Dinámica Molecular

Grupo 6:

Grethe, Juan - 57370

Grabina, Martin - 57360

Fundamentos

Fundamentos

Oscilador Amortiguado

$$f = ma = mr_2 = -kr - \gamma r_1$$

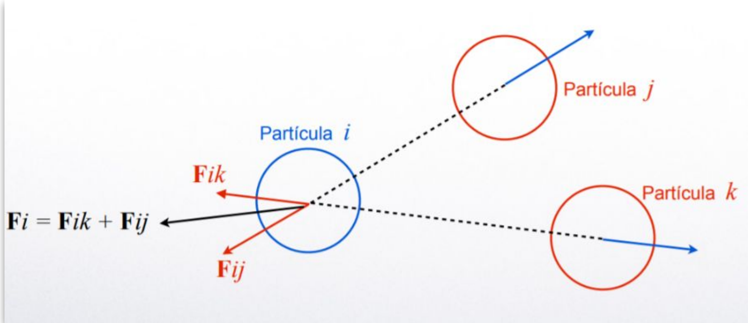
$$r = A\, exp(-(\frac{\gamma}{2m})t) \quad cos((\frac{k}{m} - \frac{\gamma^2}{4m^2})^{0.5} \quad t)$$

$$\mathrm{ECM} = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2.$$

Gas de Lennard Jones

$$F(r) = \frac{12\epsilon}{r_m}\left[\left(\frac{r_m}{r}\right)^{13} - \left(\frac{r_m}{r}\right)^{7}\right]$$

$$V_{LJ} = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] = \epsilon\left[\left(\frac{r_m}{r}\right)^{12} - 2\left(\frac{r_m}{r}\right)^{6}\right] \qquad E_c = \frac{mv^2}{2}$$



Integradores

Beeman

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

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

Velocity Verlet

$$\mathbf{r}_i(t+\Delta t)=\mathbf{r}_i(t)+\Delta t\ \mathbf{v}_i(t)+\frac{\Delta t^2}{m_i}\ \mathbf{f}_i(t)+\mathcal{O}(\Delta t^3),$$

$$\mathbf{v}_i(t+\Delta t)=\mathbf{v}(t)+\frac{\Delta t}{2m_i}(\mathbf{f}_i(t)+\mathbf{f}_i(t+\Delta t))+\mathcal{O}(\Delta t^2).$$

Gear Predictor Order 5

$$\mathbf{r}^p(t+\Delta t)=\mathbf{r}(t)+\mathbf{r}_1(t)(\Delta t)+\mathbf{r}_2(t)\frac{(\Delta t)^2}{2!}+\mathbf{r}_3(t)\frac{(\Delta t)^3}{3!}+\mathbf{r}_4(t)\frac{(\Delta t)^4}{4!}+\mathbf{r}_5(t)\frac{(\Delta t)^5}{5!}$$

$$\mathbf{r}_1^p(t+\Delta t)=\mathbf{r}_1(t)+\mathbf{r}_2(t)(\Delta t)+\mathbf{r}_3(t)\frac{(\Delta t)^2}{2!}+\mathbf{r}_4(t)\frac{(\Delta t)^3}{3!}+\mathbf{r}_5(t)\frac{(\Delta t)^4}{4!}$$

$$\mathbf{r}_2^p(t+\Delta t)=\mathbf{r}_2(t)+\mathbf{r}_3(t)(\Delta t)+\mathbf{r}_4(t)\frac{(\Delta t)^2}{2!}+\mathbf{r}_5(t)\frac{(\Delta t)^3}{3!}$$

$$\mathbf{r}_3^p(t+\Delta t)=\mathbf{r}_3(t)+\mathbf{r}_4(t)(\Delta t)+\mathbf{r}_5(t)\frac{(\Delta t)^2}{2!}$$

$$\mathbf{r}_4^p(t+\Delta t)=\mathbf{r}_4(t)+\mathbf{r}_5(t)(\Delta t)$$

$$\mathbf{r}_5^p(t+\Delta t)=\mathbf{r}_5(t)$$

$$\Delta\mathbf{a}=\Delta\mathbf{r}_2=\mathbf{a}(t+\Delta t)-\mathbf{a}^p(t+\Delta t)=\mathbf{r}_2(t+\Delta t)-\mathbf{r}_2^p(t+\Delta t)$$

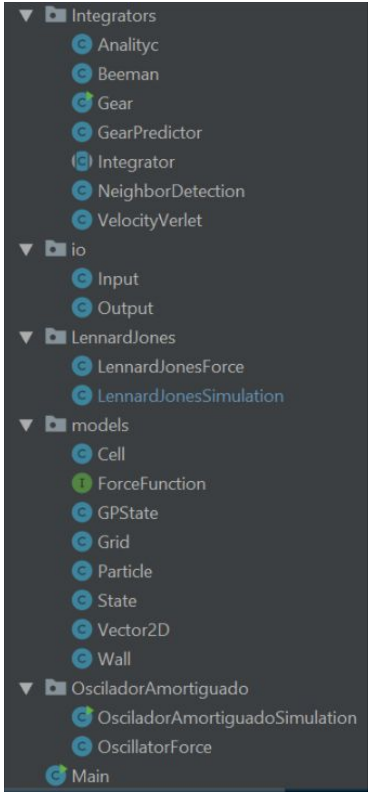
$$\Delta\mathbf{R2}=\frac{\Delta\mathbf{a}(\Delta t)^2}{2!}$$

$$\mathbf{r}_q^c=\mathbf{r}_q^p+\alpha_q\Delta\mathbf{R2}\frac{q!}{(\Delta t)^q}$$

Implementación

Implementación - Características

- Java 8 (Maven, Parallel)
- OOP
  - Integrators (Vector2D)
  - IO
  - Lennard Jones
  - Oscillator
  - Models
- Ovito



Implementación

- Partículas puntuales
- Masa = 0.1 Kg
- Velocidad inicial = 10 m/s
- Ángulo de velocidad random
- Separadas por 5 unidades

Implementación - E/S

- Entrada
  - Parámetros definidos
- Salida
  - positions.xyz
  - Archivos csv de estadística

