In[270]:=
```
ClearAll["Global`*"];
$Version
SetDirectory[NotebookDirectory[]];
(*Mathematica file in the proper location*)
```
Out[271]=
```
12.3.0 for Mac OS X x86 (64-bit) (May 10, 2021)
```

Crossed-polarization (CP) optical microscopy intensity calculation

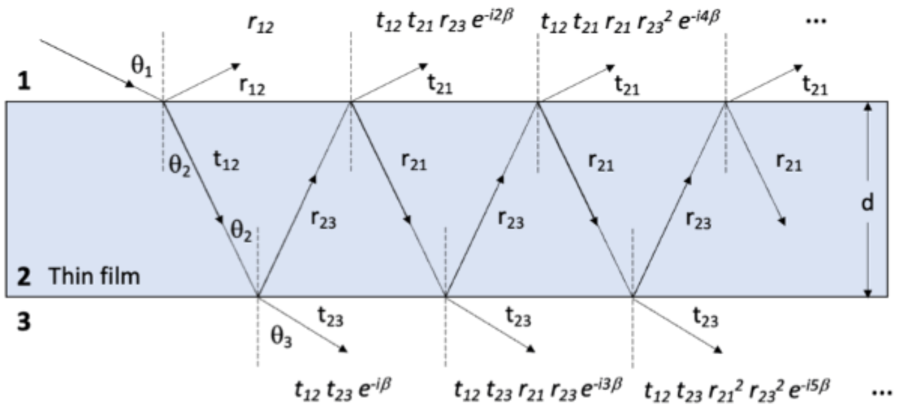
A. Input

In[273]:=
```
d = 3.5;(*nm, CNC diameter*)
n1 = 1.33;(*water*)
n3 = 1.33; (*water*)
(*
Refractive index of cellulose crystal from Reference53: Chae I,
et al.Anisotropic Optical and Frictional Properties of Langmuir–
Blodgett Film Consisting of Uniaxially-Aligned Rod-
Shaped Cellulose Nanocrystals.Advanced Materials Interfaces 7,(2020).
*)
nval = Import["nenodn.xlsx"];
Dimensions[nval]
nval // TableForm

ll = Table[nval⟦1, i, 1⟧, {i, 2, 17}];
nol = Table[nval⟦1, i, 2⟧, {i, 2, 17}];
nel = Table[nval⟦1, i, 3⟧, {i, 2, 17}];
```
Out[277]=
```
{1, 17, 4}
```
Out[278]//TableForm=

lamda	250.	300.	350.	400.	450.	500.	550.
no	1.55726	1.52514	1.50931	1.50001	1.494	1.48986	1.48688
ne	1.76155	1.7146	1.691	1.67701	1.66794	1.66168	1.65717
dn	0.20429	0.18946	0.18169	0.177	0.17394	0.17182	0.17029

B. Optical properties

CNC is considered as a thin film



In[282]:=
```
θ1 = N[θ * π/180]; (*rad*)

n2p = nel;
n2s = nol;

θ2p = ArcSin[n1 Sin[θ1] / n2p] ; (*Snell's law at the interface between 1 & 2*)
θ3p = ArcSin[n2p Sin[θ2p] / n3] ;(*Snell's law at the interface between 1 & 2*)
θ2s = ArcSin[n1 Sin[θ1] / n2s] ; (*Snell's law at the interface between 1 & 2*)
θ3s = ArcSin[n2s Sin[θ2s] / n3] ;(*Snell's law at the interface between 1 & 2*)

βp = (2 π/ll) d * n2p * Cos[θ2p] ;(* β=(2πd/λ) n2*cosθ - Equation 2 *)
βs = (2 π/ll) d * n2s * Cos[θ2s] ;(* β=(2πd/λ) n2*cosθ - Equation 2 *)

(*"thin film equations*)
t12p = (2 * Cos[θ1] * Sin[θ2p]) / (Sin[θ1 + θ2p] Cos[θ1 - θ2p]);

t12s = (2 * Cos[θ1] * Sin[θ2s]) / (Sin[θ1 + θ2s]);

t23p = (2 * Cos[θ2p] * Sin[θ3p]) / (Sin[θ2p + θ3p] Cos[θ2p - θ3p]);

t23s = (2 * Cos[θ2s] * Sin[θ3s]) / (Sin[θ2s + θ3s]);

r12p = (Tan[θ1 - θ2p]) / (Tan[θ1 + θ2p]);
```

