

Simulación Monte Carlo de un Detector de Centelleo Plástico con Fotomultiplicador: Análisis del ruido estadístico y eficiencia de atrapamiento de luz

Ana María Játiva González

CO-autores:

Gabriel Andres Tarazona Ramírez

Mariana Roa Gomez

Indice

1. Motivación
2. Marco Teorico
 - Generación de fotones
 - Reflexión total interna
 - El fotomultiplicador
3. ¿Cómo funciona el detector?
4. Método Monte Carlo
5. Resultados
6. Conclusiones

Centelleadores



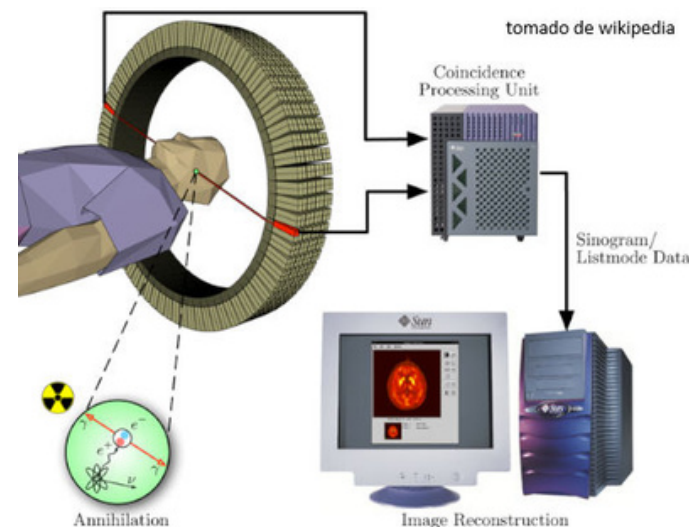
Seguridad Nuclear

Detección de materiales fisionables



Medicina

PET, SPECT, dosimetría



Física de Partículas

Experimentos LHC, neutrinos



El ruido estadístico **limita** la resolución energética.
Se busca **modelar** las fluctuaciones para optimizar el diseño.

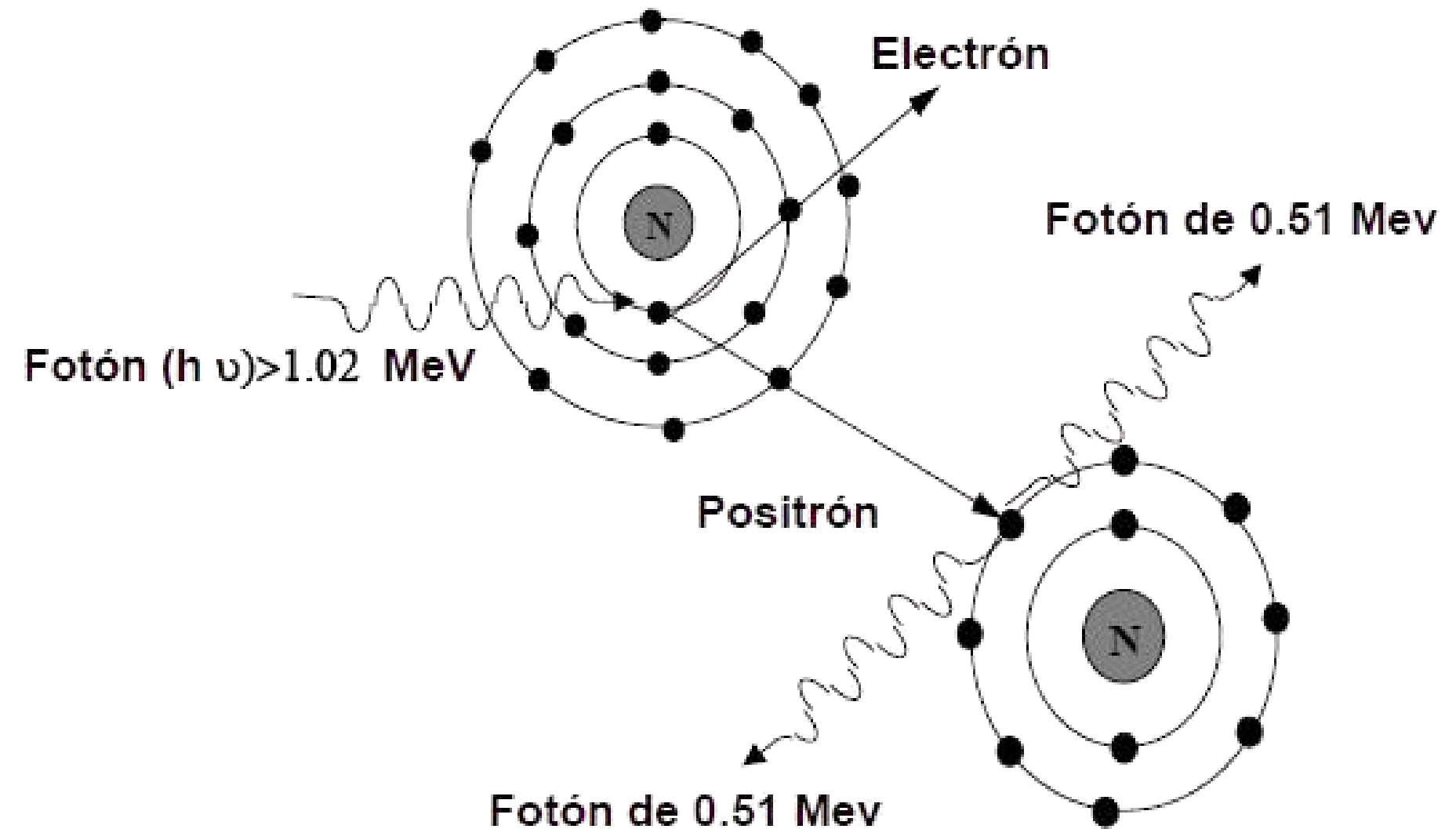
Generación de fotones

Una partícula radiactiva deposita su energía en el centelleador, esta energía promueve electrones del material a niveles de mayor energía. Al regresar a su estado fundamental, emiten un fotón de luz.