Implementación - Código

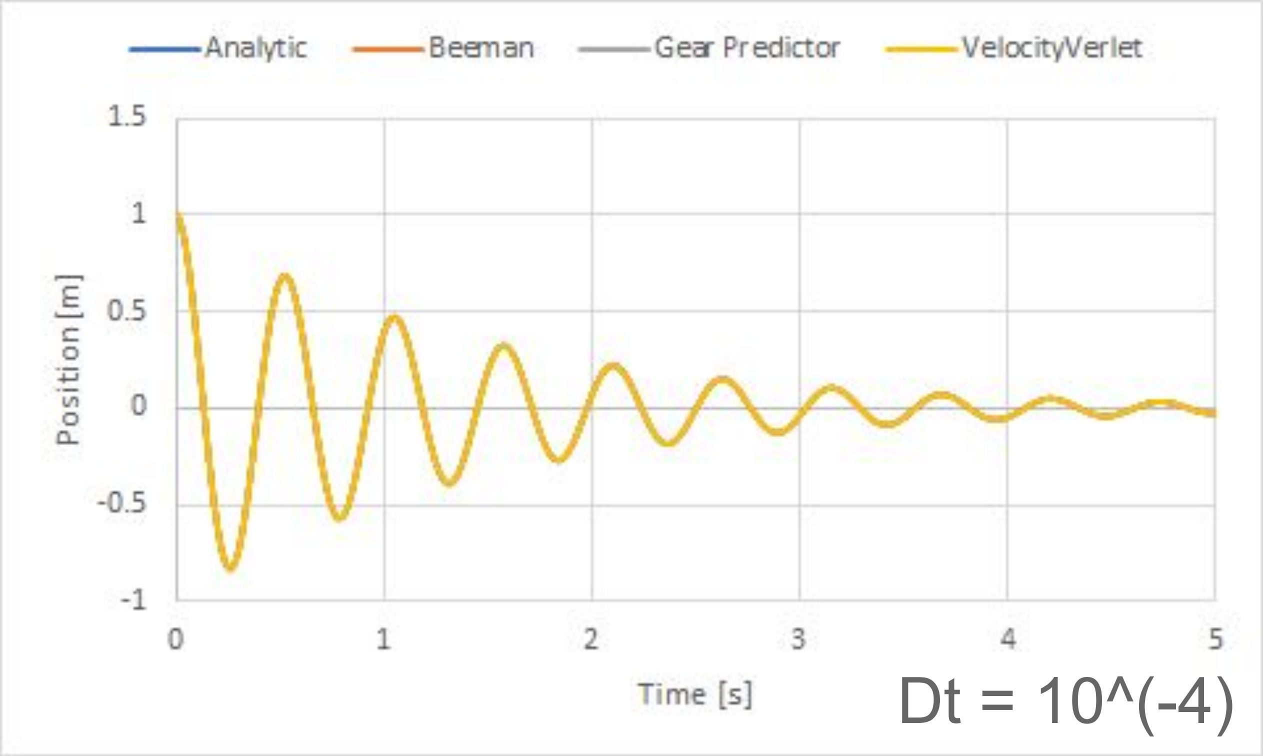
```
OscillatorForce oscillatorForce = new OscillatorForce(input.getY(), input.getK());

for (double diferential_t : diferentials){
    int index = diferentials.indexOf(diferential_t);

    analitycPositions[index] = oscillation(new Analityc(diferential_t, oscillatorForc
    beenmanPositions[index] = oscillation(new Beeman(diferential_t, oscillatorForce),
    gearPredictorPositions[index] = oscillation(new GearPredictor(diferential_t, osci
    verletPositions[index] = oscillation(new VelocityVerlet(diferential_t, oscillator

    beenmanError[index] = meanSquaredError(analitycPositions[index], beenmanPositions
    gearPredictorError[index] = meanSquaredError(analitycPositions[index], gearPredic
    verletError[index] = meanSquaredError(analitycPositions[index], verletPositions[i
}
Output.printOscillationsResults(analitycPositions,beenmanPositions,gearPredictorPosit
```

Implementación - Código

```
public void simulate(double dt) throws IOException {

    double time = 0.0;
    int iteration = 0;
    List<Particle> particles = input.getParticles();
    int leftParticles=1000;
    long start = System.currentTimeMillis();
    while (leftParticles>75) {
        Grid grid = new Grid(input.getCellSideQuantity(), input.getSystemSideLength());
        grid.setParticles(particles);
        Map<Particle, List<Particle>> neighbours = NeighborDetection.getNeighbors(grid, grid.getUsedCe

        double auxtime = time;
        neighbours.entrySet().stream().parallel().forEach(particle -> move(particle.getKey(), particle

        particles.stream().parallel().forEach(Particle::updateState);
        if (iteration % 10000 == 0){
            leftParticles = Output.printParticle(particles, time: ((int)(time*10))/10.0);
            System.out.println(time +" " + leftParticles );
            Output.printToFile(particles);
        }
        if (((int)((time % 0.1)*100000)) == 0) {
            Output.printVelocities(particles, time);
        }
        time += dt;
        iteration++;
    }
    System.out.println(System.currentTimeMillis() - start);
```

