SPM

May 27, 2022

0.0.1 Aspectos Generales de Simulación Numérica

Al igual que el problema de dispersión, definimos la función `fwhm` que calcula el ancho de un pulso con definición full width at half maximum. La función `power` que calcula la potencia de un pulso, y `normalize`, que la usa para forzar el teorema de Parseval sobre un par transformado por FFT.

```
fwhm (generic function with 1 method)
```

```
normalize (generic function with 1 method)
```

Ya que vamos a trabajar con más de un tipo de fibra, definimos un tipo de dato que represente los specs relevantes de una fibra óptica.

```
SMF: Fibra(1.2 km^-1 W^-1, 1.0046157902783952 km^-1)
NZ_DSF = Fibra(1.8 km^-1 W^-1, 1.0519618738232228 km^-1)
```

Elegimos el número de puntos en el que vamos a discretizar el dominio, y elegimos propagar un pulso de ancho de 10 ps. Definimos el vector z de 0 a 80 km

```
(0.0:0.15655577299412915:80.0) km
```

Elegimos la fibra SMF y un pulso con potencia de 1 mW. En función de esto calculamos la longitud característica de no-linealidad L_{NL} y la de atenuación L_{eff}

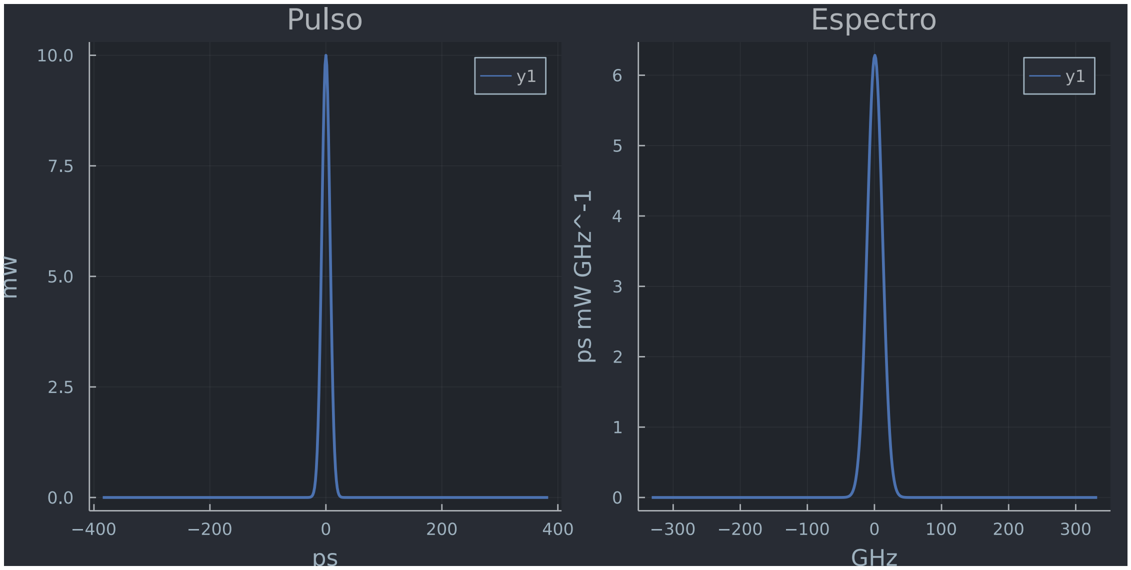
$$L_{NL} = \frac{1}{\gamma P_0} \qquad L_{eff} = \frac{1}{\alpha}$$

```
Longitud característica de atenuación = 0.9954054173515269 km
Longitud característica de no-lineal. = 83.33333333333333 km
```

Definimos un vector de frecuencias en función del ancho final del espectro en el caso con más SPM (máximo γ, máximo P_0)

$$B_f = \sqrt{1 + \frac{4}{3\sqrt{3}}\phi_{max}}\, B_0 \qquad \phi_{max} = \gamma P_0 L$$

Definimos el vector de tiempo en función de la resolución del vector de frecuencias. Sobre ese vector de tiempo definimos el pulso gaussiano de potencia P_0



Nuevamente definimos una grilla de coordenadas discretas $(i,\, j)$ en donde vamos a evaluar el problema. Esta grilla se va a usar tanto para indexar coordenadas $(z,\, t)$ como $(z,\, f)$.

```
512×512 Matrix{Tuple{Int64, Int64}}:
 (1, 1)    (1, 2)    (1, 3)    …  (1, 510)    (1, 511)    (1, 512)
 (2, 1)    (2, 2)    (2, 3)       (2, 510)    (2, 511)    (2, 512)
 (3, 1)    (3, 2)    (3, 3)       (3, 510)    (3, 511)    (3, 512)
 (4, 1)    (4, 2)    (4, 3)       (4, 510)    (4, 511)    (4, 512)

 (510, 1)  (510, 2)  (510, 3)     (510, 510)  (510, 511)  (510, 512)
 (511, 1)  (511, 2)  (511, 3)  …  (511, 510)  (511, 511)  (511, 512)
 (512, 1)  (512, 2)  (512, 3)     (512, 510)  (512, 511)  (512, 512)
```