$N_{\text{fotones}} = \text{np.random.poisson}(10000)$

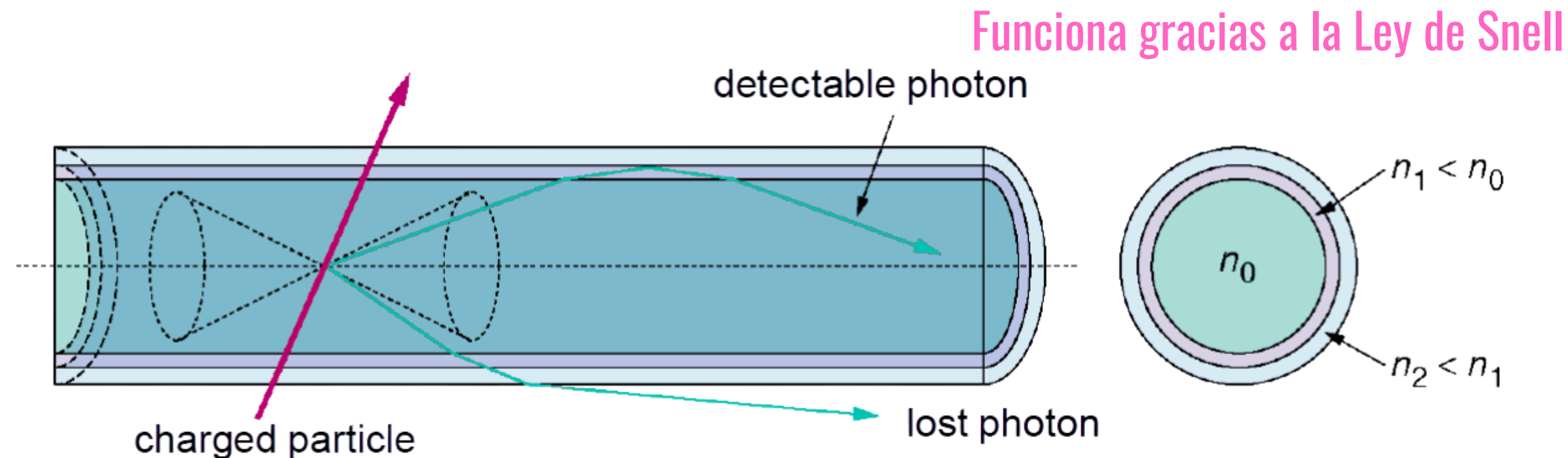
Media: $\mu = 10,000$

Desviación estándar: $\sigma = 100$



Atrapamiento de Luz

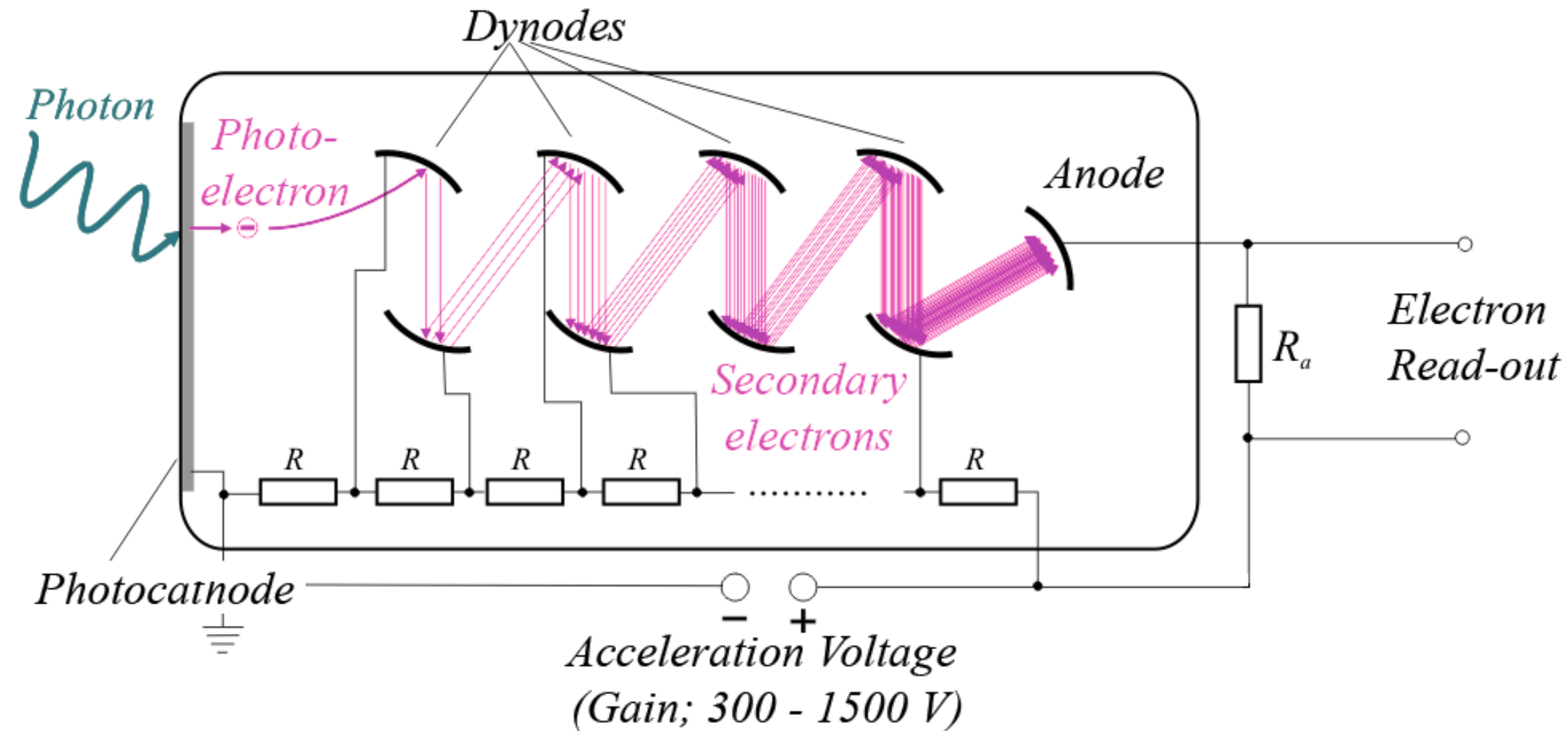
Principio óptico que consiste en confinar fotones dentro de un material mediante múltiples reflexiones, incrementando su recorrido interno para favorecer su absorción.



$$\theta_c = \arcsen(1.00 / 1.58) = \arcsen(0.6329) = 39.3^\circ$$

El fotón debe viajar hacia la base y tener un ángulo $\theta < 39.3^\circ$

Fotomultiplicador



$$QE = 25\% = 0.25$$

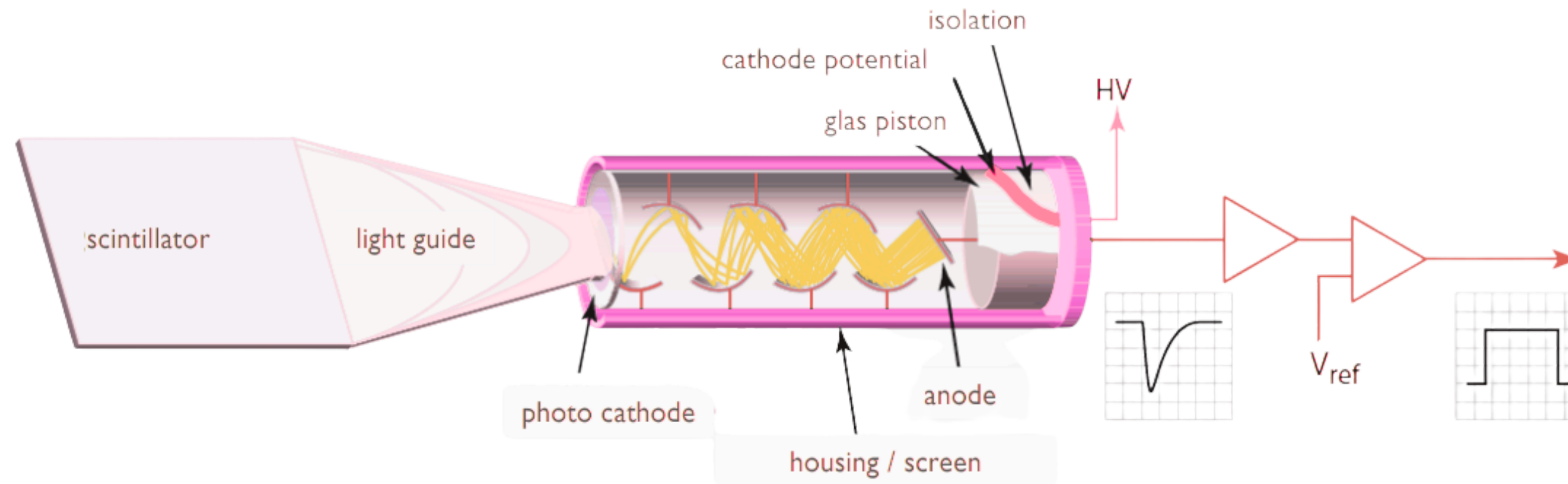
N fotoelectrones \sim Binomial(1,129, 0.25) = 282 (en promedio)

$$G = 4^{10} = 1,048,576 \approx 1 \text{ millón}$$

Electrones en el ánodo: $282 \times 1e6 \approx 294$ millones.

Carga final: $Q = N_e \cdot e = 47.36$ picocoulombs (pC).