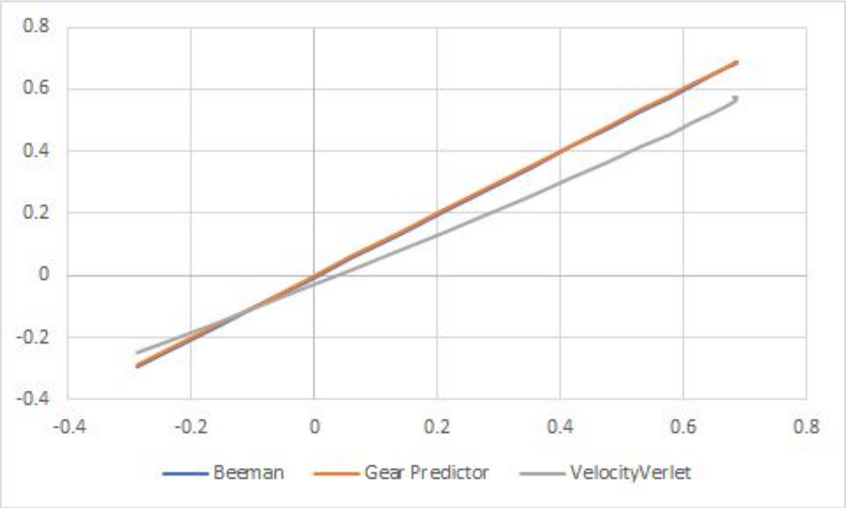
Resultados

Oscilador Amortiguado - Trayectoria

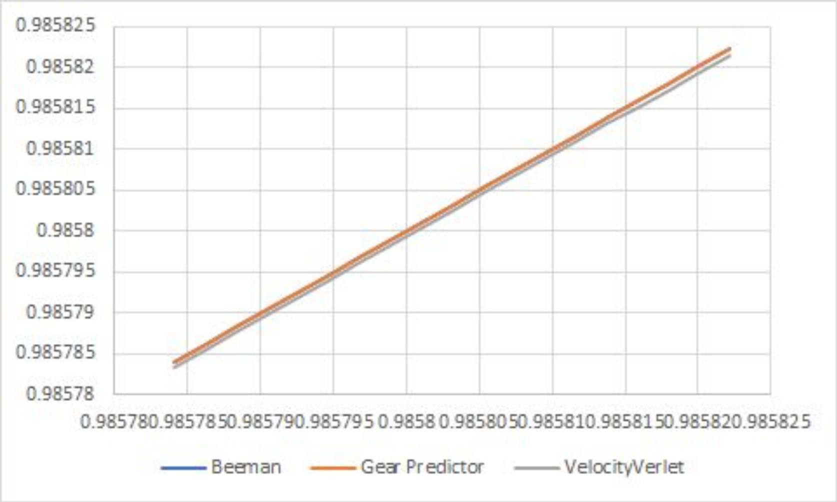


Oscilador Amortiguado - Detalle

Dt = 10^(-2)



Dt = 10^(-6)

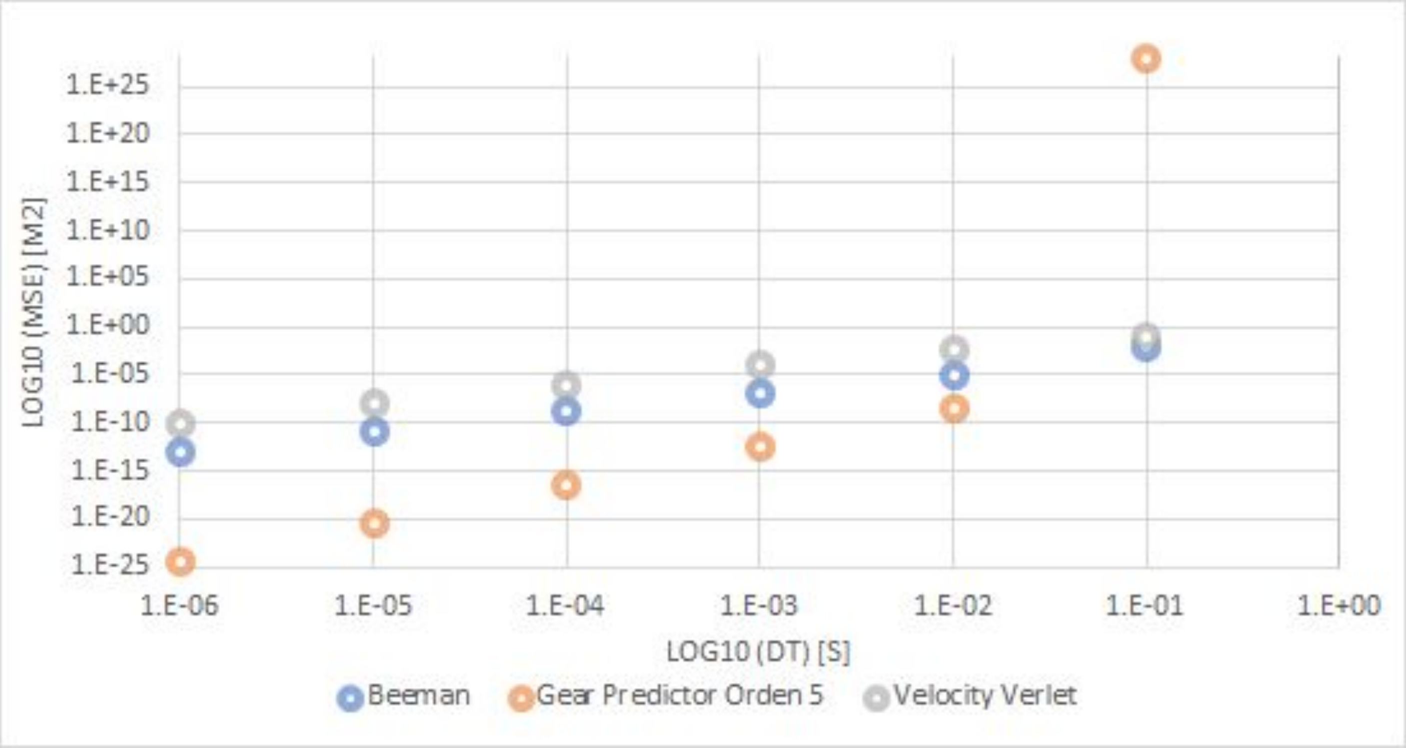


Posición [m]