$$r12s = -\frac{\text{Sin}[\theta 1 - \theta 2s]}{\text{Sin}[\theta 1 + \theta 2s]};$$

$$r23p = \frac{\text{Tan}[\theta 2p - \theta 3p]}{\text{Tan}[\theta 2p + \theta 3p]};$$

$$r23s = -\frac{\text{Sin}[\theta 2s - \theta 3s]}{\text{Sin}[\theta 2s + \theta 3s]};$$

$$t123p = \frac{t12p * t23p\ \text{Exp}[-\ I * \beta p]}{1 + r12p * r23p\ \text{Exp}[-I * 2\ \beta p]};\ (*\ t_{123}=\frac{t_{12}*t_{23}\ \text{Exp}[-i\beta]}{1+r_{12}*r_{23}\ \text{Exp}[-i2\beta]}\ -\ \text{Equation 1}\ *)$$

$$t123s = \frac{t12s * t23s\ \text{Exp}[-\ I * \beta s]}{1 + r12s * r23s\ \text{Exp}[-I * 2\ \beta s]};\ (*\ t_{123}=\frac{t_{12}*t_{23}\ \text{Exp}[-i\beta]}{1+r_{12}*r_{23}\ \text{Exp}[-i2\beta]}\ -\ \text{Equation 1}\ *)$$

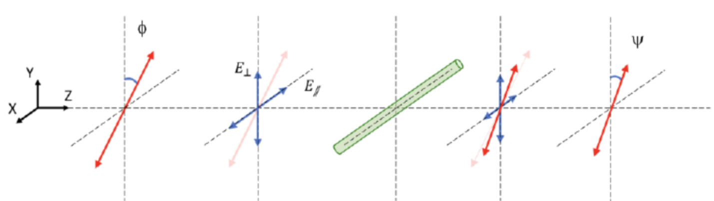
```
T123p = t123p * Conjugate[t123p];
T123s = t123s * Conjugate[t123s];
```

C. Light properties after passing through a single CNC (add figure)

- Electric field, intensity and polarization angle as a function of the polarization angle with respect to the CNC angle

(* Supplementary Figure 8a *)