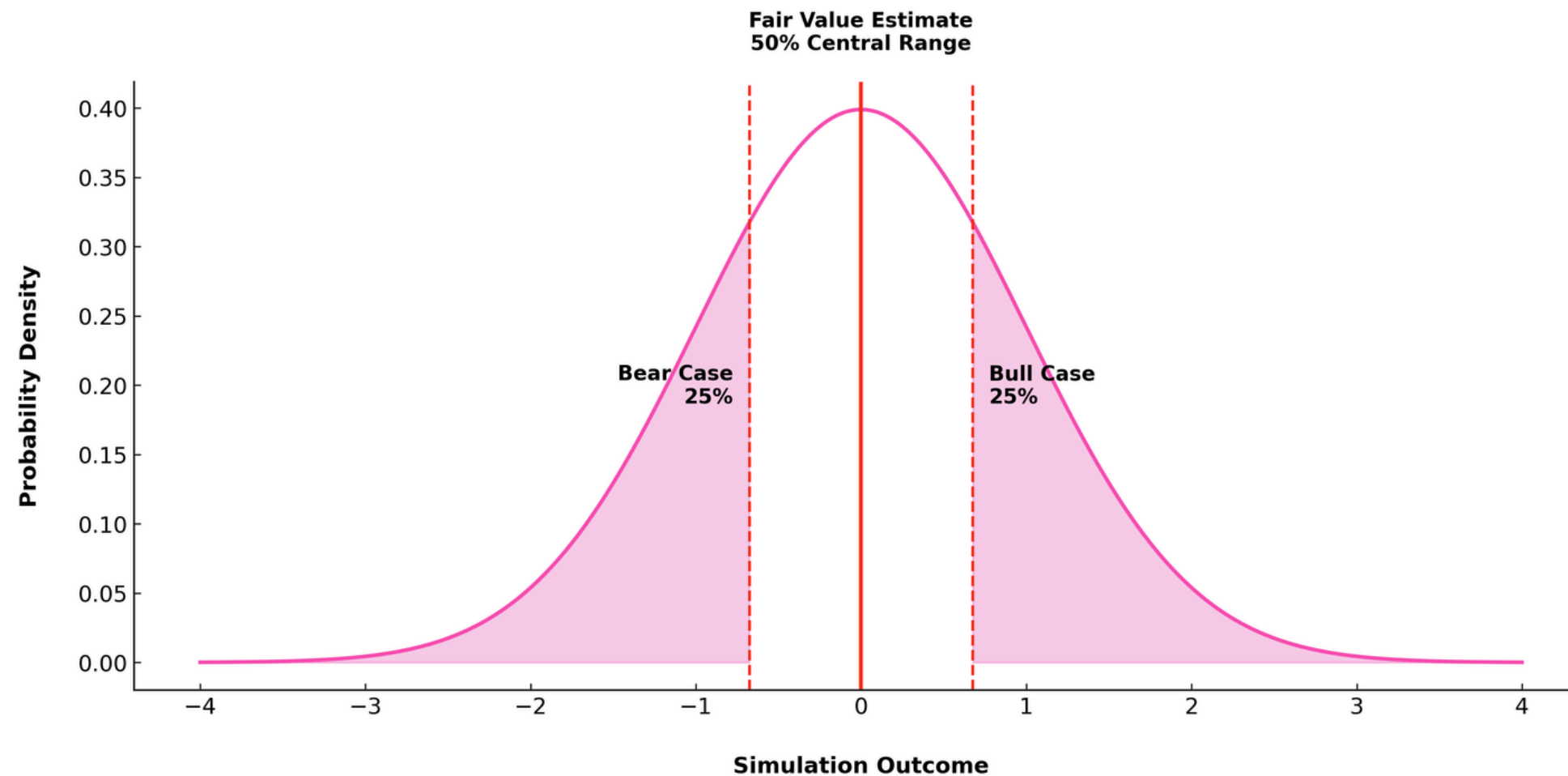
Funcionamiento de detector



Una partícula deposita energía → El centelleador emite luz → Algunos fotones son atrapados →
Se convierten en electrones → Se multiplican → Obtenemos un pulso eléctrico