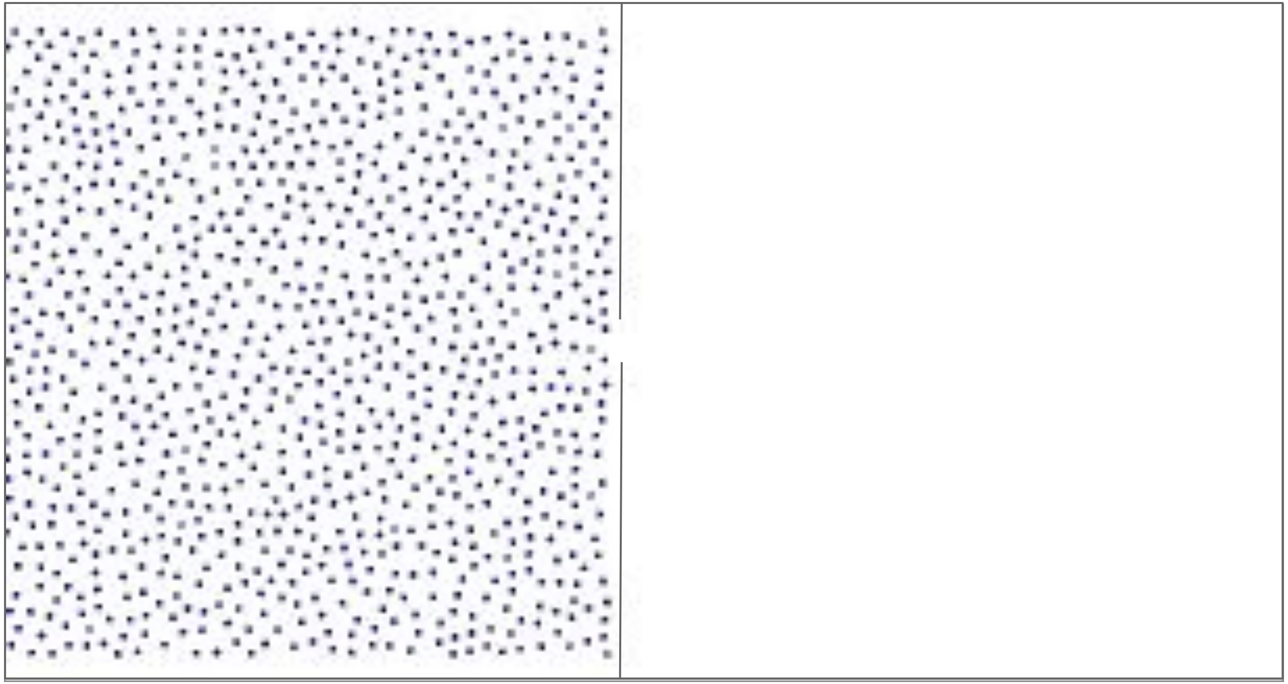
Tiempo [s]