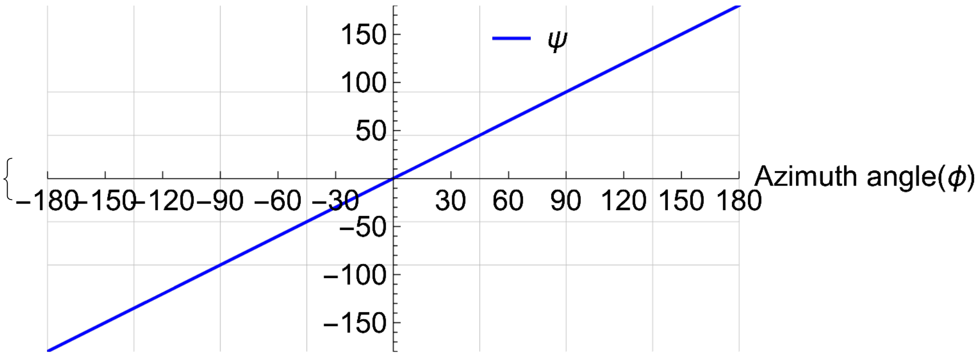
Ep = $E_{\parallel}$, Es = $E_{\perp}$



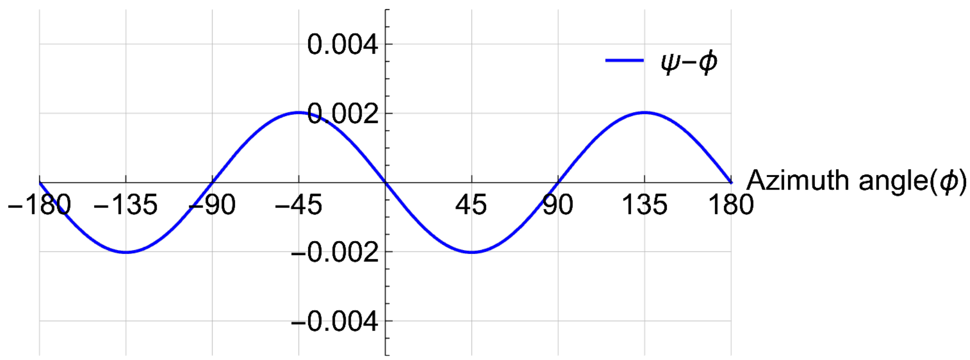
In[303]:=
```
(*ϵ= ~0 *)
ϵ = 10^-9;
θ = ϵ (*deg*);

(*Incident beam Ep0, Es0, Ip0, Is0*)
Ep0[ϕ_] := 1 Sin[ϕ * (π / 180)];
Es0[ϕ_] := 1 Cos[ϕ * (π / 180)];
Ip0[ϕ_] := Ep0[ϕ]^2;
Is0[ϕ_] := Es0[ϕ]^2;

(* Output beam- Ep1, Es1, Ip1, Is1 after a single CNC *)
Ep[ϕ_] := Ep0[ϕ] * t123p[[8]];(*[[8]]=600nm *)
Es[ϕ_] := Es0[ϕ] * t123s[[8]];(*[[8]]=600nm *)
Ip[ϕ_] := Ep[ϕ] * Conjugate[Ep[ϕ]];
Is[ϕ_] := Es[ϕ] * Conjugate[Es[ϕ]];

(* Output beam- ψ, (ψ-ϕ) (ϕ dependence) *)
ψ[ϕ_] := Piecewise[{
   {ArcTan[Sqrt[Ep[ϕ] * Conjugate[Ep[ϕ]]] / Sqrt[Es[ϕ] * Conjugate[Es[ϕ]]]] * 180/π - 180,
    ϕ < -90},
   {ArcTan[Sqrt[Ep[ϕ] * Conjugate[Ep[ϕ]]] / - Sqrt[Es[ϕ] * Conjugate[Es[ϕ]]]] * 180/π + 180,
    ϕ > 90},
   {ArcTan[Sqrt[Ep[ϕ] * Conjugate[Ep[ϕ]]] / Sqrt[Es[ϕ] * Conjugate[Es[ϕ]]]] * 180/π, ϕ > 0},
   {ArcTan[Sqrt[Ep[ϕ] * Conjugate[Ep[ϕ]]] / - Sqrt[Es[ϕ] * Conjugate[Es[ϕ]]]] * 180/π, ϕ > -90}
  }] (* ψ=tan^-1[E∥/E⊥] - Equation 3 *)

(*Output beam |Etot| and Itot*)
Et[ϕ_] := Sqrt[Ep[ϕ] * Conjugate[Ep[ϕ]] + Es[ϕ] * Conjugate[Es[ϕ]]] ;
(* |Etotal| = Sqrt[E∥*E∥*+E⊥*E⊥*]  - Equation 4 *)
It[ϕ_] := Et[ϕ] * Conjugate[Et[ϕ]];

(*transmission coefficient*)
tt[ϕ_] := Et[ϕ] / 1;
```

```
In[317]:= (*check plots*)
{Plot[ψ[ϕ], {ϕ, -180, 180}, PlotRange → {{-180, 180}, {-180, 180}},
   Ticks → {Table[30 i, {i, -12, 12}], Automatic},
   GridLines → {Table[45 i, {i, -4, 4}], {-90, -45, 45, 90}},
   PlotLegends → Placed[{"ψ"}, {0.7, 0.9}], PlotStyle → {Blue},
   AxesLabel → {"Azimuth angle(ϕ)", None}, ImageSize → 500, AspectRatio → 1 / 2,