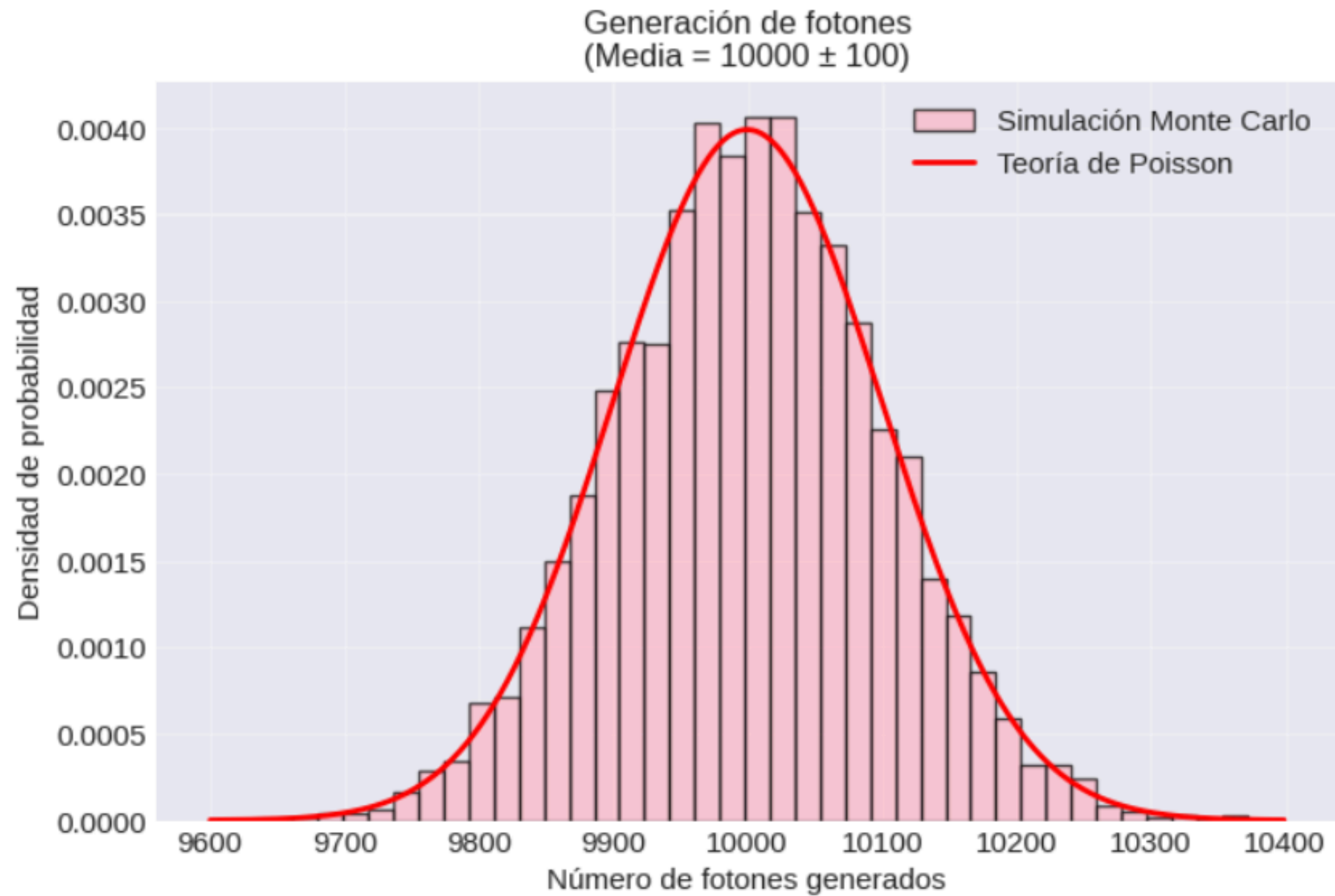
Método Monte Carlo

En lugar de simular 1 partícula, se simulan 10,000 partículas diferentes



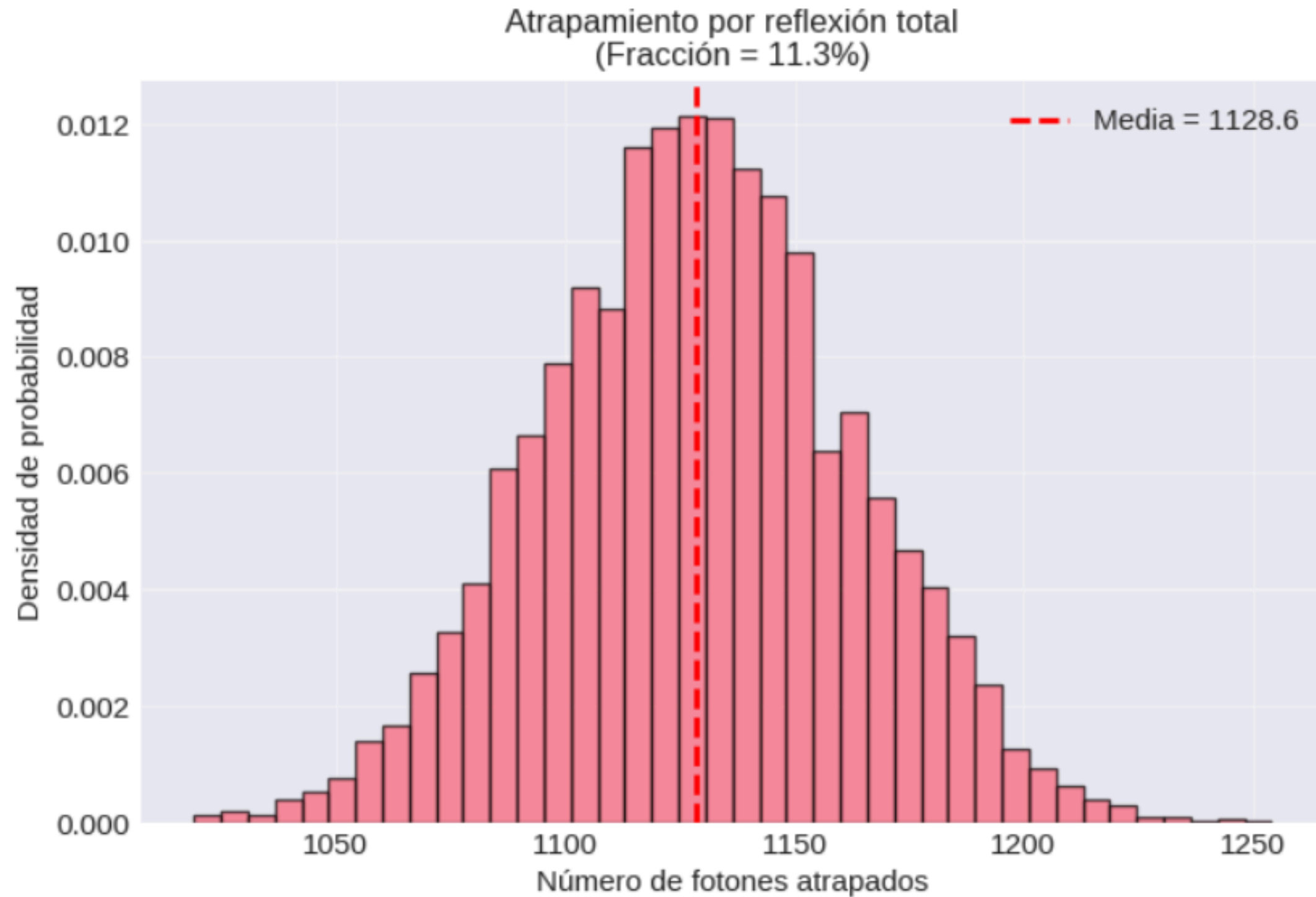
Obtengo una distribución y puedo hallar: Media, Desviación y resolución energética

Resultados de la Simulación Monte Carlo (10.000 eventos)



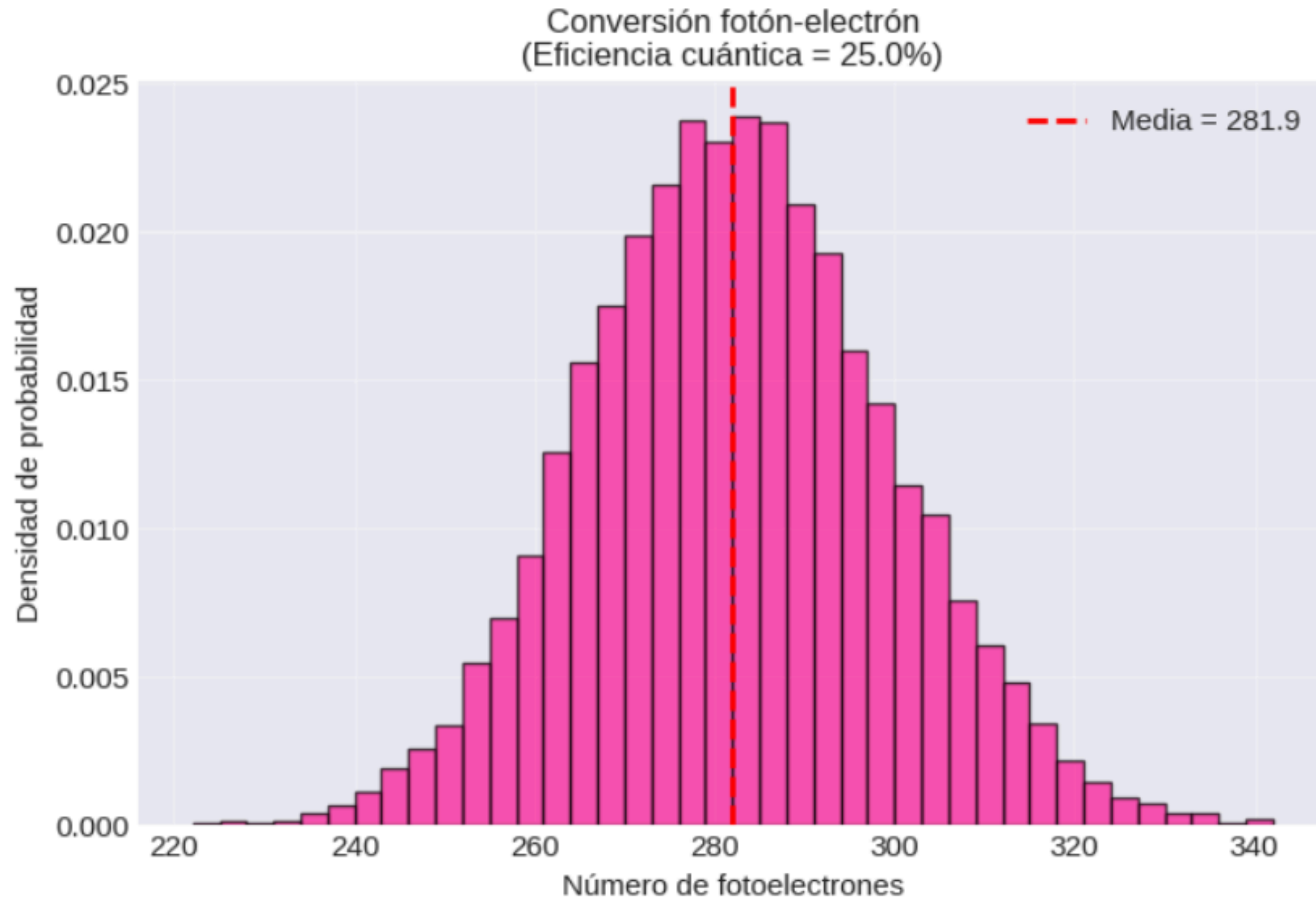
```
media_fotones = np.mean(fotones_generados)  
std_fotones = np.std(fotones_generados)
```