Oscilador Amortiguado

ECM en función al DT por Integrador



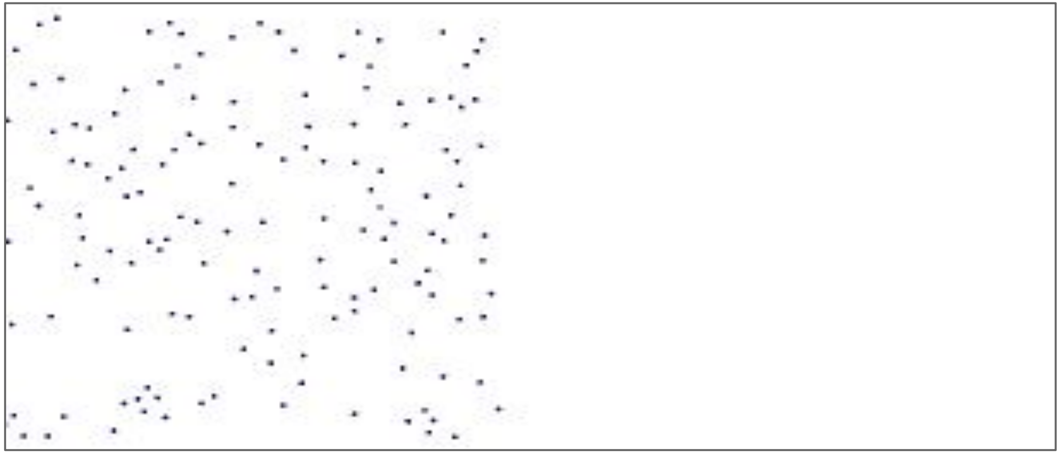
Resultados - Animaciones



1000 partículas

Primeros 250 segundos

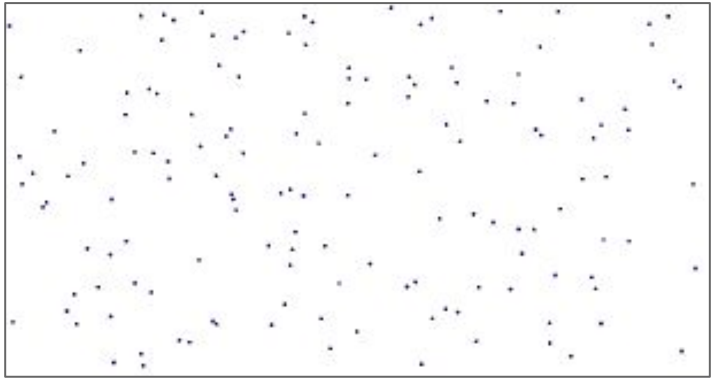
Resultados - Animaciones



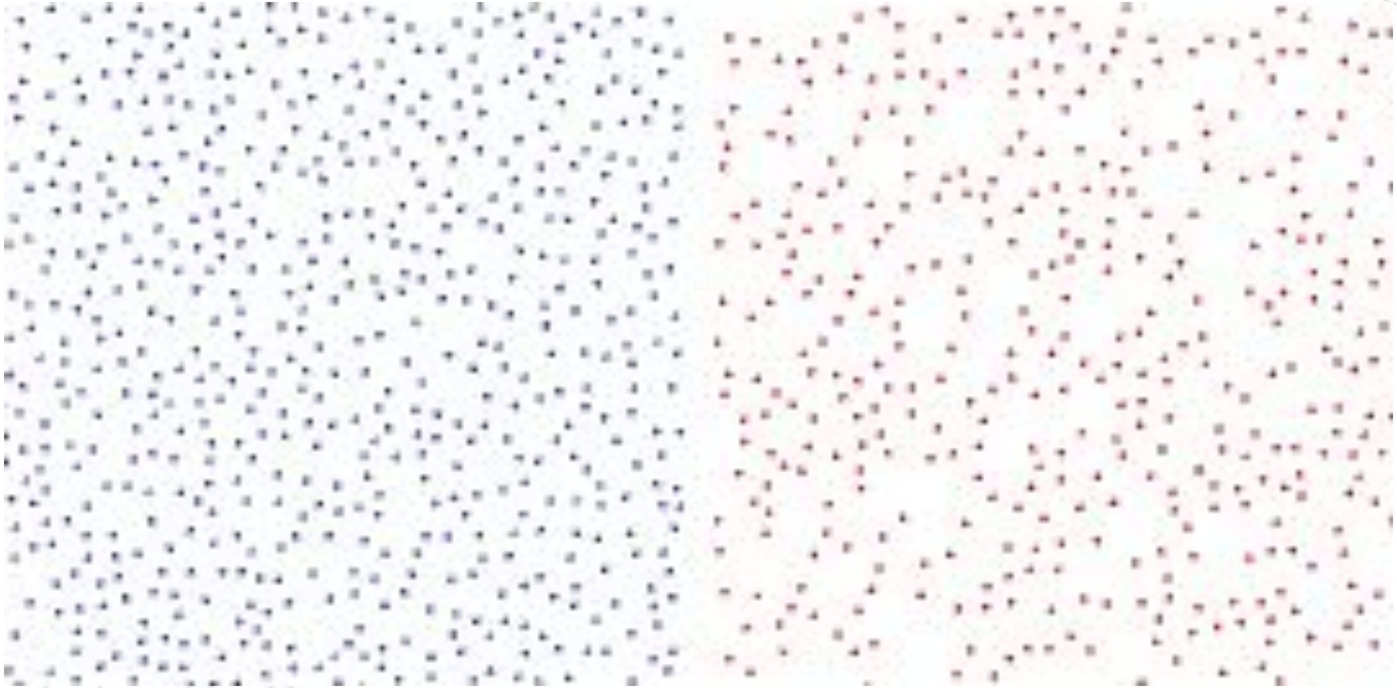
N=150
Dt=0.0001

Inicio

Cerca del equilibrio

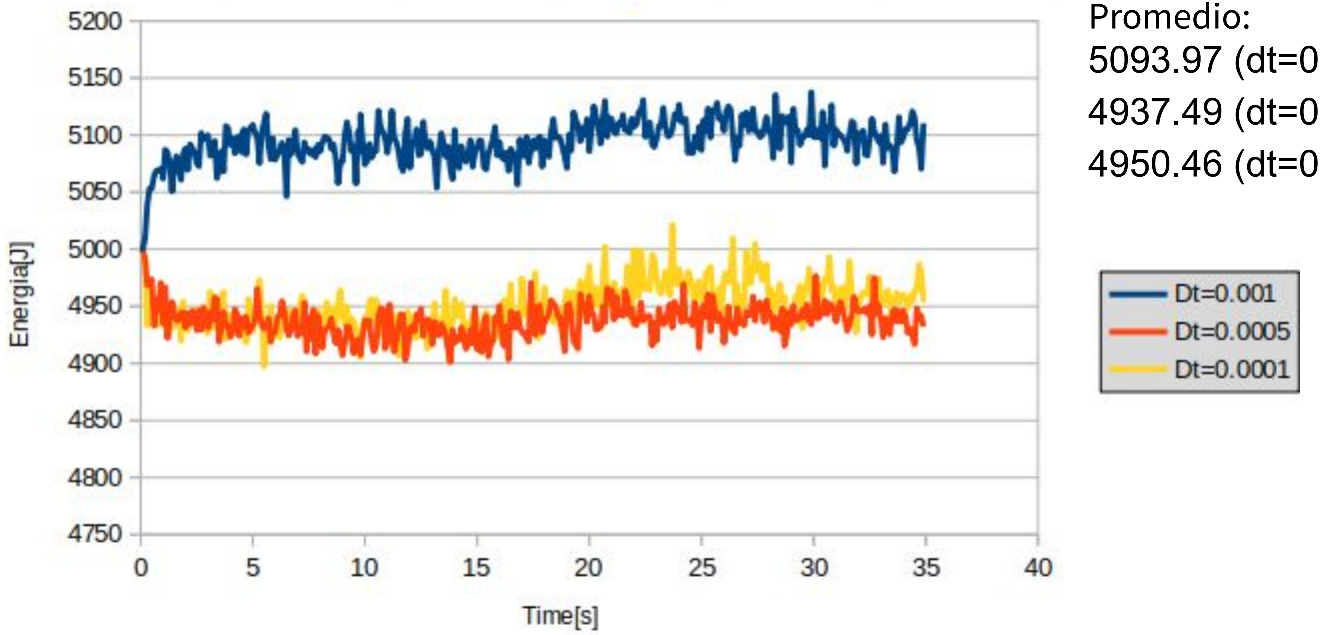


Resultados - Animación [N=1000, Dt=0.0001]



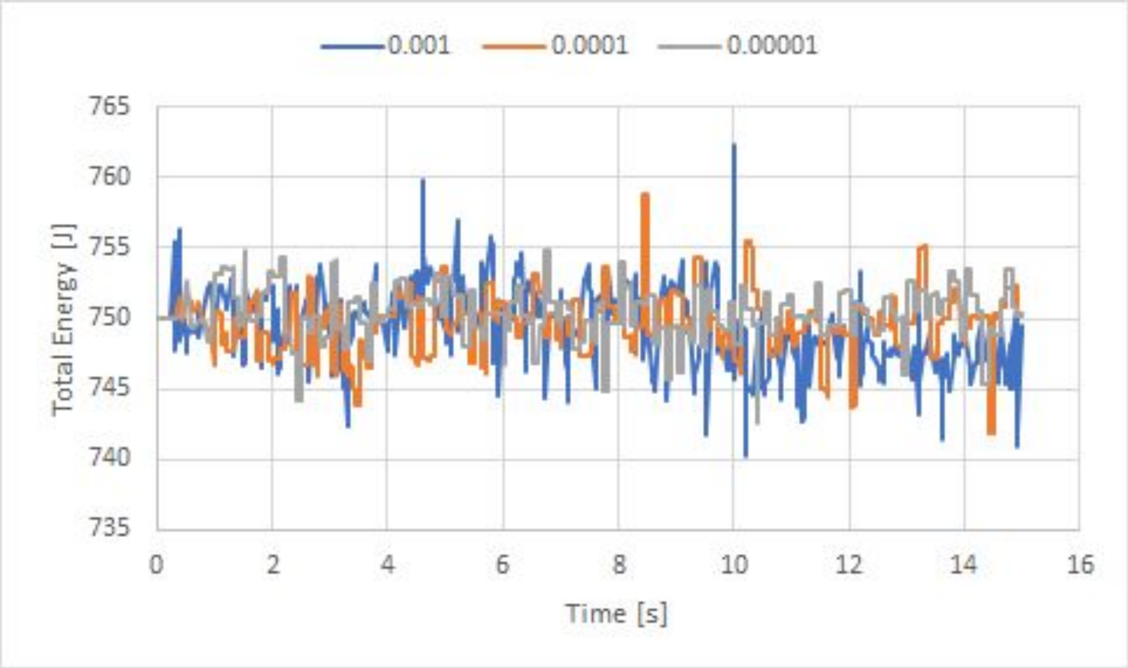
Final

Resultados - Evolución Energía para 1000 partículas



Promedio:
5093.97 (dt=0.001)
4937.49 (dt=0.0005)
4950.46 (dt=0.0001)

