Ampliación de Matemáticas
Master Universitario en Sistemas Espaciales - ETSIAE

Weekly milestones. First semester.

- **Milestone 1 : Prototypes to integrate orbits without functions.**

  1. Write a script to integrate Kepler orbits with an Euler method.
  2. Write a script to integrate Kepler orbits with a Crank-Nicolson method.
  3. Write a script to integrate Kepler orbits with a Runge–Kutta fourth order.
  4. Change time step and plot orbits. Discuss results.

- **Milestone 2 : Prototypes to integrate orbits with functions.**

  1. Write a function called Euler to integrate one step. The function $F(U,t)$ of the Cauchy problem should be input argument.
  2. Write a function called Crank_Nicolson to integrate one step.
  3. Write a function called RK4 to integrate one step.
  4. Write a function called Inverse_Euler to integrate one step.
  5. Write a function to integrate a Cauchy problem. Temporal scheme, initial condition and the function $F(U,t)$ of the Cauchy problem should be input arguments.
  6. Write a function to express the force of the Kepler movement. Put emphasis on the way the function of the Cauchy problem is written:
     $F = [\dot{r}, -r/|r|^3]$ where $r, \dot{r} \in \mathbb{R}^2$
  7. Integrate a Kepler with these latter schemes and explain the results.
  8. Increase and decrease the time step and explained the results.

- **Milestone 3 : Error estimation of numerical solutions.**

  1. Write a function to evaluate errors of numerical integration by means of Richardson extrapolation. This function should be based on the Cauchy problem solution implemented in milestone 2.
  2. Numerical error or different temporal schemes: Euler, Inverse Euler, Crank Nicolson and fourth order Runge Kutta method.
  3. Write a function to evaluate the convergence rate of different temporal schemes.
  4. Convergence rate of the different methods with the time step.

- **Milestone 4 : Linear problems. Regions of absolute stability.**

  1. Integrate the linear oscillator $\ddot{x} + x = 0$ with some initial conditions. Use Euler, Inverse Euler, Leap–Frog, Crank–Nicolson and fourth order Runge Kutta method.
  2. Regions of absolute stability of the above methods.
  3. Explain the numerical results based on regions of absolute stability.

- **Milestone 5 : N body problem.**

  1. Write a function to integrate the N body problem.
  2. Simulate an example and discuss the results.

- **Milestone 6 : Lagrange points and their stability.**

  1. Write a high order embedded Runge-Kutta method.
  2. Write function to simulate the circular restricted three body problem.
  3. Determination of the Lagrange points $F(U) = 0$.
  4. Stability of the Lagrange points: L_1, L_2, L_3, L_4, L_5.
  5. Orbits around the Lagrange points by means of different temporal schemes.

- **Milestone 7 : Orbits of the circular restricted three body problem.**

  1. Investigate existing temporal schemes to integrate Cauchy problems. Fortran and Python programming codes. Use ODE integrators of the library scipy (odeint, solve_ivp). LSODA, ODEPACK, ODEX.
  2. Integrate Arenstorf's periodic orbit. Compare results among GBS, RK and AM methods.
  3. Stability of Lyapunov orbits. Shampine and Gordon orbits.
  4. Error tolerance and computational effort.
  5. Mixing Fortran and Python: calling Fortran from Python and viceversa.
  6. Parallel programming with GPUs. N-body problem.
  7. Standalone Python codes.
  8. Automatic testing with GitHub.
  9. Mixing Fortran, C++ and Python in one executable file.
  10. Python script for GMAT.