0.0.2 Automodulación de Fase sin Atenuación

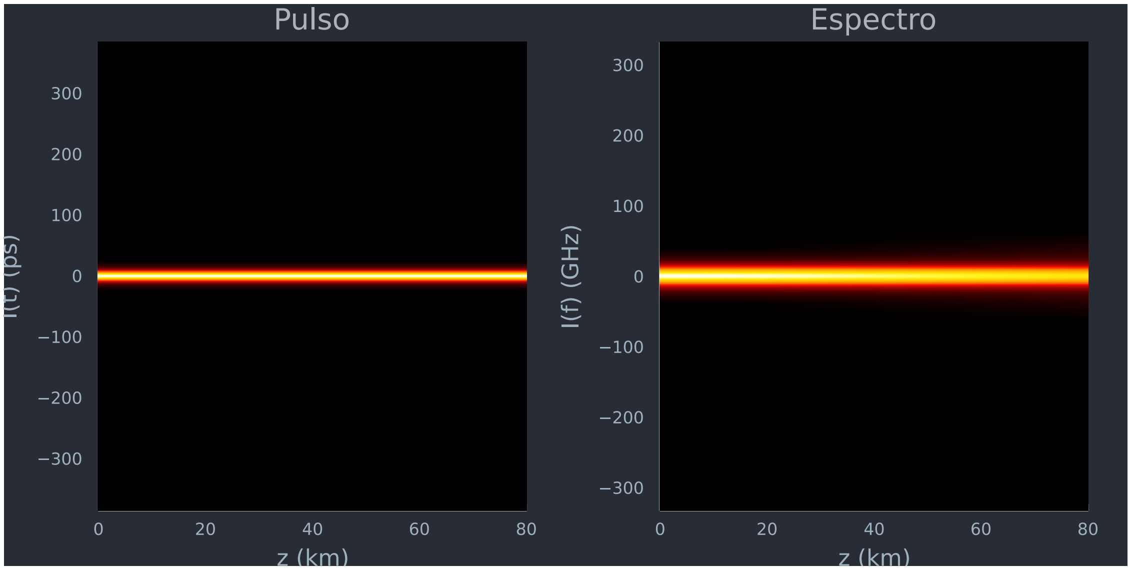
Para para calcular la evolución del espectro a lo largo de la fibra, usamos el resultado

$$A(z,t) = A(0,t)\exp\left[i\gamma|A(z,0)|^2 z\right] \longrightarrow A_t[i,j] = A_{t0}[j]\exp\left[i\gamma|A_{t0}[j]|^2 z_i\right]$$

```
A (generic function with 1 method)
```

Evaluamos $A_t[i,j]$ en los puntos $(i,\, j) \leftrightarrow (z,\, t)$. Transformamos Fourier a lo largo de la dimensión j para obtener el resultado $A_f[i,j]$ en el dominio $(i,\, j) \leftrightarrow (z,\, f)$

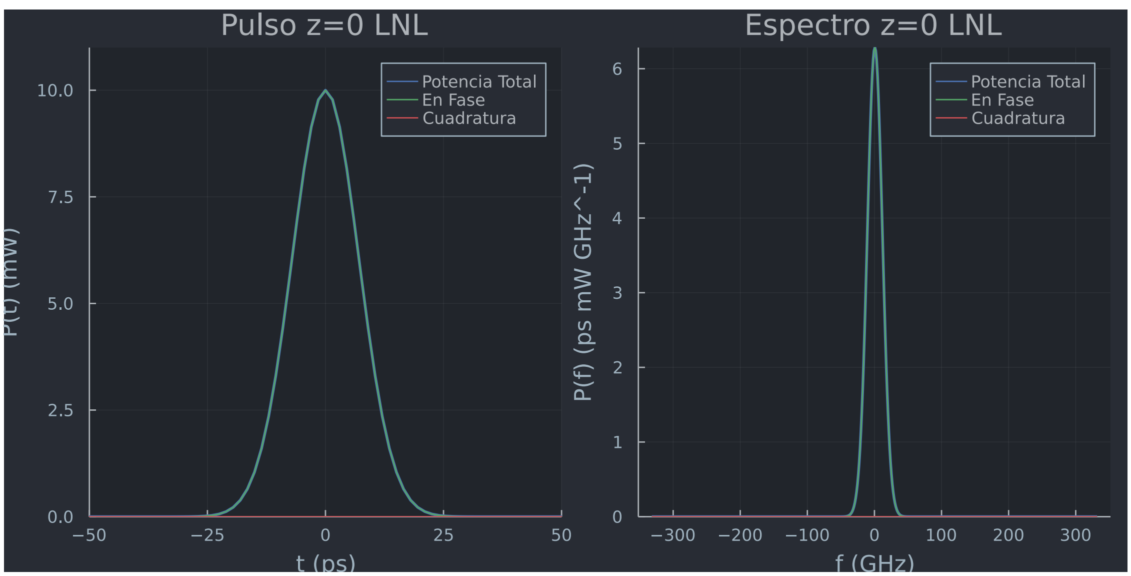
Visualizamos la evolución de la forma del pulso y del espectro en forma *heatmap*

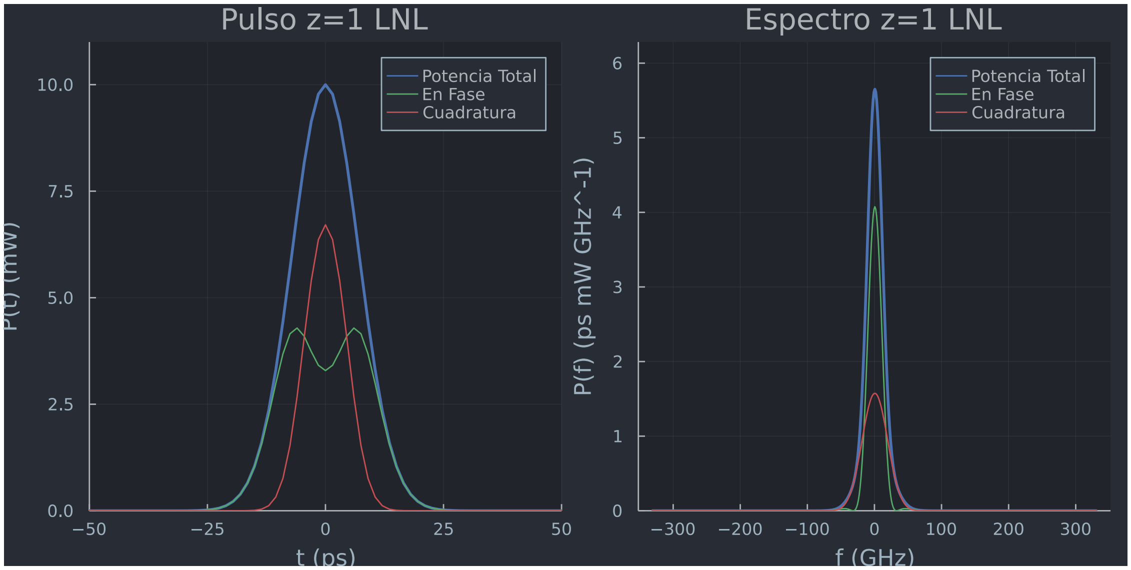


Y también con una animación que muestra la forma del pulso mientras este se propaga a través de la fibra, y tomas instantáneas de esa animación

```
Info: Saved animation to
  fn = /tmp/jl_gkvx6Y.gif
@ Plots /home/inox/.julia/packages/Plots/1KWPG/src/animation.jl:114
```

```
Plots.AnimatedGif("/tmp/jl_gkvx6Y.gif")
```