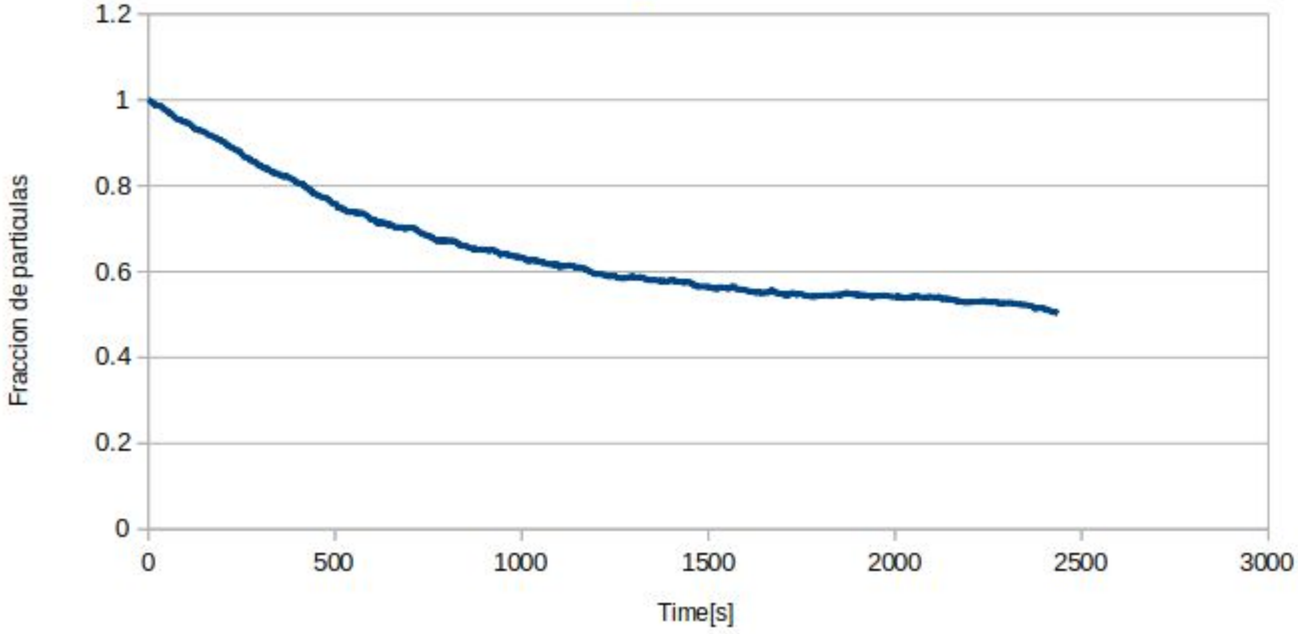
Resultados - Evolución de la energía para 150 partículas



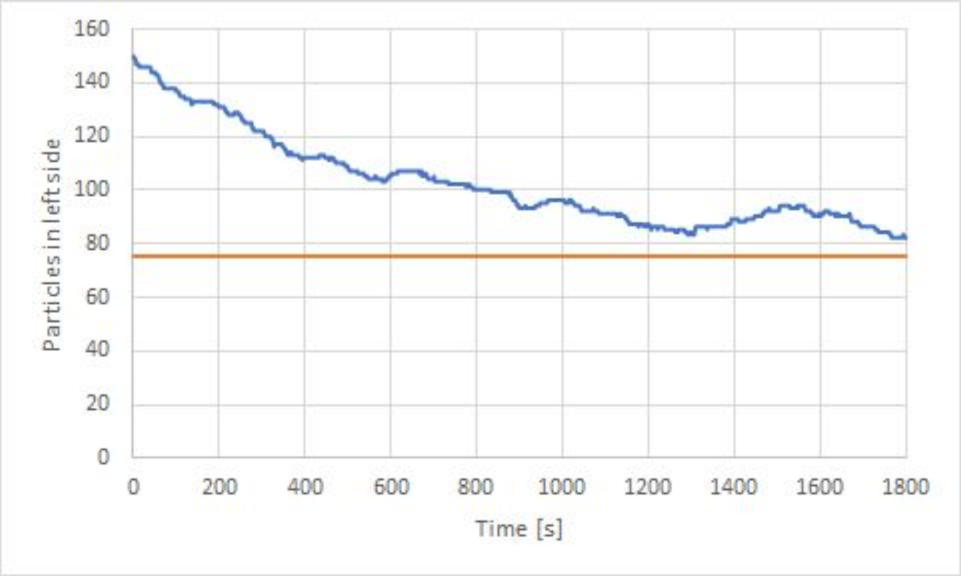
	Promedio	Variación
0.001	749.4182	0.581779
0.0001	749.6598	0.340164
0.00001	750.2945	0.294534