Resultados de la Simulación Monte Carlo (10.000 eventos)



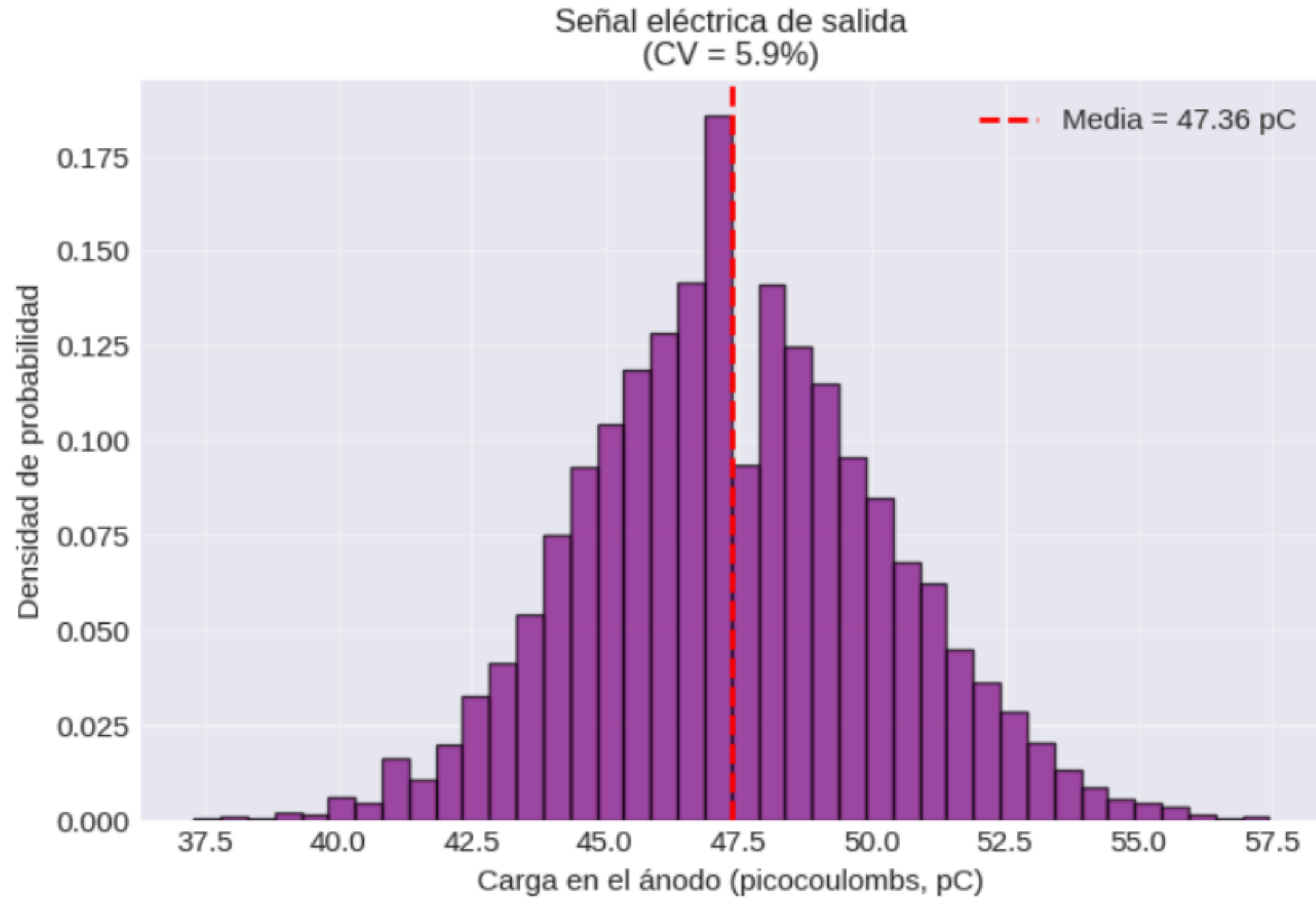
```
media_atrapados = np.mean(fotones_atrapados)
std_atrapados = np.std(fotones_atrapados)
fraccion_atrapados = 100 * media_atrapados / media_fotones
```

Resultados de la Simulación Monte Carlo (10.000 eventos)



```
media_pe = np.mean(fotoelectrones_obtenidos)  
std_pe = np.std(fotoelectrones_obtenidos)
```

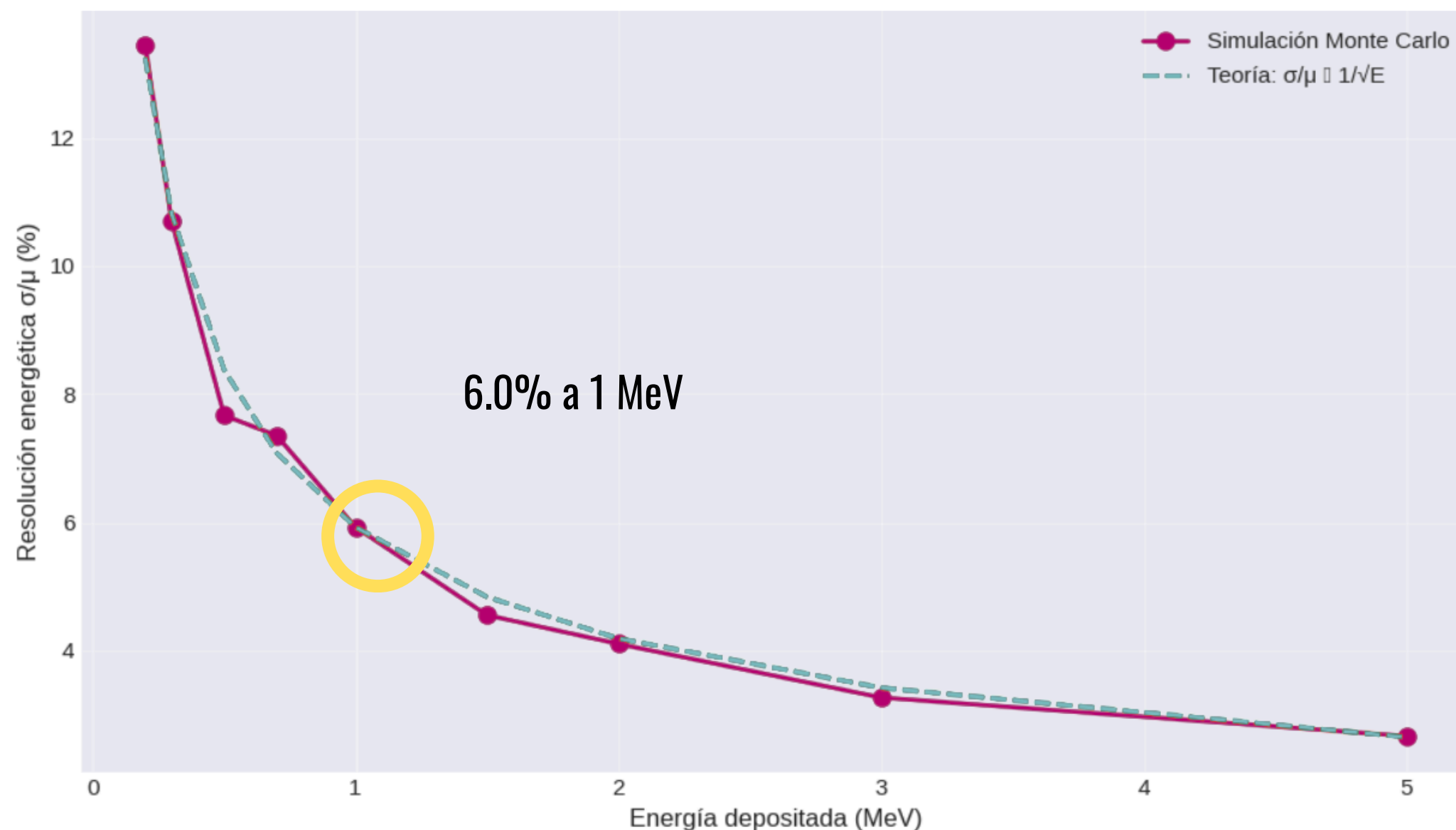
Resultados de la Simulación Monte Carlo (10.000 eventos)



```
media_carga = np.mean(carga_anodo)  
std_carga = np.std(carga_anodo)
```

```
cv_fotones = 100 * std_fotones / media_fotones
```