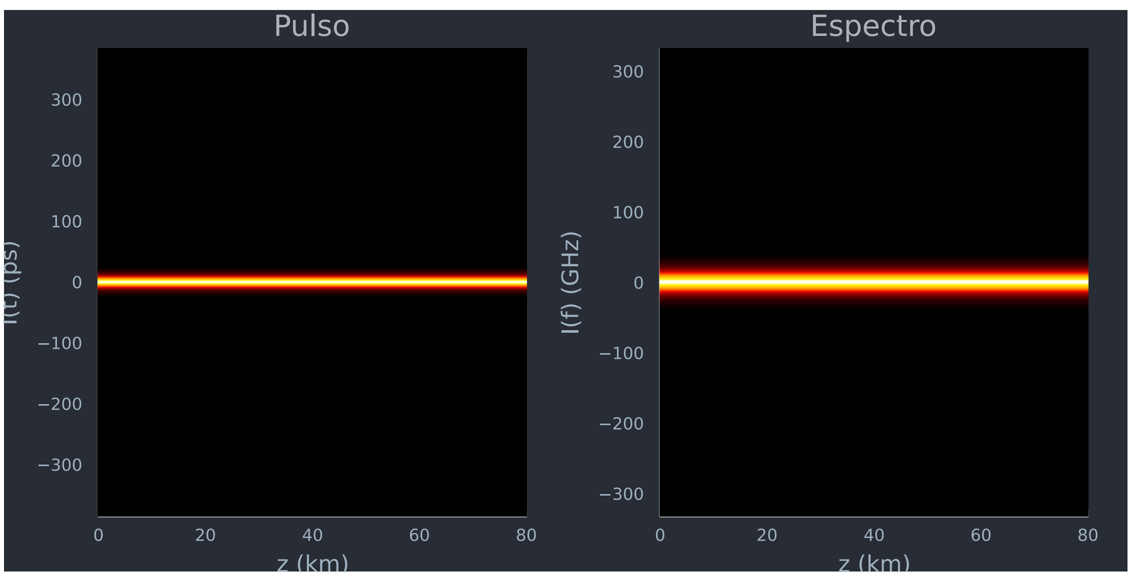
0.0.3 Automodulación de Fase con Atenuación

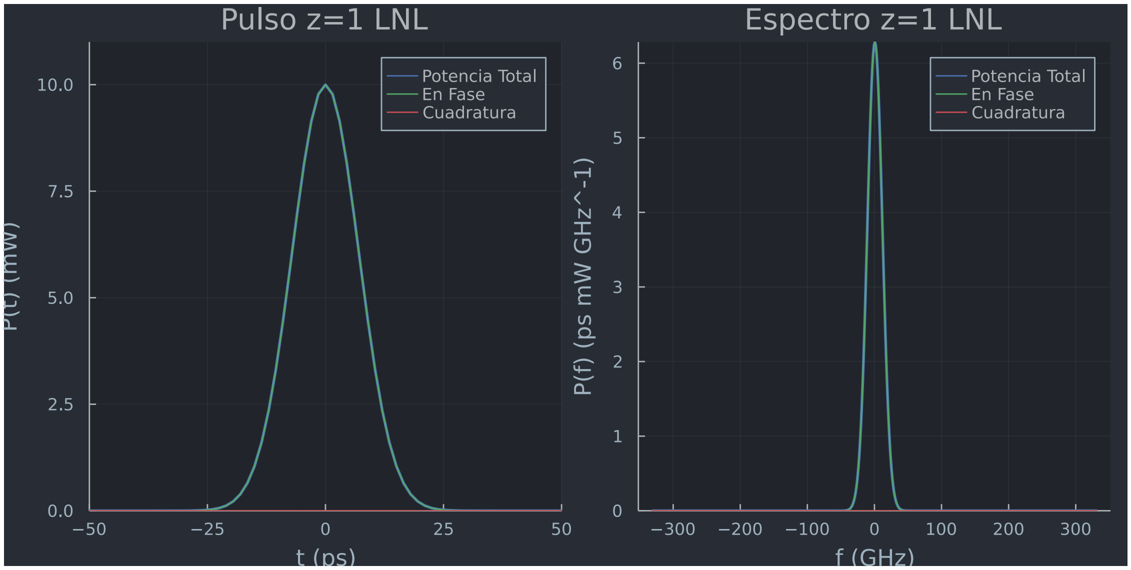
Ahora consideramos el efecto de la atenuación, usamos el resultado

$$A(z,t) = A(0,t)\exp\left[i\gamma|A(z,0)|^2 z_{eff}\right] \longrightarrow A_t[i,j] = A_{t0}[j]\exp\left[i\gamma|A_{t0}[j]|^2 z_{eff}[i]\right]$$

Donde la longitud efectiva es: $z_{eff} = \dfrac{1-e^{-\alpha z}}{\alpha}$

```
A_att (generic function with 1 method)
```