Resultados - Cantidad de partículas en lado izquierdo

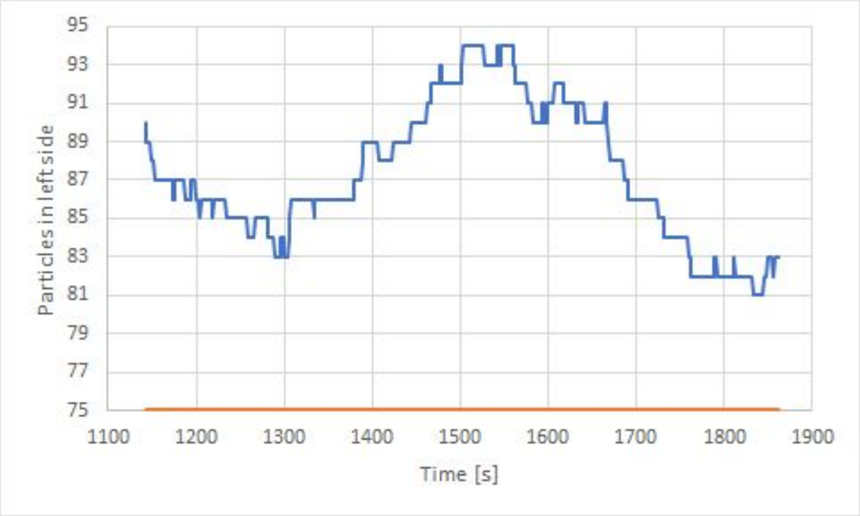
N=1000
dt=0.0001



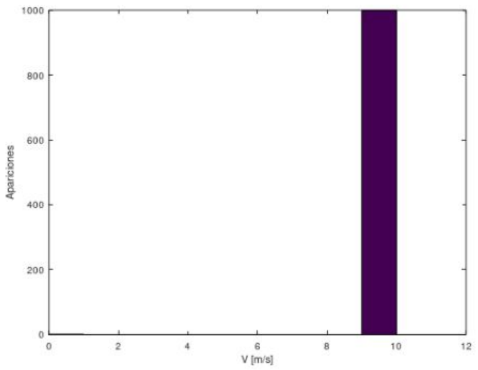
Resultados - Cantidad de partículas en lado izquierdo



