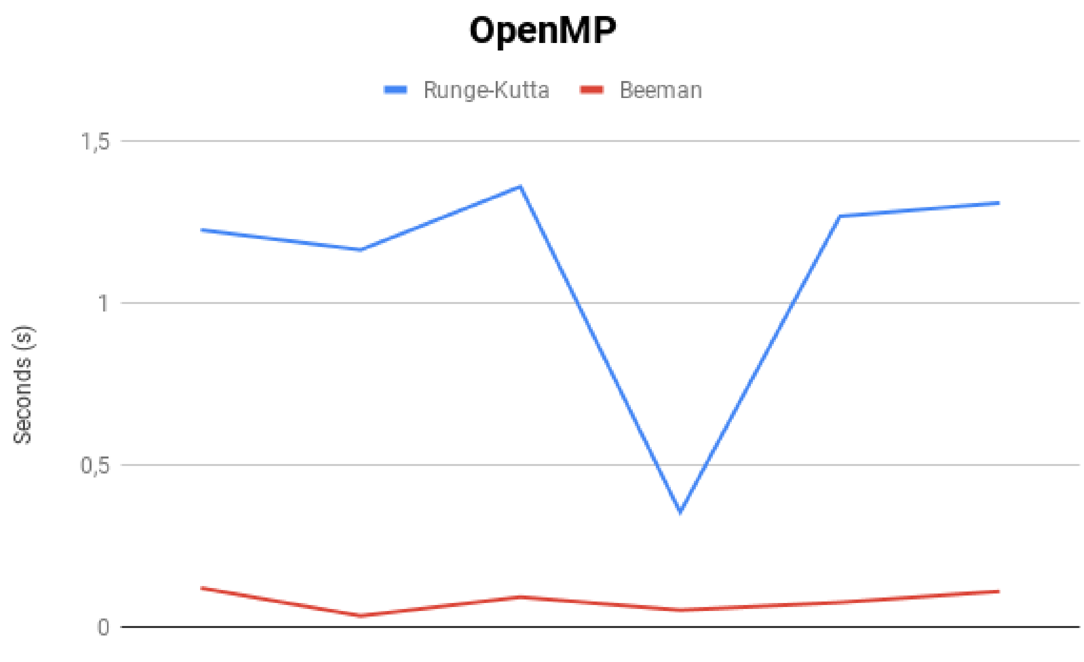

Figure 4: Comparison in time between Runge-Kutta and Beeman methods using OpenMP.

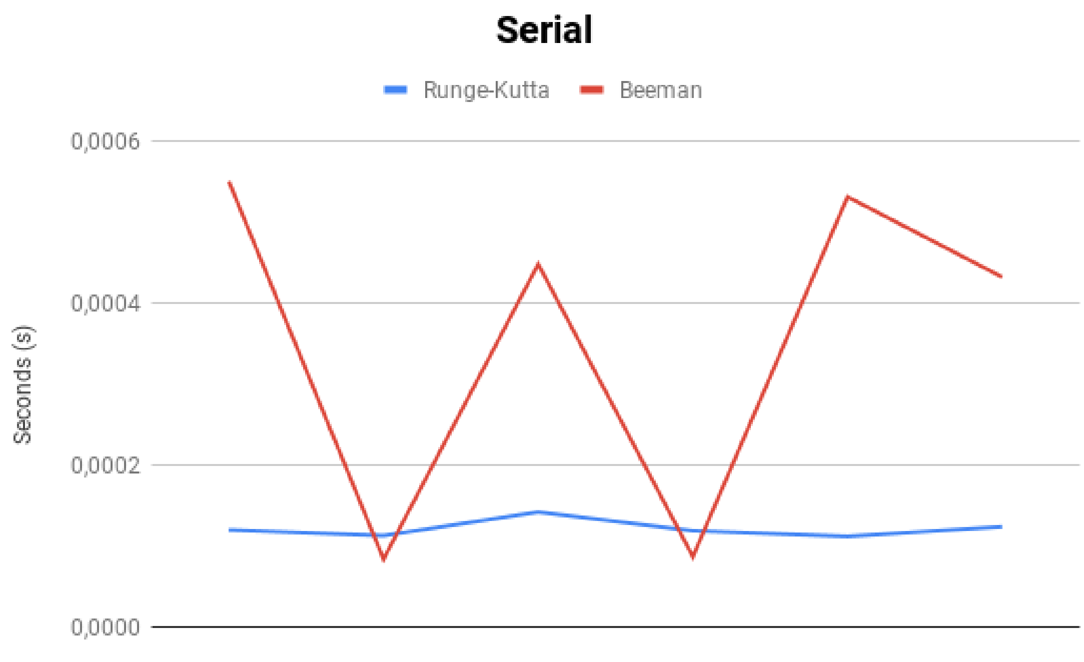

Figure 5: Comparison in time between Runge-Kutta and Beeman methods in serial version.

4 Conclusions and Recommendations

Free kick problem is one type parabolic movement and can be modeled using integrators such as Runge-Kutta and Beeman methods. However, their parallel version does not offer too much difference with respect to serial version. The problem is the few data that the problem manages. Despite of that, we could see that Beeman is faster than Runge-Kutta for obvious reasons. For future works, it is recommended to work with problems that deals with more data, so that parallelizing them can provide significant change of computing time. The algorithms used for this problem are suitable for other similar problems of [2], such as: Golf, Basketball, and projectiles in general. Also some other problems that involve parabolic movement can be modeled with the codes used in this problem.

References

[1] P. Chai and E. Anzalone. Numerical integration techniques in orbital mechanics applications, 2015.

[2] G. Palmer. *Physics for Game Programmers*. Apress, 2005.

Fernando Patricio Zhapa Camacho
Signature of the student