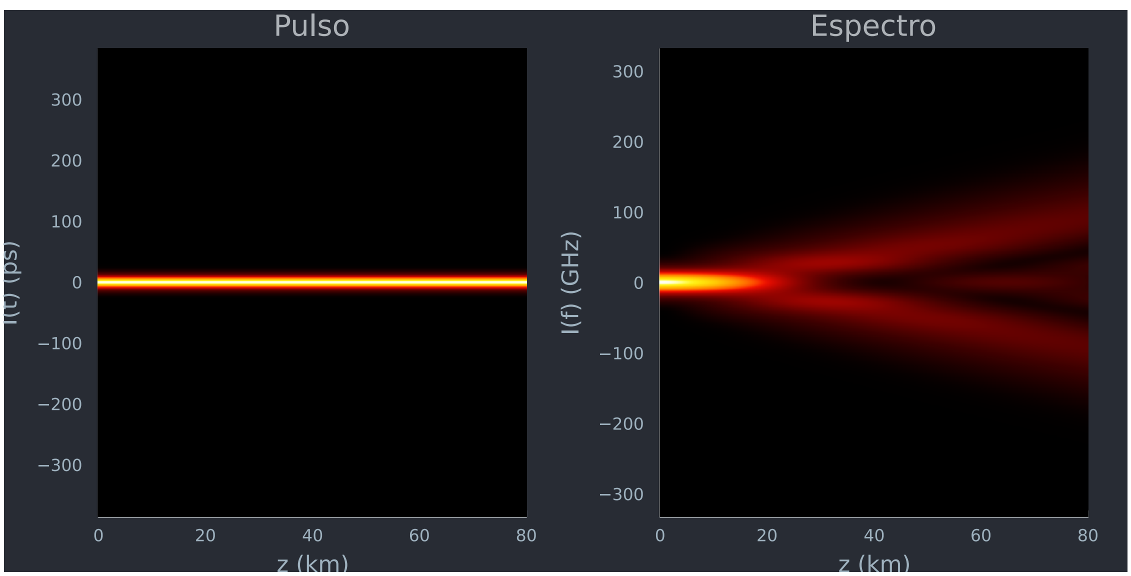


0.0.4 Efecto de incrementar potencia del pulso

Repetimos el análisis incrementando la potencia del pulso a 100 mW

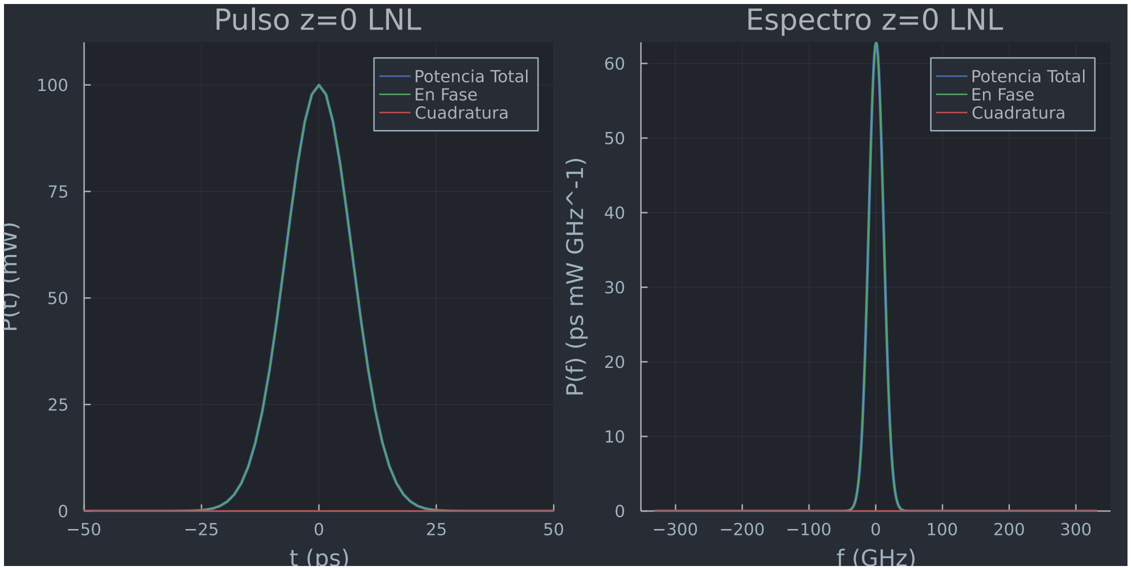
```
Longitud característica de atenuación = 0.9954054173515269 km
Longitud característica de no-lineal. = 8.333333333333334 km
```

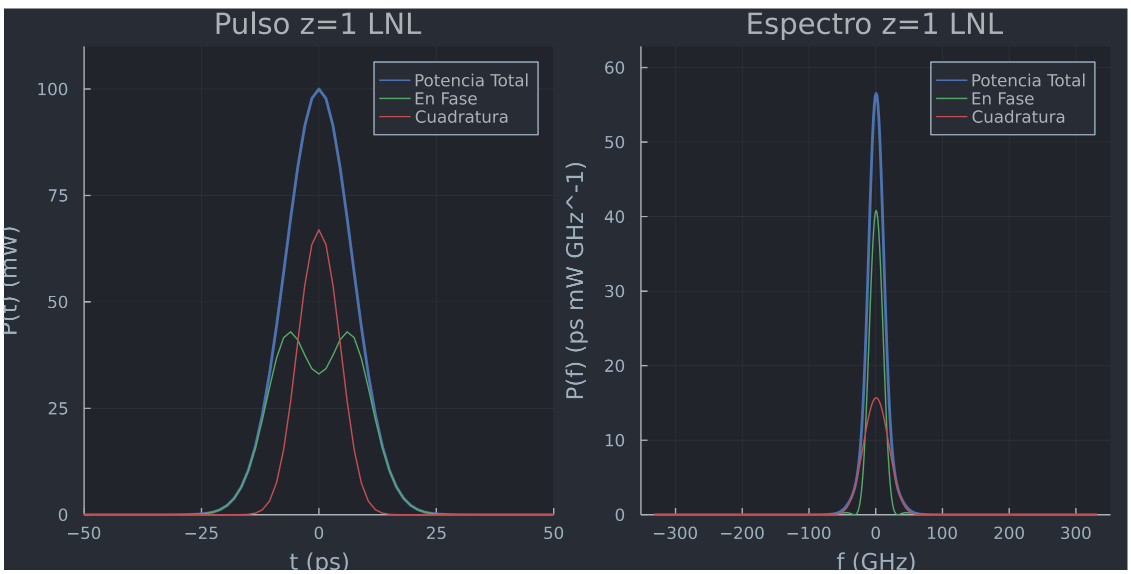
Primero sin atenuación:

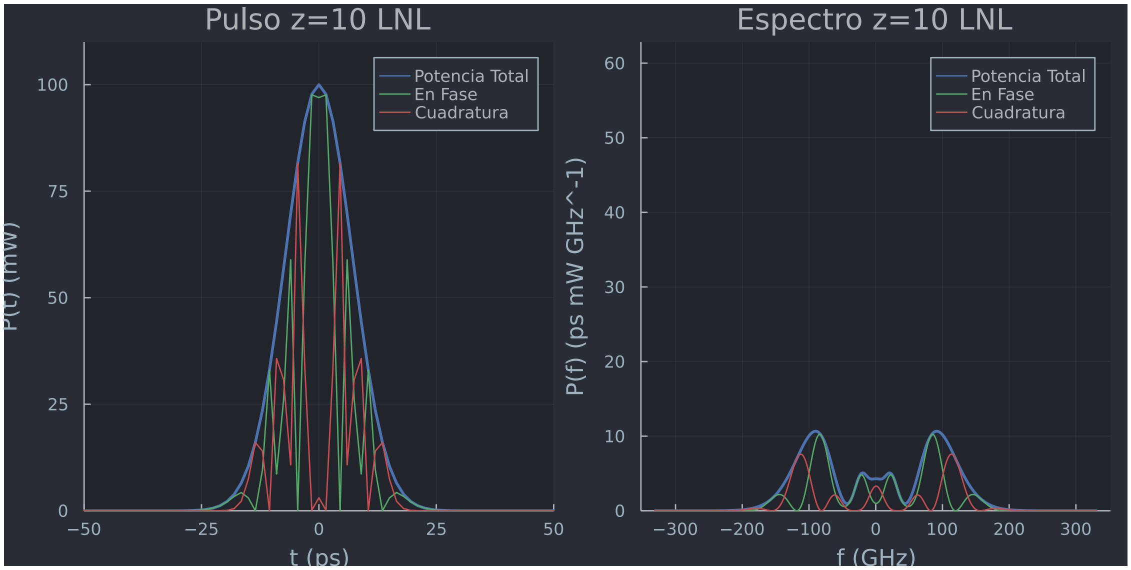



```
Info: Saved animation to
  fn = /tmp/jl_VDN1dd.gif
@ Plots /home/inox/.julia/packages/Plots/1KWPG/src/animation.jl:114
```

```
Plots.AnimatedGif("/tmp/jl_VDN1dd.gif")
```



