

Calculates SKR and QBER a for 'plug and play' phase coding BB84-QKD system, given bandwidth and shape parameters of channel bands for Liquid Crystal on Silicon (LCoS)-based WSS nodes (C. Pulikkaseril, et al., "Spectral modeling of channel band shapes in wavelength selective switches", *Optics Express*, Vol.19, No.9 (2011).), for short fiber segment in order to mitigate the effect of Spontaneous Raman Scattering (SpRS).

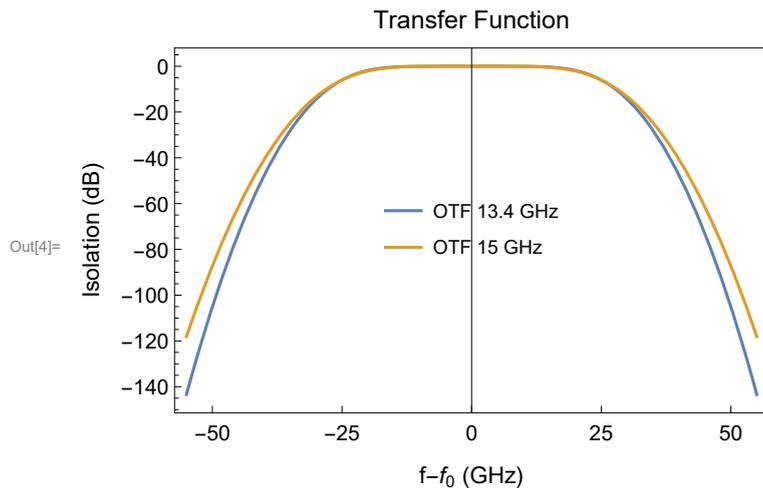
In[1]:= **B = 50; (*put Bandwidth of optical filter*)**

In[2]:=
$$\sigma[\text{BWotf}_-] = \frac{\text{BWotf}}{2 \sqrt{2} \text{Log}[2]};$$

$$S[f_-, \text{BWotf}_-] = \left(\frac{1}{2} \left(\text{Erf}\left[\frac{-B/2 - f}{\sqrt{2} \sigma[\text{BWotf}]} \right], \frac{B/2 - f}{\sqrt{2} \sigma[\text{BWotf}]} \right] \right)^2;$$

The bandpass filter response of 50 GHz Bandwidth for different OTF parameters (13.4 GHz, 15 GHz)

In[4]:= **Plot[{10 * Log10[S[f, 13.4]], 10 * Log10[S[f, 15]]}, {f, -55, 55},
PlotRange -> Automatic, PlotLegends -> Placed[{"OTF 13.4 GHz", "OTF 15 GHz"}, Center],
Frame -> True, FrameTicks -> {{All, None}, {{-50, -25, 0, 25, 50}, None}},
FrameLabel -> {Style["f-f₀ (GHz)", Black, FontSize -> 12],
Style["Isolation (dB)", Black, FontSize -> 12], Style["Transfer Function",
Black, FontSize -> 13]}, FrameTicksStyle -> Directive[Black, 11]**



Calculates filtering isolation for different values of OTF parameter for a 100GHz grid WDM topology

In[5]:= **{10 * Log10[NIntegrate[S[f, 13.4], {f, 50, 150}]],
10 * Log10[NIntegrate[S[f, 13.4], {f, 49.5, 149.5}]],
10 * Log10[NIntegrate[S[f, 13.4], {f, 49, 149}]],
10 * Log10[NIntegrate[S[f, 13.4], {f, 48.5, 148.5}]],
10 * Log10[NIntegrate[S[f, 13.4], {f, 48, 148}]],
10 * Log10[NIntegrate[S[f, 13.4], {f, 47.5, 147.5}]],
10 * Log10[NIntegrate[S[f, 13.4], {f, 47, 147}]]}
(* (relative to max. of transfer function) leakage
power into q. ch. passband in 100GHz Grid*)**

Out[5]= {-107.246, -103.689, -100.196, -96.7644, -93.3961, -90.0905, -86.8473}

```

In[6]:=  $\lambda = 1550 \times 10^{-9}$ ; (*quantum channel wavelength*)
 $\alpha = \frac{0.2}{4.343}$ ; (*fiber attenuation*)
 $\rho = 1.7 \times 10^{-9}$ ; (*put Raman cross-section value*)
 $\Delta\lambda = 0.8$ ; (*i.e. 100GHz q. passband (3dB bandwidth of SPAD's filter)*)
 $\text{Praman}[z_] = 10^{-3} * \rho \Delta\lambda z \text{Exp}[-\alpha z]$ ;
 $\text{Pcrst}[\text{dB}_, z_] = 10^{-3} * 10^{\frac{\text{dB}}{10}} \text{Exp}[-\alpha z]$ ; (*leaked power generated from the classical
PSD with average power of 0dBm. dB is the filtering isolation. An example
of finding filtering isolation for different OTF values is shown above*)