   LabelStyle → {FontFamily → "Arial", FontSize → 15, Black}],
  Plot[{ψ[ϕ] - ϕ}, {ϕ, -180, 180}, PlotRange → {{-180, 180}, {-0.005, 0.005}},
   Ticks → {Table[45 i, {i, -12, 12}], Automatic},
   GridLines → {Table[45 i, {i, -12, 12}], Automatic},
   PlotLegends → Placed[{"ψ-ϕ"}, {0.9, 0.85}], PlotStyle → {Blue},
   AxesLabel → {"Azimuth angle(ϕ)", None}, ImageSize → 500, AspectRatio → 1 / 2,
   LabelStyle → {FontFamily → "Arial", FontSize → 15, Black}]
   (* Supplementary Figure 8b *)
}

{Plot[Et[ϕ], {ϕ, -180, 180}, PlotRange → {{-180, 180}, All},
   Ticks → {Table[30 i, {i, -12, 12}], Automatic},
   GridLines → {Table[45 i, {i, -12, 12}], Automatic},
   PlotLegends → Placed[{"|Et|"}, {0.84, 0.87}], PlotStyle → {Blue},
   AxesLabel → {"  Azimuth ang(ϕ)", None}, ImageSize → 500, AspectRatio → 1 / 2,
   LabelStyle → {FontFamily → "Arial", FontSize → 15, Black}]
   (* Supplementary Figure 8c *),
  Plot[It[ϕ], {ϕ, -180, 180}, PlotRange → {{-180, 180}, All},
   Ticks → {Table[30 i, {i, -12, 12}], Automatic},
   GridLines → {Table[45 i, {i, -12, 12}], Automatic},
   PlotLegends → Placed[{"It"}, {0.84, 0.87}], PlotStyle → {Blue},
   AxesLabel → {"  Azimuth ang(ϕ)", None}, ImageSize → 500, AspectRatio → 1 / 2,
   LabelStyle → {FontFamily → "Arial", FontSize → 15, Black}]}

Plot[tt[ϕ], {ϕ, -180, 180}, PlotRange → {{-180, 180}, All},
  Ticks → {Table[30 i, {i, -12, 12}], Automatic},
  GridLines → {Table[45 i, {i, -12, 12}], Automatic},
  PlotLegends → Placed[{"ttot"}, {0.84, 0.87}], PlotStyle → {Blue},
  AxesLabel → {"  Azimuth ang(ϕ)", None}, ImageSize → 500, AspectRatio → 1 / 2,
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black}]
```