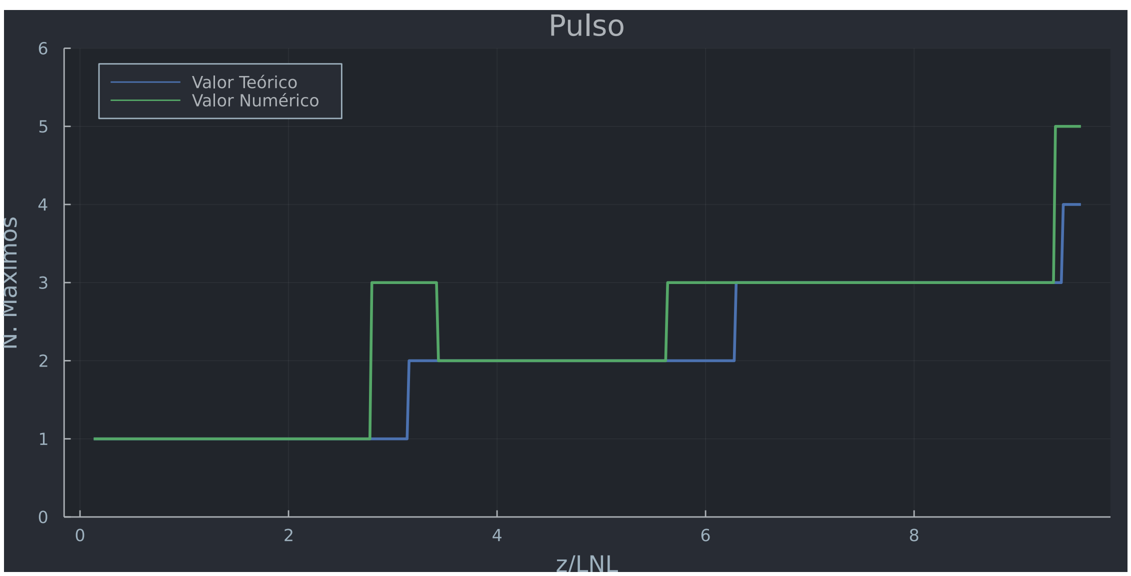




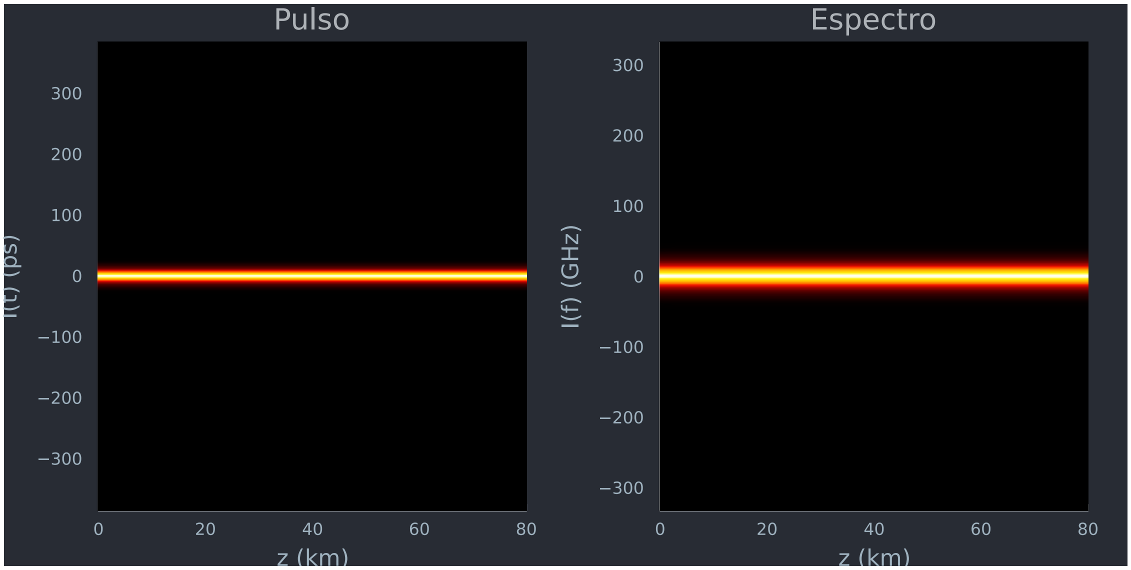
Ya que en este ejemplo se puede apreciar visualmente el ensanchamiento del espectro, podemos medir el número de máximos y comparar el valor medido con la aproximación teórica

$$M \simeq \frac{\phi_{max}}{\pi} + \frac{1}{2} \qquad\qquad \phi_{max} = \gamma P_0 z_{eff}$$

El número de máximos de la simulación numérica se calcula importando la librería `Peaks` provista por el lenguaje, notar que el resultado analítico es solo una aproximación, no se espera una correspondencia exacta con la simulación numérica



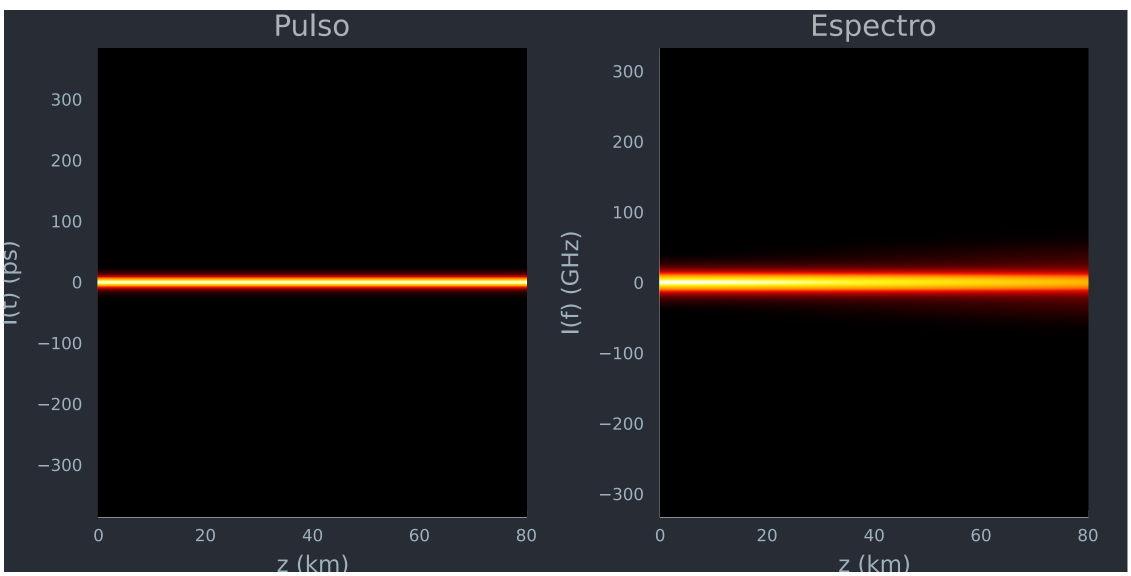
Luego con atenuación:



0.0.5 Fibra TW-RS

Ahora simulamos la propagación del pulso con la fibra TW-RS, primero con un pulso de 10 mW, sin atenuación

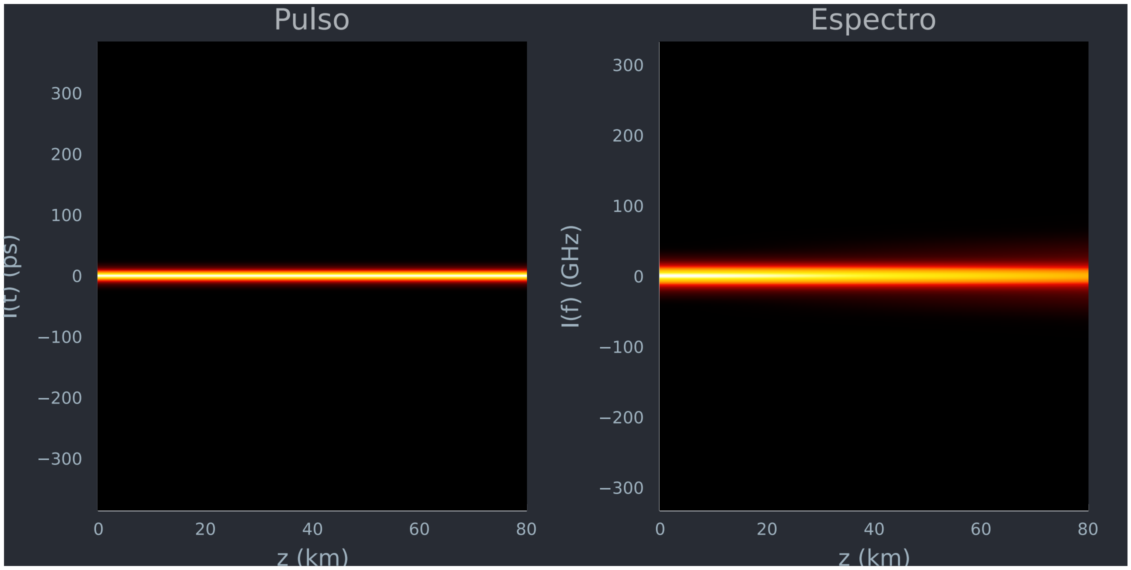
```
Longitud característica de atenuación = 0.9506047936562816 km
Longitud característica de no-lineal. = 55.55555555555555 km
```




```
Info: Saved animation to
  fn = /tmp/jl_VvoeRm.gif
@ Plots /home/inox/.julia/packages/Plots/1KWPG/src/animation.jl:114
```