```

```

In[11]:=  $\text{Noisephotonscrst}[\text{dB}_, z_] = \frac{(\text{Pcrst}[\text{dB}, z]) * 10^{-9}}{6.62606896 \times 10^{-34} \frac{3 \times 10^8}{\lambda}}$ ;
 $\text{Noisephotonsraman}[z_] = \frac{(\text{Praman}[z]) * 10^{-9}}{6.62606896 \times 10^{-34} \frac{3 \times 10^8}{\lambda}}$ ;

```

```

In[13]:=  $T[z_] = \text{Exp}[-\alpha z]$ ; (*fiber loss*)
 $\mu[z_] = T[z]$ ; (*mean photon number for BB84*)
 $\eta_{\text{demux}}[\text{dB2}_] = 10^{-\frac{\text{dB2}}{10}}$ ; (*put dB loss of DEMUX*)
 $\eta_{\text{bob}} = 10^{-\frac{2.65}{10}}$ ; (*put dB loss of Bob's internal components*)
 $\eta_{\text{spad}} = 0.1$ ; (*q. efficiency*)
 $\text{cps} = 4 \times 10^3$ ; (*dark counts per second*)
 $\text{pap} = 0.008$ ; (*after-pulse probability to the total detection
probability fraction*)
 $\text{QBERdark} = 2 \text{cps} 10^{-9}$ ; (*dark count probability for 2 detectors and 1ns gate width*)
 $\text{Ntotal}[\text{dB}_, z_, \text{dB2}_] = \text{Noisephotonscrst}[\text{dB}, z] + \eta_{\text{demux}}[\text{dB2}] \text{Noisephotonsraman}[z]$ ;
 $\text{QBERwdm}[\text{dB}_, z_, \text{dB2}_] = \eta_{\text{spad}} \eta_{\text{bob}} \text{Ntotal}[\text{dB}, z, \text{dB2}]$ ;
 $b = 0.005$ ; (*baseline error rate  $b = \frac{1-V}{2}$ , V: Visibility*)
 $\text{frep} = 5 \times 10^6$ ; (*repetition rate*)
 $\text{tdead} = 0.1 \times 10^{-6}$ ; (*SPAD's dead time*)

```

The calculation of SKR was carried out assuming general incoherent attacks in the presence of multiphoton pulses, via the theoretical treatment in *PHYSICAL REVIEW A 71, 042316 (2005)*.

```

In[26]:= Ls = 24; (*delay line length*)
ηduty[z_] =  $\frac{Ls}{Ls + z}$ ;
QBERap[dB_, m_, z_, dB2_] =
  pap (m T[z] ηdemux[dB2] ηbob ηspad + QBERdark + QBERwdm[dB, z, dB2]);
ηdead[dB_, m_, z_, dB2_] = (1 + τdead frep (m T[z] ηdemux[dB2] ηbob ηspad +
  QBERdark + QBERap[dB, m, z, dB2] + QBERwdm[dB, z, dB2]))-1;
Rsec[dB_, m_, z_, dB2_] = Rsift[dB, m, z, dB2] (Iab[dB, m, z, dB2] - Iae[m, z]);
Rsift[dB_, m_, z_, dB2_] = 0.5
  (m T[z] ηdemux[dB2] ηbob ηspad + QBERdark + QBERap[dB, m, z, dB2] + QBERwdm[dB, z, dB2])
  frep ηduty[z] ηdead[dB, m, z, dB2];
QBER[dB_, m_, z_, dB2_] = (0.5 (QBERdark + QBERap[dB, m, z, dB2] + QBERwdm[dB, z, dB2]) +
  b m T[z] ηdemux[dB2] ηbob ηspad) /
  (m T[z] ηdemux[dB2] ηbob ηspad + QBERdark + QBERap[dB, m, z, dB2] + QBERwdm[dB, z, dB2]);
Iab[dB_, m_, z_, dB2_] = 1 -  $\frac{6}{5}$  (-QBER[dB, m, z, dB2] Log[2, QBER[dB, m, z, dB2]] -
  (1 - QBER[dB, m, z, dB2]) Log[2, 1 - QBER[dB, m, z, dB2]]);
Iae[m_, z_] =  $\frac{1}{1 + \frac{QBERdark}{m T[z] \eta spad}}$   $\left( \left( 1 - \frac{m}{2 T[z]} \right) \right.$ 
   $\left. \left( 1 - (-P[m, z] \text{Log}[2, P[m, z]] - (1 - P[m, z]) \text{Log}[2, 1 - P[m, z]]) \right) + \frac{m}{2 T[z]} \right)$ ;
P[m_, z_] = 0.5 +  $\sqrt{\Delta[m, z] (1 - \Delta[m, z])}$ ;
Δ[m_, z_] =  $\frac{1 - V}{2 - m / T[z]}$ ;
V = 1 - 2 b;

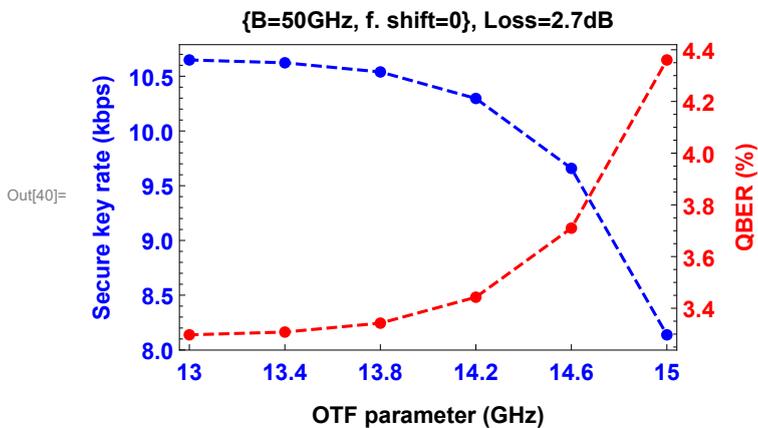
```