Out[317]= {

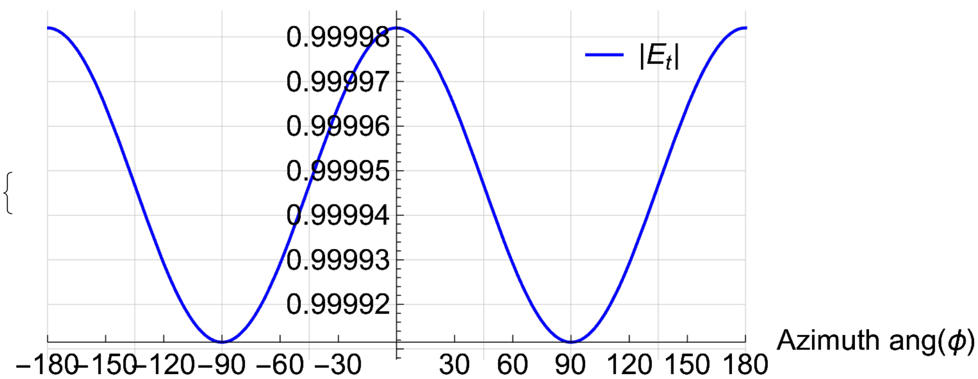


,

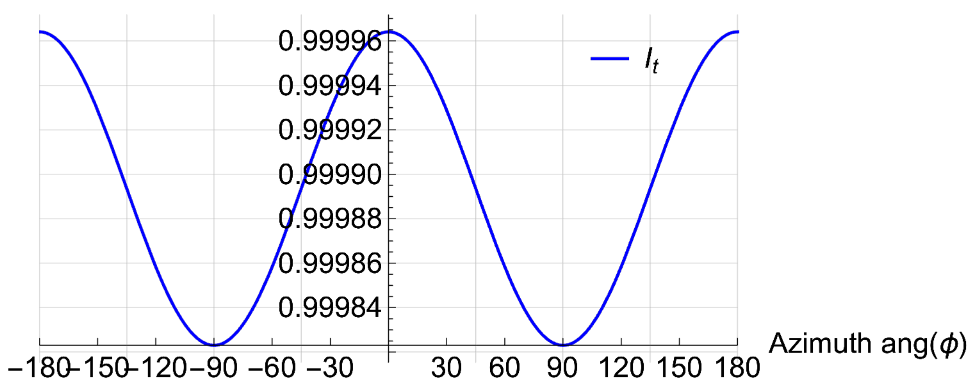


}

Out[318]= {

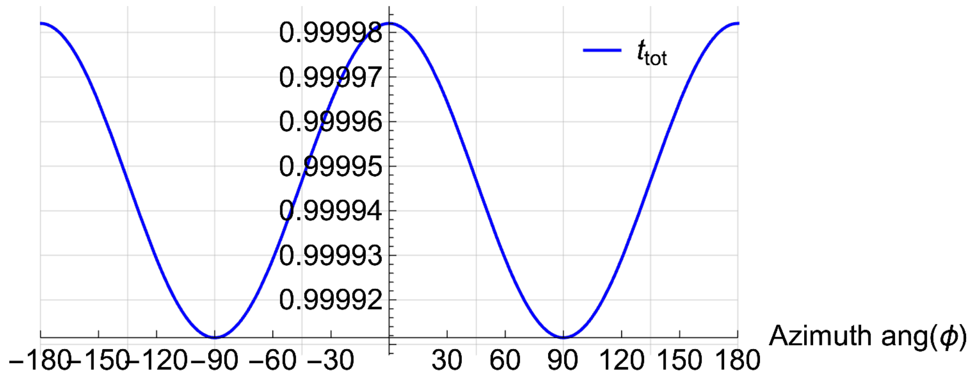


,



}

Out[319]=

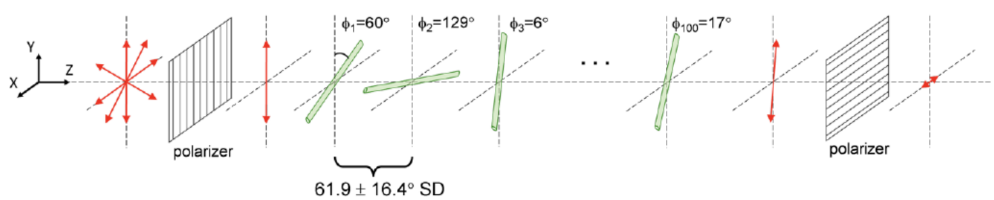


D. 100 CNC angle generation for the five CMF organizations: crossed-polylamellate, random, helicoidal (10°), uniaxial, and biomodal.

In[320]:=
```
n = 100;(* number of CNCs (layers)*)
ϕ1 = 60; (* angle of the first CNC*)
(*60 deg was chosen as an example - Supplementary Figure 8 *)
ϕ1 = ϕ1;(* ϵ is a small number, 10^-9 *)

(*defined functions*)
F360[x_] := x - Floor[x, 360];
F180[x_] := x - Floor[x, 180];
F180off[x_] := N[ArcSin[Sin[x * π/180]] * 180/π];
F90off[x_] := N[ArcCos[Cos[x * π/180]] * 180/π];
```

(* Supplementary Figure 8d *)

Crossed-polylamellate structure as an example



```
In[327]:= (* Crossed-polylamellate structure*)
φcl = Table[0, {i, 1, n}];
φcl[[1]] = φ1;
dφcl = Table[0, {i, 1, n}];
dφcl[[1]] = φ1;

For[i = 1, i < n, i++,
 φcl[[i + 1]] = Round[
   φcl[[i]] + RandomVariate[NormalDistribution[61.9, 15.4]] * (-1)^RandomInteger[{1,10}]];
 dφcl[[i + 1]] = φcl[[i + 1]] - φcl[[i]];
]

φcl;
φcl360 = F360[φcl]
φcl180 = F180[φcl];
φcl180Norm = F180[φcl180 + 90];

Out[333]= {60, 328, 261, 189, 102, 137, 187, 132, 200, 146, 223, 145, 53, 22, 293, 232,
 283, 230, 287, 242, 179, 119, 182, 250, 309, 226, 277, 181, 250, 311, 353, 279,
 223, 174, 218, 162, 231, 169, 119, 174, 211, 148, 78, 114, 59, 1, 290, 351,
 271, 331, 276, 358, 53, 346, 56, 3, 71, 149, 204, 249, 321, 23, 84, 7, 68, 121,
 185, 123, 176, 236, 275, 191, 125, 69, 23, 333, 14, 76, 30, 308, 258, 314, 239,
 154, 87, 27, 70, 131, 214, 117, 155, 125, 53, 351, 321, 269, 331, 32, 328, 267}
```