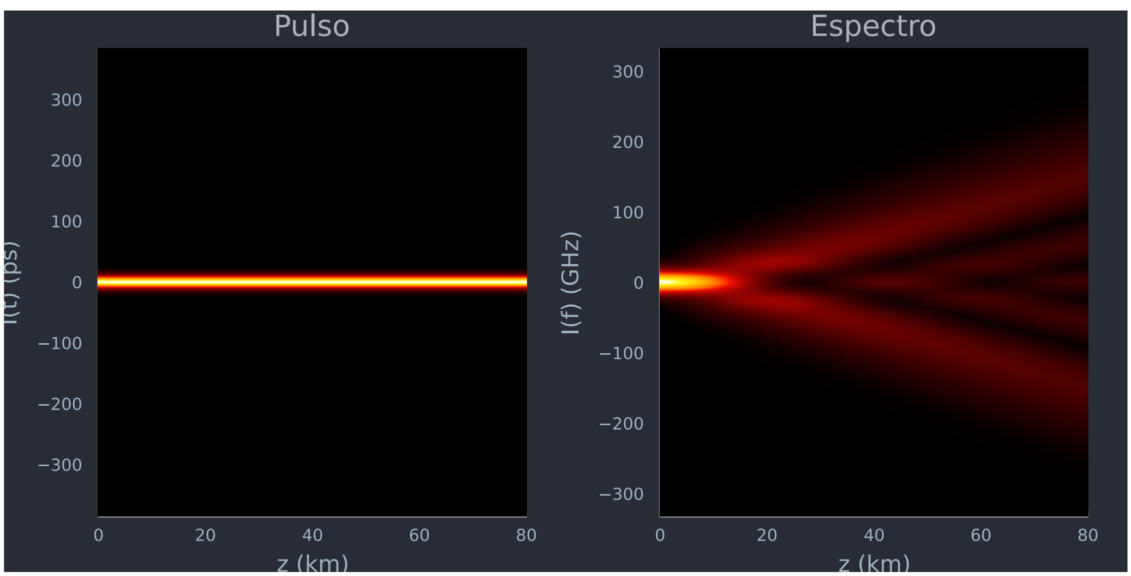
```
Plots.AnimatedGif("/tmp/jl_VvoeRm.gif")
```

Y con atenuación



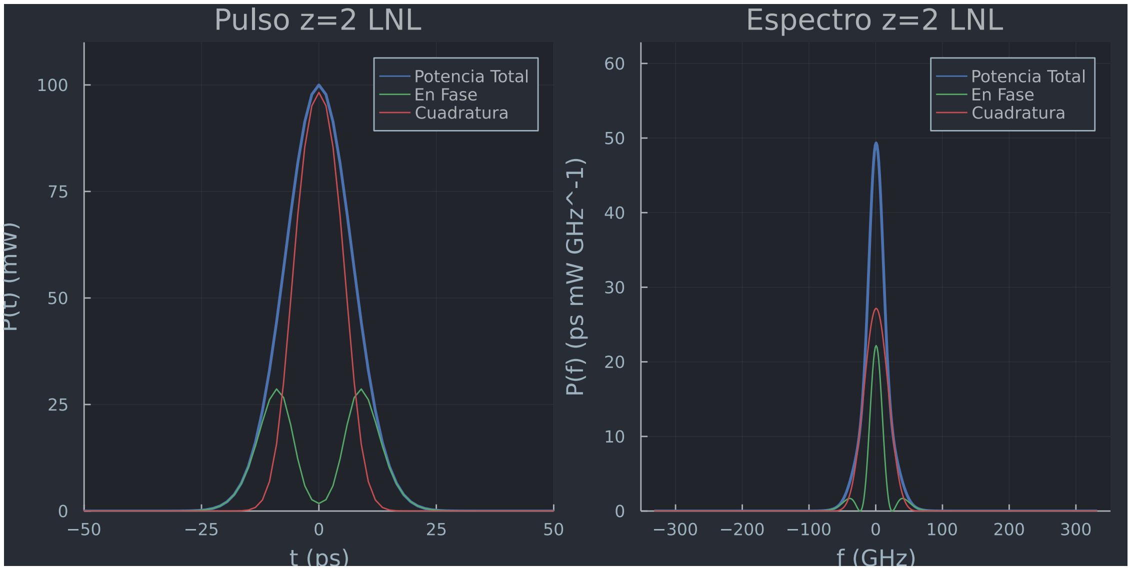
Finalmente la fibra TW-RS con un pulso de 100 mW, sin atenuación