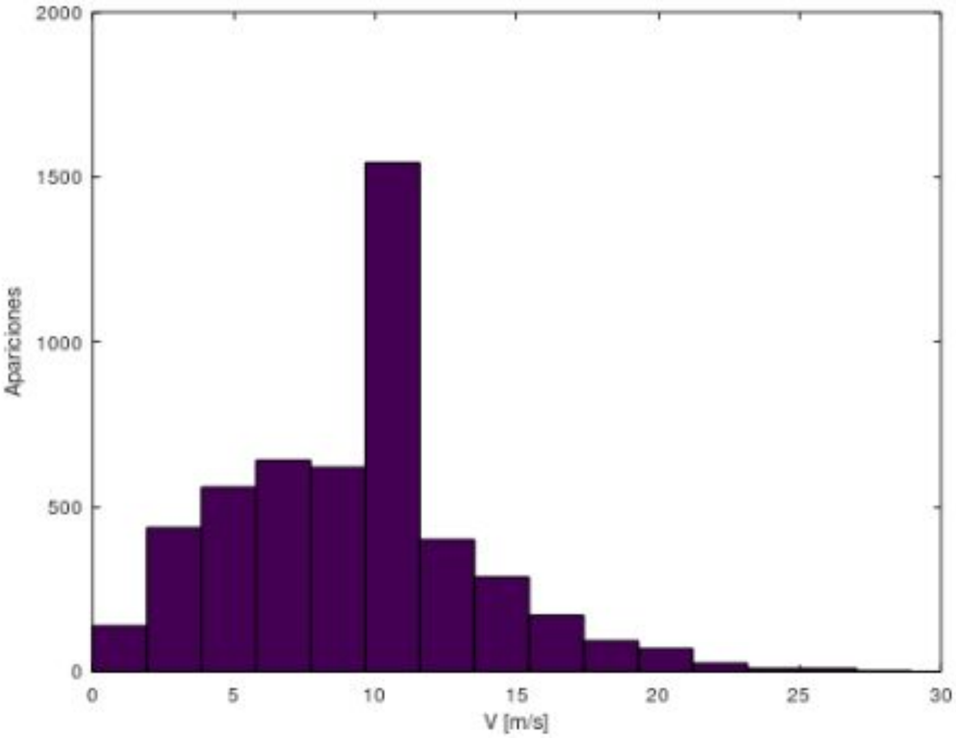
N=150
Dt=0.0001



Resultados - Distribución de velocidades 1000 partículas

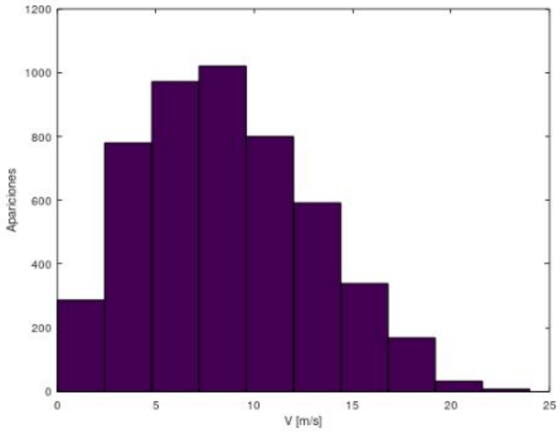


Inicial

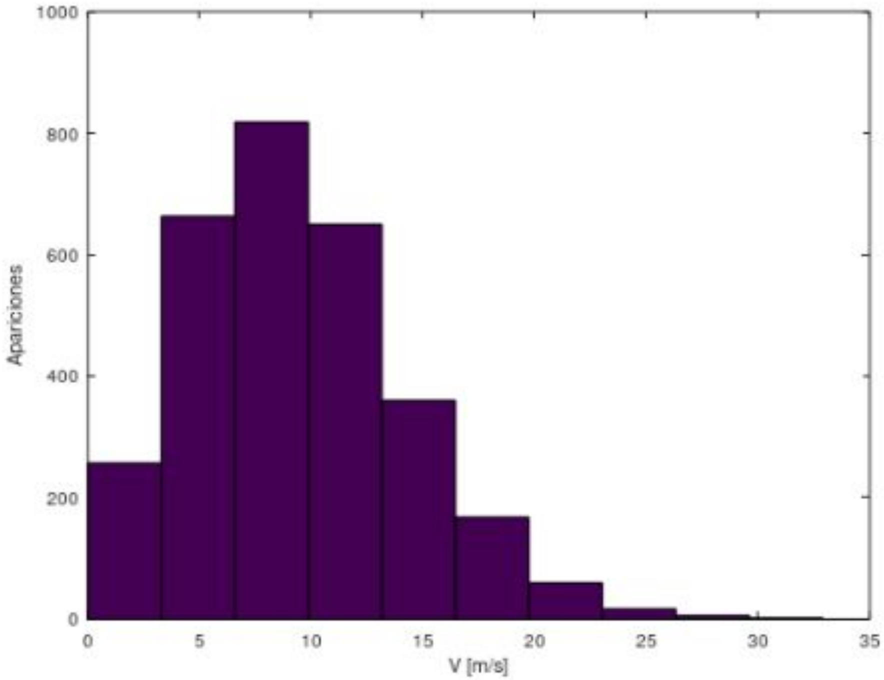


0-250 segundos

Resultados - Distribución de velocidades con 1000 partículas

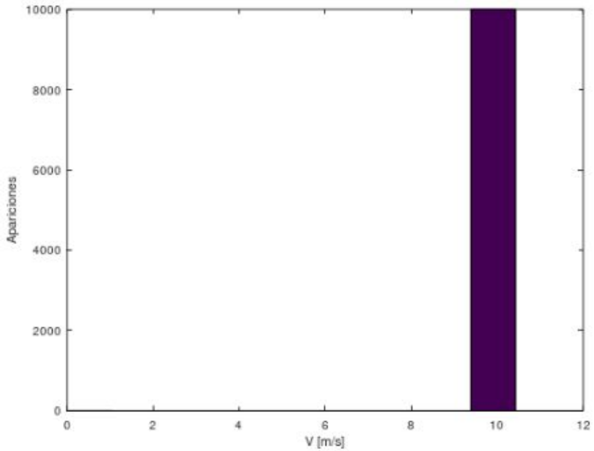


500 a 800 segundos

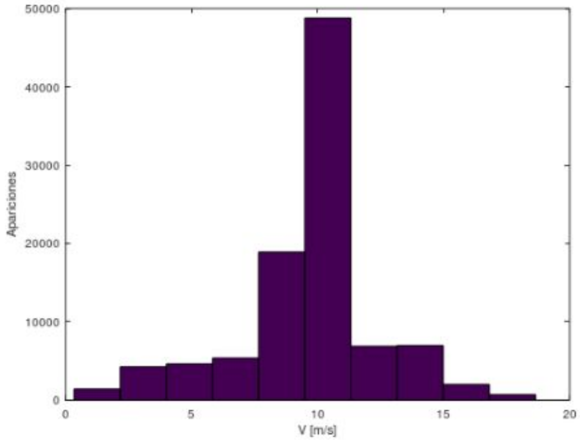


Final

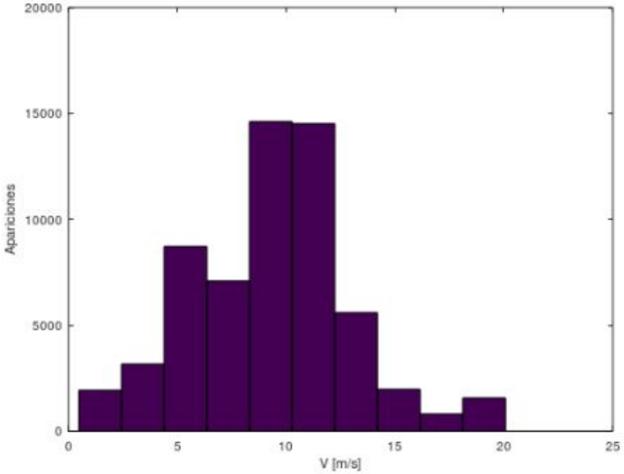
Resultados - Distribución de velocidades [N=150;Dt=0.001]



Inicio

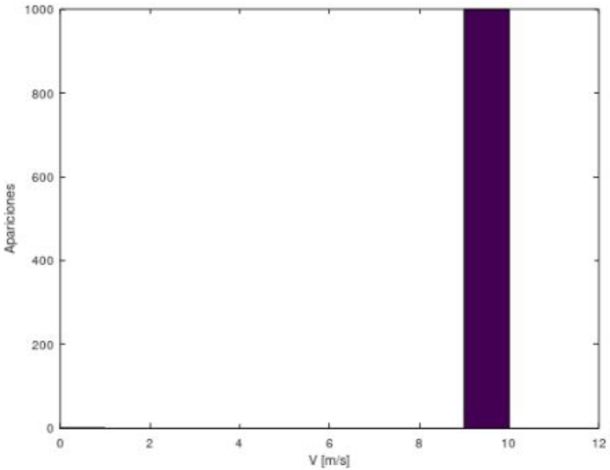


Mitad

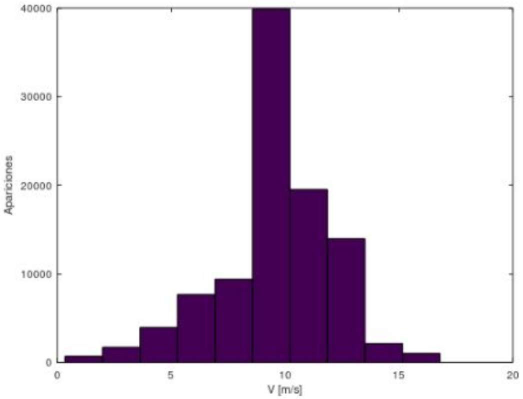


Final

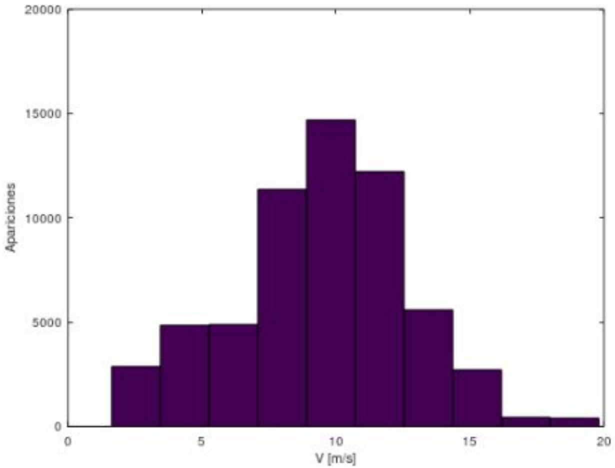
Resultados - Distribución de velocidades [N=150;Dt=**0.0001**]



Inicio



Mitad



Final

Conclusiones

Conclusiones - Oscilador

→ Para dt < 0.01 Gear Prediction es integrador más preciso.

→ A medida que disminuye el dt, disminuye el ECM con respecto a la solución analitica.

Conclusiones - Gas de Lennard-Jones

→ El sistema “oscila amortiguadamente” cerca de ft~0.5 con un 5%.N de amplitud aproximado.

→ A menor densidad de partículas, es mayor el tiempo en que tarda en pasar cada partícula al otro lado.

→ A mayor densidad, se requiere un dt mas chico.

→ La distribución de velocidades no depende del dt, y tiende a ser una distribución más homogénea que la inicial.

“

Muchas Gracias

¿Preguntas?