SKR and QBER calculations for z=4km fiber length and loss of DEMUX 2.7dB, given filtering isolation dB, carried out for different values of filter parameters. An example of finding filtering isolation for different OTF values is shown above.

```

In[38]:= plot1 = ListPlot[{{13, Rsec[-112.97376861836756`, μ[4], 4, 2.7] / 1000},
  {13.4, Rsec[-107.2464784343955`, μ[4], 4, 2.7] / 1000},
  {13.8, Rsec[-101.9841839116792`, μ[4], 4, 2.7] / 1000},
  {14.2, Rsec[-97.1361219555738`, μ[4], 4, 2.7] / 1000},
  {14.6, Rsec[-92.65833792763456`, μ[4], 4, 2.7] / 1000},
  {15, Rsec[-88.51261486131733`, μ[4], 4, 2.7] / 1000}}, Mesh → Full,
  Joined → True, Filling → None, PlotRange → Automatic, ImagePadding → 51,
  PlotStyle → {Blue, Dashed}, Frame → {True, True, False, False},
  FrameTicks → {{13, 13.4, 13.8, 14.2, 14.6, 15}, Automatic},
  FrameLabel → {Style["OTF parameter (GHz)", Black, FontSize → 12, Bold],
  Style["Secure key rate (kbps)", FontSize → 12]},
  LabelStyle → Directive[Blue, FontSize → 12, Bold]};
plot2 = ListPlot[{{13, 100 * QBER[-112.97376861836756`, μ[4], 4, 2.7]},
  {13.4, 100 * QBER[-107.2464784343955`, μ[4], 4, 2.7]},
  {13.8, 100 * QBER[-101.9841839116792`, μ[4], 4, 2.7]},
  {14.2, 100 * QBER[-97.1361219555738`, μ[4], 4, 2.7]},
  {14.6, 100 * QBER[-92.65833792763456`, μ[4], 4, 2.7]},
  {15, 100 * QBER[-88.51261486131733`, μ[4], 4, 2.7]}}, Mesh → Full, Joined → True,
  Filling → None, PlotRange → Automatic, ImagePadding → 51, PlotStyle → {Red, Dashed},
  Frame → {False, False, True, True}, FrameTicks → {{None, All}, {None, None}},
  FrameLabel → {None, None, Style["{B=50GHz, f. shift=0}, Loss=2.7dB", Black, Bold],
  Style["QBER (%)", FontSize → 12]},
  LabelStyle → Directive[Red, FontSize → 12, Bold]};
Overlay[{plot1, plot2}]