Curva de Resolución Energética del Detector



```
N_prom = energia_mev * rendimiento
cargas = []
```

```
for i in range(num_eventos):
    # Generación de fotones
    N_f = np.random.poisson(N_prom)
```

```
    # Direcciones y atrapamiento
    theta = np.arccos(2 * np.random.random(N_f) - 1)
    dz = np.cos(theta)
    atrapados = np.sum((dz > 0) & (theta < angulo_critico))
```

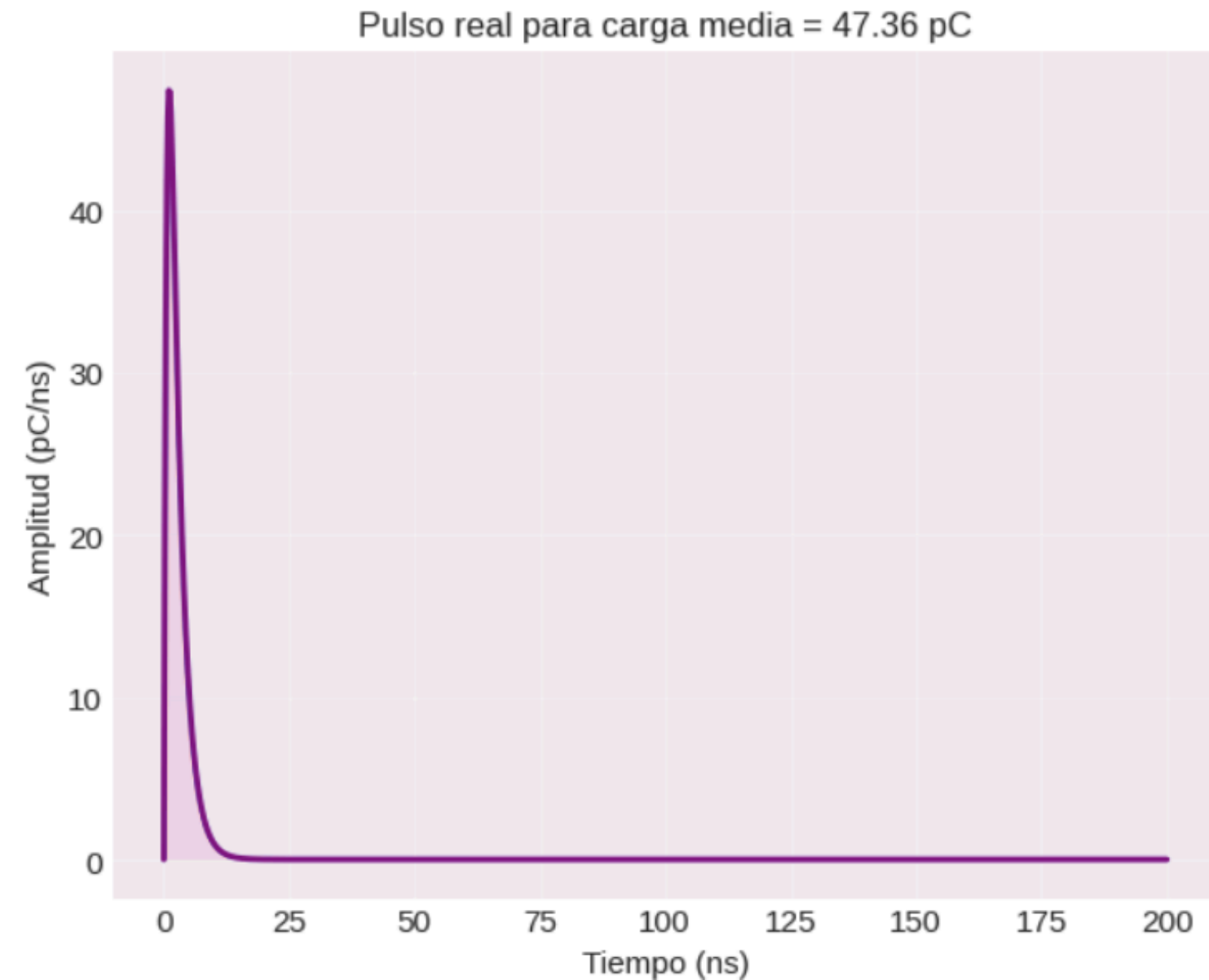
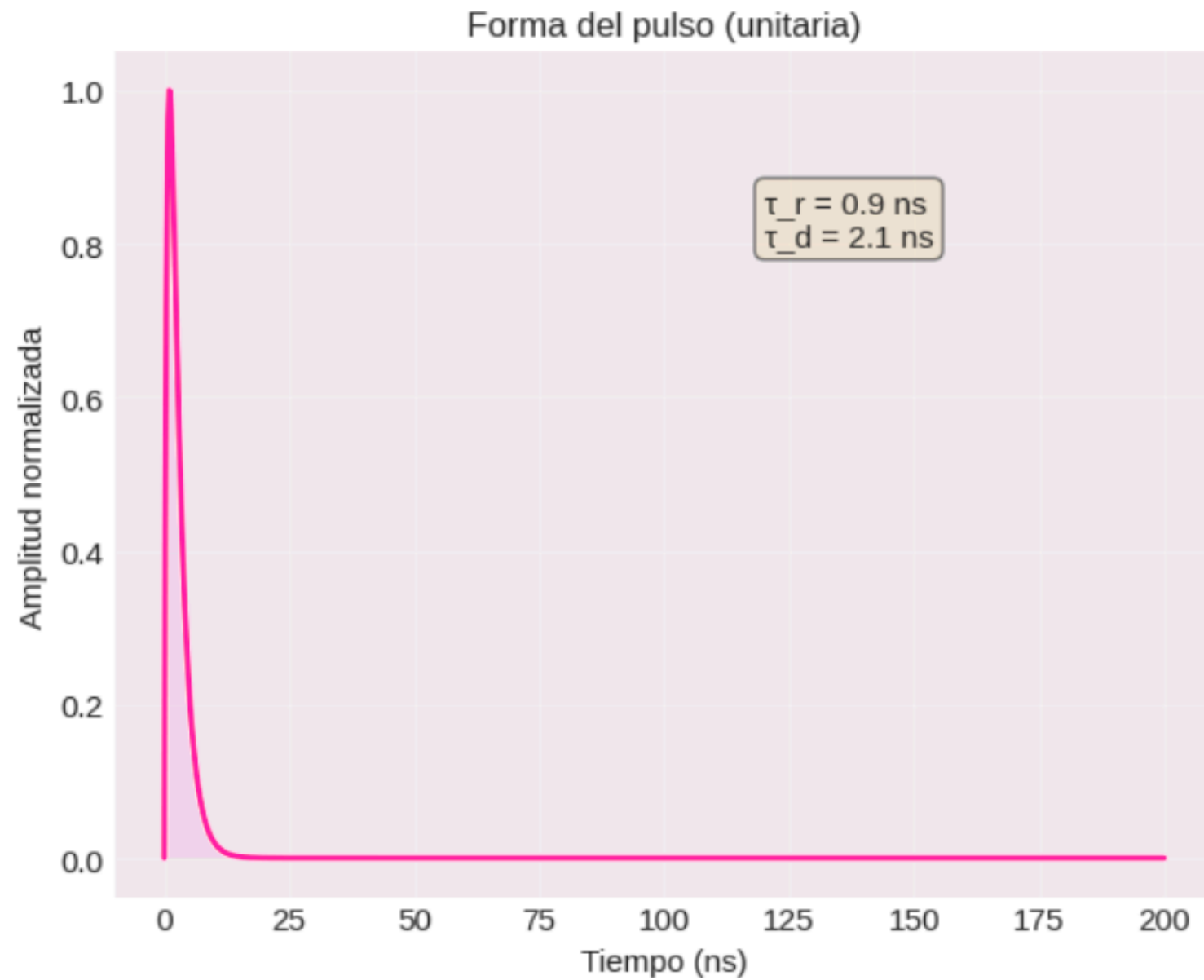
```
    # Conversión y multiplicación
    N_pe = np.random.binomial(atrapados, eficiencia_cuantica)
    carga = N_pe * ganancia_total * carga_electron
    cargas.append(carga)
```