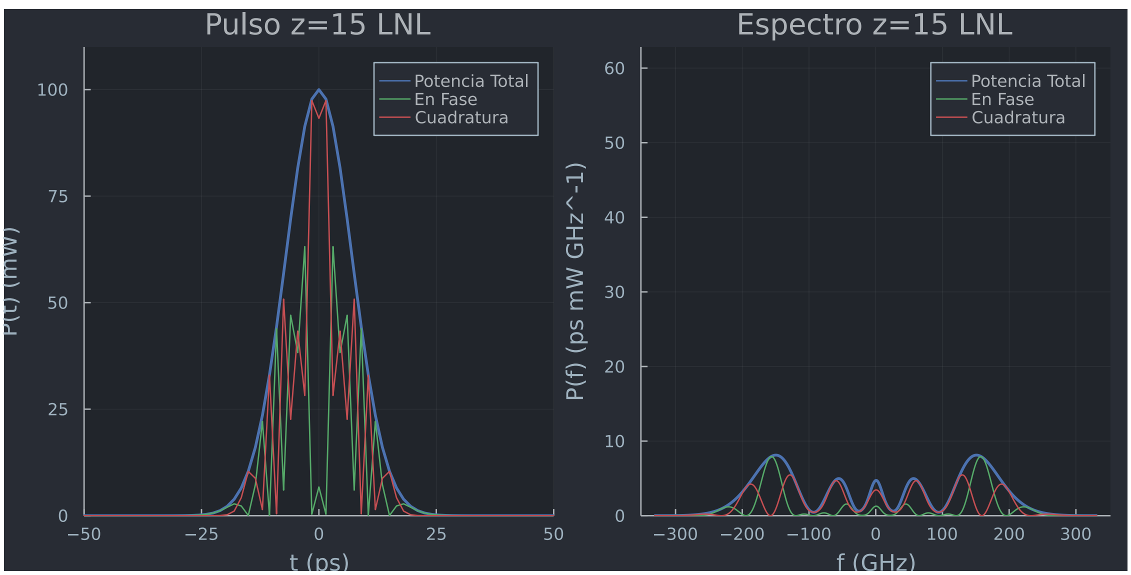
```
Longitud característica de atenuación = 0.9506047936562816 km
Longitud característica de no-lineal. = 5.555555555555555 km
```



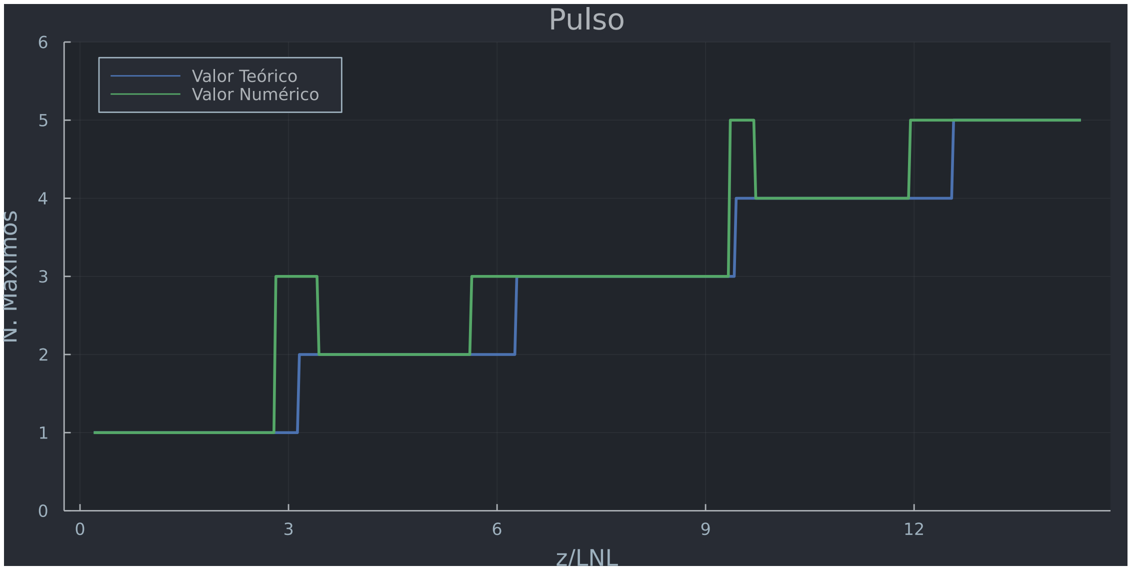

```
Info: Saved animation to
  fn = /tmp/jl_Oy3evu.gif
@ Plots /home/inox/.julia/packages/Plots/1KWPG/src/animation.jl:114
```

```
Plots.AnimatedGif("/tmp/jl_Oy3evu.gif")
```

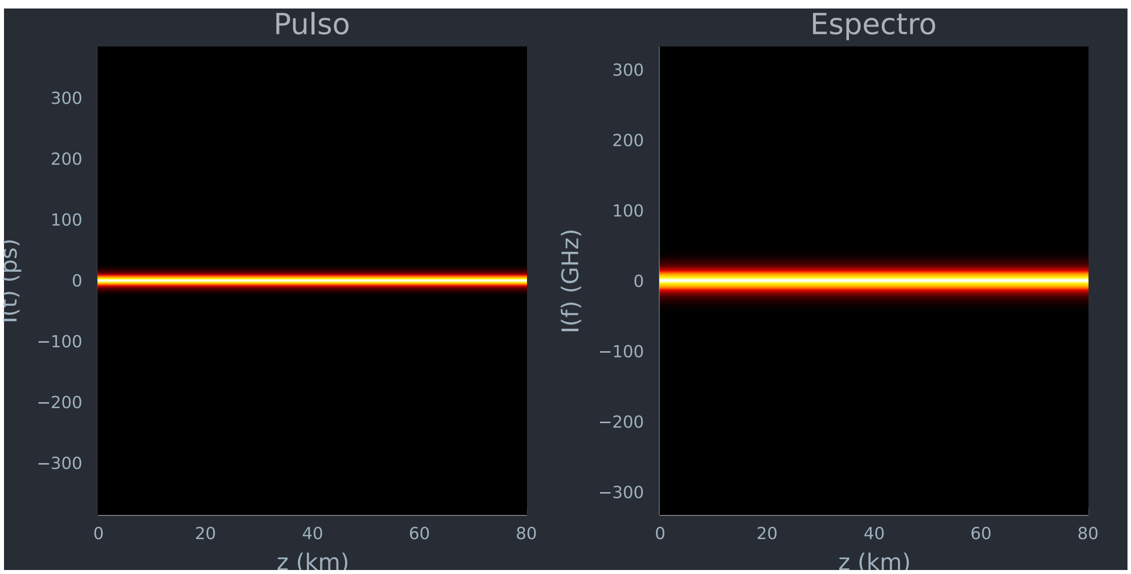




Vemos nuevamente el incremento de máximos en el espectro



Y con atenuación



En todos los casos se ve que la atenuación efectivamente mata a la SPM muy rápidamente, para estos valores de α y γ