```

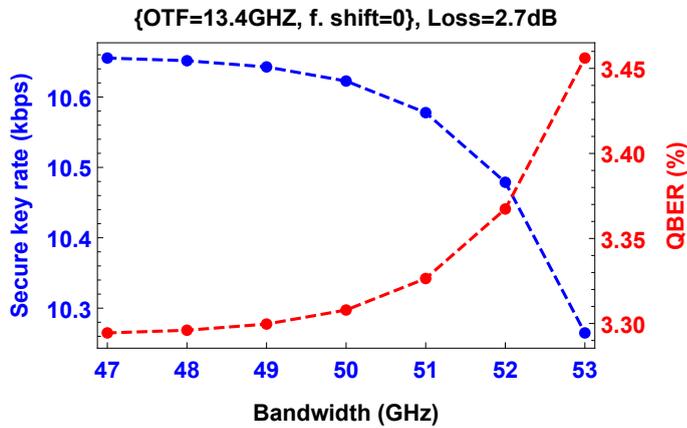


```

In[41]= plot3 = ListPlot[{{47, Rsec[-118.29688209999898`, μ[4], 4, 2.7] / 1000},
  {48, Rsec[-114.55006835550131`, μ[4], 4, 2.7] / 1000},
  {49, Rsec[-110.86663783922245`, μ[4], 4, 2.7] / 1000},
  {50, Rsec[-107.2464784343955`, μ[4], 4, 2.7] / 1000},
  {51, Rsec[-103.68947322568381`, μ[4], 4, 2.7] / 1000},
  {52, Rsec[-100.19550024527479`, μ[4], 4, 2.7] / 1000},
  {53, Rsec[-96.76443220370274`, μ[4], 4, 2.7] / 1000}}, Mesh → Full,
  Joined → True, Filling → None, PlotRange → Automatic, ImagePadding → 51,
  PlotStyle → {Blue, Dashed}, Frame → {True, True, False, False},
  FrameTicks → {{47, 48, 49, 50, 51, 52, 53}, Automatic},
  FrameLabel → {Style["Bandwidth (GHz)", Black, FontSize → 12, Bold],
  Style["Secure key rate (kbps)", FontSize → 12, Bold]},
  LabelStyle → Directive[Blue, FontSize → 12, Bold]};
plot4 = ListPlot[{{47, 100 * QBER[-118.29688209999898`, μ[4], 4, 2.7]},
  {48, 100 * QBER[-114.55006835550131`, μ[4], 4, 2.7]},
  {49, 100 * QBER[-110.86663783922245`, μ[4], 4, 2.7]},
  {50, 100 * QBER[-107.2464784343955`, μ[4], 4, 2.7]},
  {51, 100 * QBER[-103.68947322568381`, μ[4], 4, 2.7]},
  {52, 100 * QBER[-100.19550024527479`, μ[4], 4, 2.7]},
  {53, 100 * QBER[-96.76443220370274`, μ[4], 4, 2.7]}}, Mesh → Full,
  Joined → True, Filling → None, PlotRange → Automatic, ImagePadding → 51,
  PlotStyle → {Red, Dashed}, Frame → {False, False, True, True},
  FrameTicks → {{None, All}, {None, None}},
  FrameLabel → {None, None, Style["{OTF=13.4GHZ, f. shift=0}, Loss=2.7dB",
  Black, FontSize → 12, Bold], Style["QBER (%)", FontSize → 12, Bold]},
  LabelStyle → Directive[Red, FontSize → 12, Bold]};
Overlay[{plot3, plot4}]