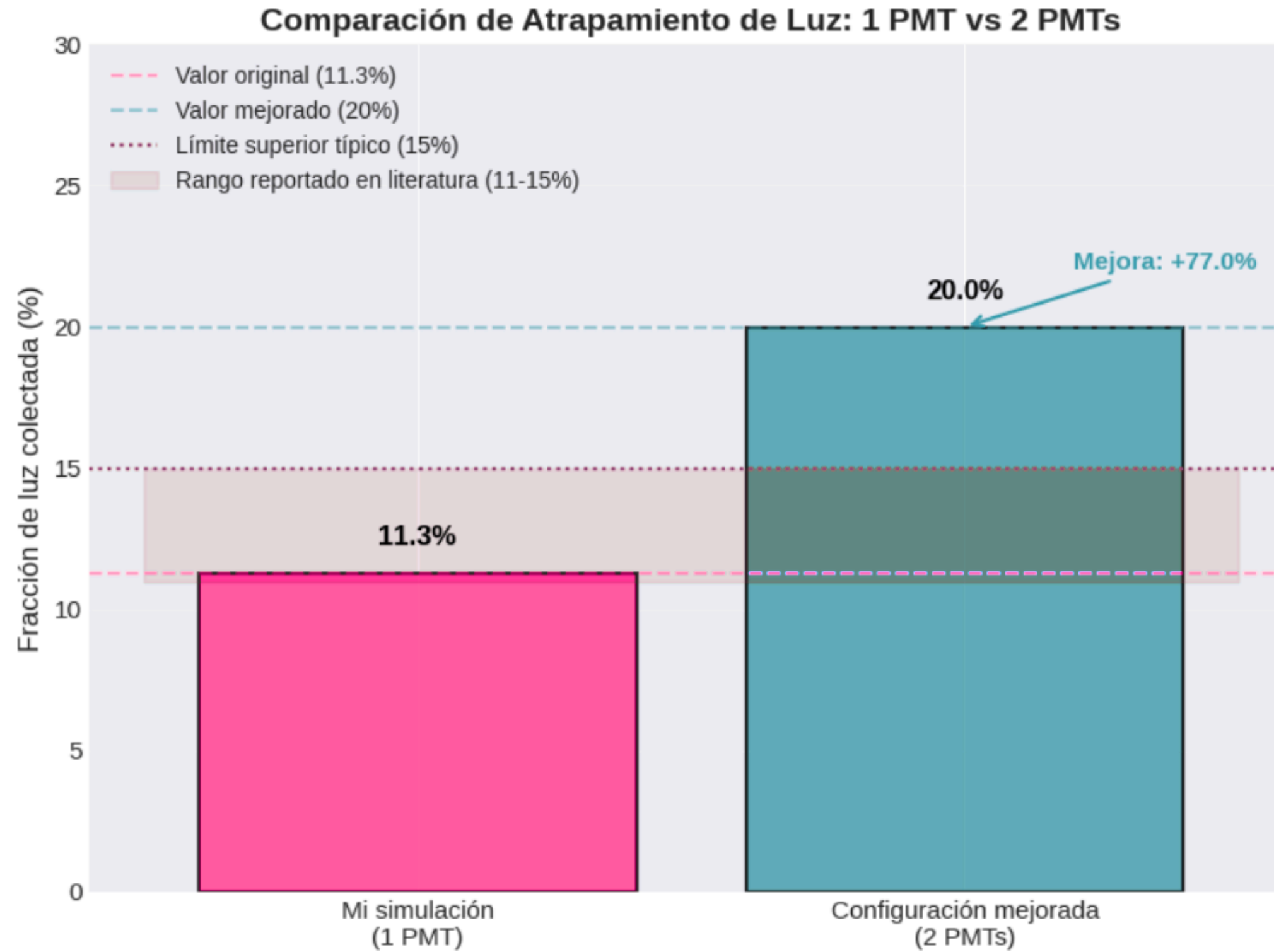
```
media = np.mean(cargas)
std = np.std(cargas)
resolucion = (std / media) * 100 #  $\sigma/\mu$  en porcentaje
```

```
return media, std, resolucion
```

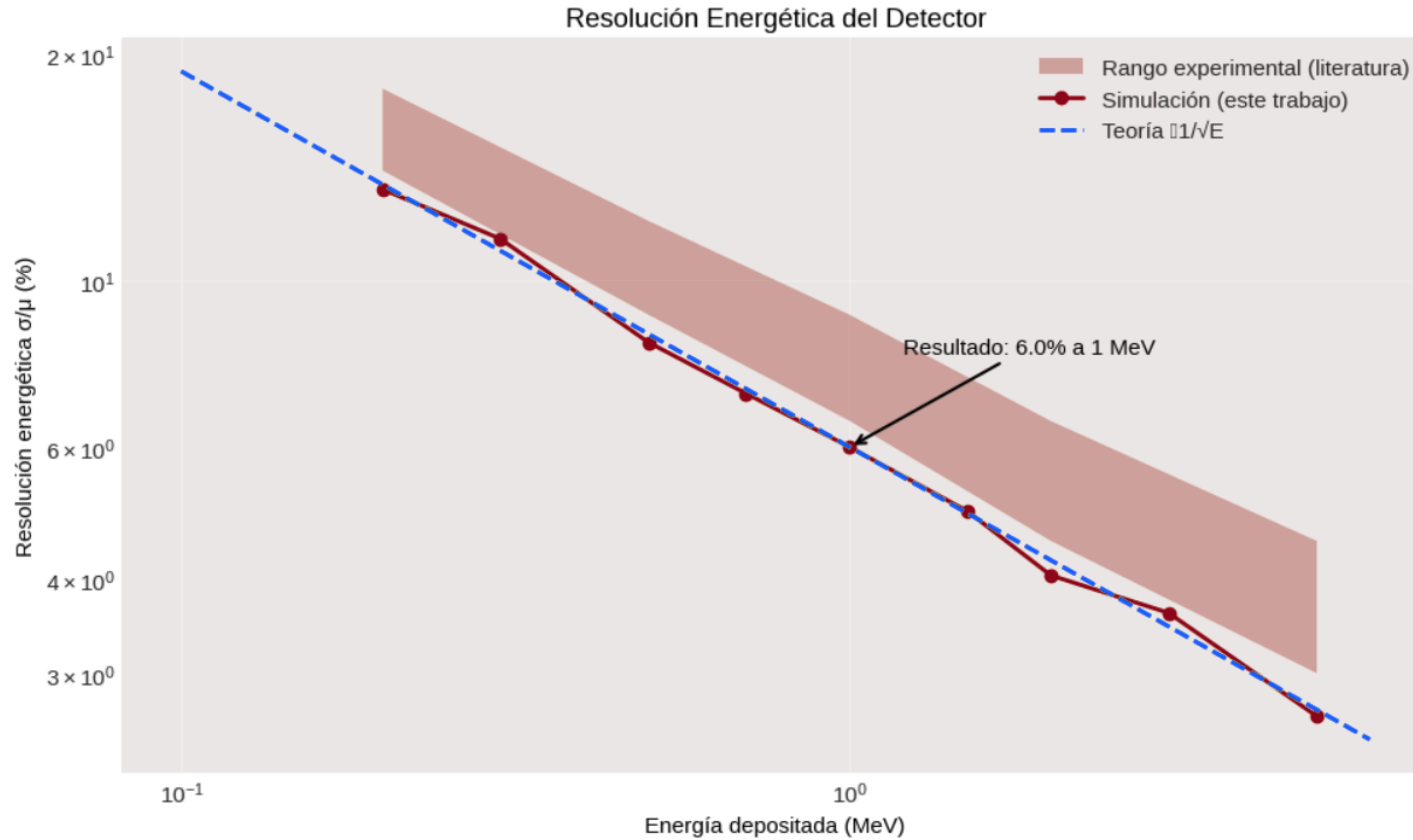
```
# Ajuste teórico: la resolución mejora con  $1/\sqrt{E}$ 
resolucion_referencia = resoluciones[energias_prueba.index(1.0)] # valor a 1 MeV
curva_teorica = [resolucion_referencia * np.sqrt(1.0/E) for E in energias_prueba]
plt.plot(energias_prueba, curva_teorica, 'b--', linewidth=2,
        label='Teoría:  $\sigma/\mu \propto 1/\sqrt{E}$ ', color="yellow")
```