```

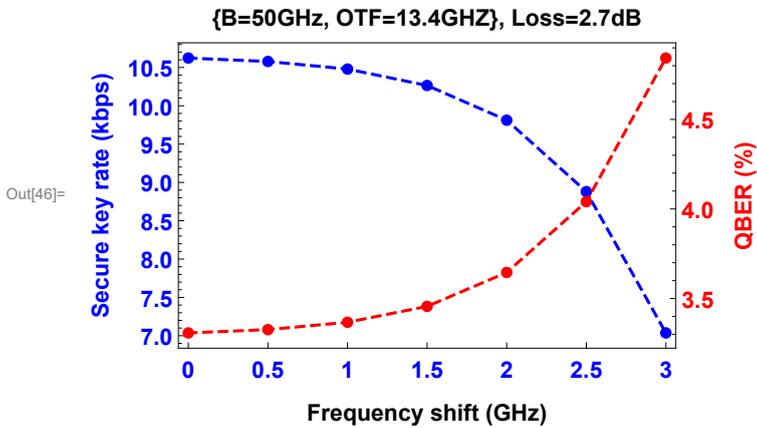
Out[43]=



```

In[44]= plot5 = ListPlot[{{0, Rsec[-107.2464784343955`, μ[4], 4, 2.7] / 1000},
  {0.5, Rsec[-103.68947322568381`, μ[4], 4, 2.7] / 1000},
  {1, Rsec[-100.19550024527479`, μ[4], 4, 2.7] / 1000},
  {1.5, Rsec[-96.76443220370274`, μ[4], 4, 2.7] / 1000},
  {2, Rsec[-93.39613620444082`, μ[4], 4, 2.7] / 1000},
  {2.5, Rsec[-90.09047344124821`, μ[4], 4, 2.7] / 1000},
  {3, Rsec[-86.84729887720303`, μ[4], 4, 2.7] / 1000}}, Mesh → Full,
  Joined → True, Filling → None, PlotRange → Automatic, ImagePadding → 51,
  PlotStyle → {Blue, Dashed}, Frame → {True, True, False, False},
  FrameTicks → {{0, 0.5, 1, 1.5, 2, 2.5, 3}, Automatic},
  FrameLabel → {Style["Frequency shift (GHz)", Black, FontSize → 12, Bold],
  Style["Secure key rate (kbps)", FontSize → 12, Bold]},
  LabelStyle → Directive[Blue, FontSize → 12, Bold]};
plot6 = ListPlot[{{0, 100 * QBER[-107.2464784343955`, μ[4], 4, 2.7]},
  {0.5, 100 * QBER[-103.68947322568381`, μ[4], 4, 2.7]},
  {1, 100 * QBER[-100.19550024527479`, μ[4], 4, 2.7]},
  {1.5, 100 * QBER[-96.76443220370274`, μ[4], 4, 2.7]},
  {2, 100 * QBER[-93.39613620444082`, μ[4], 4, 2.7]},
  {2.5, 100 * QBER[-90.09047344124821`, μ[4], 4, 2.7]},
  {3, 100 * QBER[-86.84729887720303`, μ[4], 4, 2.7]}}, Mesh → Full,
  Joined → True, Filling → None, PlotRange → Automatic, ImagePadding → 51,
  PlotStyle → {Red, Dashed}, Frame → {False, False, True, True},
  FrameTicks → {{None, All}, {None, None}},
  FrameLabel → {None, None, Style["{B=50GHz, OTF=13.4GHZ}, Loss=2.7dB", Black, Bold],
  Style["QBER (%)", FontSize → 12, Bold]},
  LabelStyle → Directive[Red, FontSize → 12, Bold]};
Overlay[{plot5, plot6}]

```



```

In[47]:= plot7 = ListPlot[{{2.7, Rsec[-107.2464784343955`, μ[4], 4, 2.7] / 1000},
    {3.1, Rsec[-107.2464784343955`, μ[4], 4, 3.1] / 1000},
    {3.5, Rsec[-107.2464784343955`, μ[4], 4, 3.5] / 1000},
    {3.9, Rsec[-107.2464784343955`, μ[4], 4, 3.9] / 1000},
    {4.3, Rsec[-107.2464784343955`, μ[4], 4, 4.3] / 1000},
    {4.7, Rsec[-107.2464784343955`, μ[4], 4, 4.7] / 1000}}, Mesh → Full,
    Joined → True, Filling → None, PlotRange → Automatic, ImagePadding → 51,
    PlotStyle → {Blue, Dashed}, Frame → {True, True, False, False},
    FrameTicks → {{2.7, 3.1, 3.5, 3.9, 4.3, 4.7}, Automatic},
    FrameLabel → {Style["Insertion loss (dB)", Black, FontSize → 12, Bold],
    Style["Secure key rate (kbps)", Black, FontSize → 12, Bold]},
    LabelStyle → Directive[Blue, FontSize → 12, Bold]};
plot8 = ListPlot[{{2.7, 100 * QBER[-107.2464784343955`, μ[4], 4, 2.7]},
    {3.1, 100 * QBER[-107.2464784343955`, μ[4], 4, 3.1]},
    {3.5, 100 * QBER[-107.2464784343955`, μ[4], 4, 3.5]},
    {3.9, 100 * QBER[-107.2464784343955`, μ[4], 4, 3.9]},
    {4.3, 100 * QBER[-107.2464784343955`, μ[4], 4, 4.3]},
    {4.7, 100 * QBER[-107.2464784343955`, μ[4], 4, 4.7]}}, Mesh → Full, Joined → True,
    Filling → None, PlotRange → Automatic, ImagePadding → 51, PlotStyle → {Red, Dashed},
    Frame → {False, False, True, True}, FrameTicks → {{None, All}, {None, None}},
    FrameLabel → {None, None, Style["{B=50GHZ, OTF=13.4GHZ, f. shift=0}", Black, Bold],
    Style["QBER (%)", Black, FontSize → 12, Bold]},
    LabelStyle → Directive[Red, FontSize → 12, Bold]};
Overlay[{plot7, plot8}]

```

Out[49]=