La señal final: Pulso del PMT

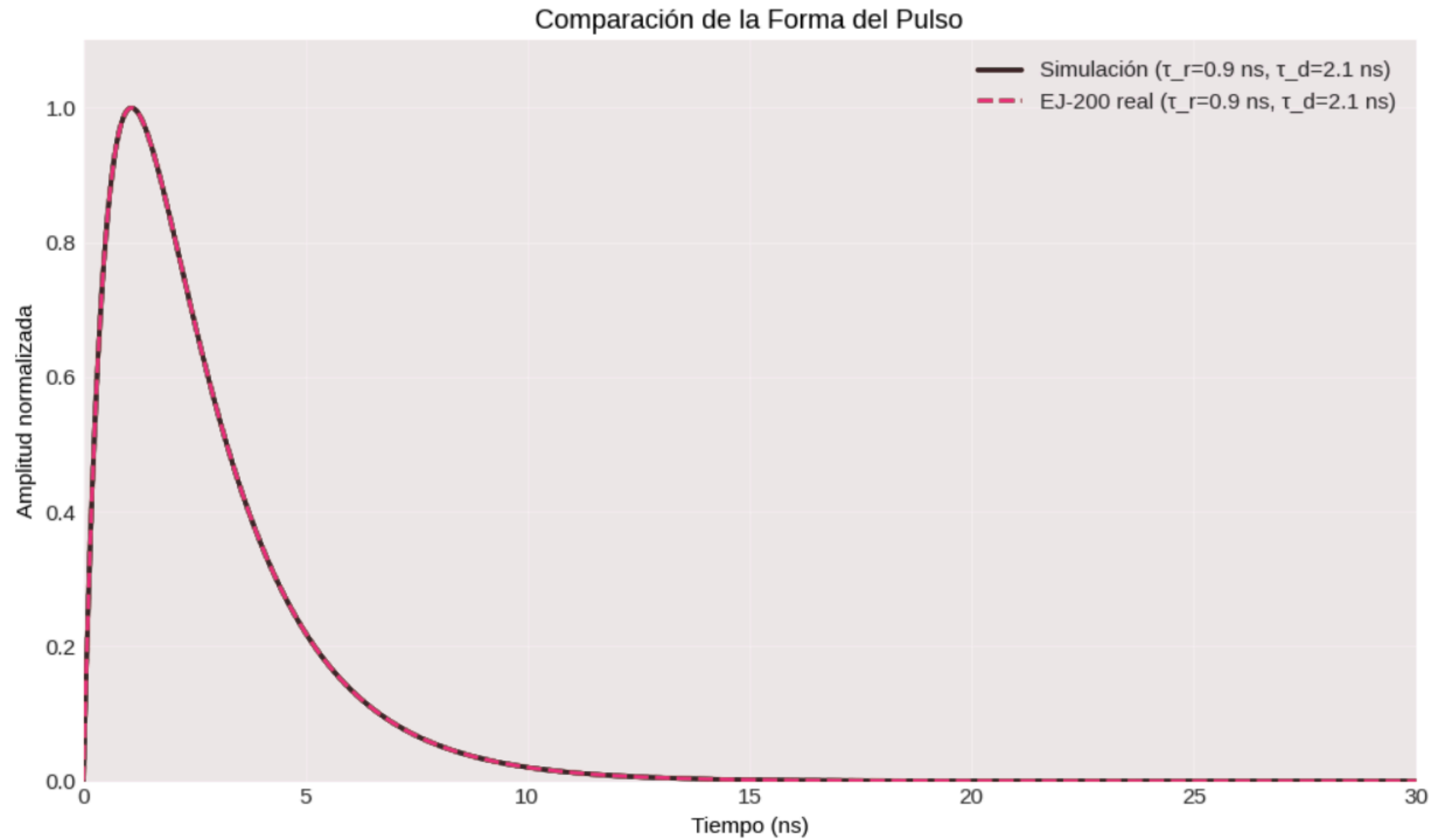




Comparamos con Zhu et al. (IBIC2016) - material EJ-200



Comparamos con Zhu et al. (IBIC2016) - material EJ-200



Comparamos con Zhu et al. (IBIC2016) - material EJ-200

Conclusiones

- La simulación reproduce correctamente las fluctuaciones estadísticas fundamentales del detector.
- El atrapamiento por reflexión total interna es el factor más crítico en la eficiencia (11.3% de fotones útiles).
- El pulso eléctrico obtenido ($\tau_r=0.9\text{ns}$, $\tau_d=2.1\text{ns}$) es consistente con PMT comerciales.

Perspectivas:

- Añadir reflectividad en borde.
- Incluir dependencia espectral de la QE.
- Implementar en Geant4 (estándar industrial).
- Validación experimental en laboratorio.

Referencias

- Ceravolo, S., Gargiulo, R., Paesani, D., Russo, A., & Sarra, I. (2023). X-ray spectroscopy with a photon-counting SiPM-based scintillation detector. *Journal of Instrumentation*, 18, T09007.
- Knoll, G. F. (2010). *Radiation Detection and Measurement* (4th ed.). Wiley.
- Mitev, K. K. (2016). Measurement of ^{222}Rn by absorption in plastic scintillators and alpha/beta pulse shape discrimination. *Radiation Physics and Chemistry*, 110, 236-243.
- Cern.ch. <https://indico.cern.ch/event/975141/contributions/4186059/attachments/2171928/4215106/2020-Lecture-4-3-Scintillators.pdf>
- Matsusada.com. https://www.matsusada.com/application/ps/photomultiplier_tubes
- Zhu, J., Fang, M. H., Wang, J., & Wei, Z. Y. (2016). Timing Window and Optimization for Position Resolution and Energy Calibration of Scintillation Detector. In *Proceedings of the 5th International Beam Instrumentation Conference (IBIC'16)* (pp. 123-126). Barcelona, Spain.

Repositorio

<https://github.com/annjgs/Monte-Carlo-Simulation-of-a-Plastic-Scintillator-Detector-with-Photomultiplier-Tube/tree/code>

¡Gracias por su atención!

ana.jativa@correounivalle.edu.co

gabriel.tarazona@correounivalle.edu.co

mariana.roa@correounivalle.edu.co