In[336]:=
```
(* random orientation *)
ϕra = Round[RandomReal[{0, 360}, n]];
ϕra[[1]] = ϕ1;

dϕra = Table[0, {i, 1, n}];
dϕra[[1]] = ϕ1;

For[i = 1, i < n, i++,
 dϕra[[i + 1]] = ϕra[[i + 1]] - ϕra[[i]];
]

ϕra;
ϕra360 = F360[ϕra]
ϕra180 = F180[ϕra];
ϕra180Norm = F180[ϕra180 + 90];
```

Out[342]=
```
{60, 185, 51, 58, 288, 286, 231, 63, 65, 115, 231, 73, 129, 159, 154, 250, 54,
 296, 256, 310, 242, 139, 122, 301, 297, 71, 65, 69, 55, 19, 339, 136, 160, 181,
 281, 46, 170, 36, 172, 25, 175, 145, 21, 47, 240, 301, 256, 19, 137, 6, 261,
 22, 303, 333, 143, 63, 75, 323, 264, 28, 8, 165, 13, 113, 268, 342, 112, 307,
 331, 137, 130, 345, 91, 288, 187, 288, 191, 235, 337, 140, 64, 88, 13, 271,
 130, 22, 192, 216, 157, 301, 315, 144, 7, 218, 323, 109, 222, 297, 94, 103}
```

In[345]:=
```
(* helicoidal 10° *)
ϕhe = Table[0, {i, 1, n}];
ϕhe[[1]] = ϕ1;

For[i = 1, i < n, i++,
 ϕhe[[i + 1]] = ϕhe[[i]] + 10
]
ϕhe;
ϕhe360 = F360[ϕhe]
ϕhe180 = F180[ϕhe];
ϕhe180Norm = F180[ϕhe180 + 90];
```

Out[349]=
```
{60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210, 220,
 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350, 0, 10, 20,
 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190,
 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330, 340, 350,
 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170,
 180, 190, 200, 210, 220, 230, 240, 250, 260, 270, 280, 290, 300, 310, 320, 330}
```

In[352]:=
```
(* uniaxial 10° *)
ϕun = Round[RandomVariate[NormalDistribution[90, 10], n], 1]
ϕun;
ϕun360 = F360[ϕun];
ϕun180 = F180[ϕun];
ϕun180Norm = F180[ϕun180 + 90];
```

Out[352]=
```
{78, 85, 86, 86, 89, 111, 91, 102, 107, 78, 97, 93, 107, 83, 84, 84, 98, 65, 91, 80, 102,
 84, 82, 81, 89, 94, 101, 107, 104, 114, 96, 85, 77, 90, 84, 85, 87, 94, 89, 94,
 89, 69, 88, 89, 92, 87, 86, 98, 101, 90, 86, 95, 88, 65, 100, 78, 91, 86, 93, 87,
 90, 89, 82, 94, 81, 86, 91, 103, 85, 87, 105, 82, 77, 99, 90, 89, 90, 112, 98, 89,
 100, 89, 85, 87, 108, 97, 106, 78, 84, 91, 91, 84, 83, 107, 90, 94, 91, 93, 80, 87}
```

In[357]:=
```
(* bimodal ° *)
ϕbi = RandomVariate[MixtureDistribution[{1, 1},
    {NormalDistribution[42, 8], NormalDistribution[135, 10]}], n];
ϕbi = Round[ϕbi, 1]
ϕbi360 = F360[ϕbi];
ϕbi180 = F180[ϕbi];
ϕbi180Norm = F180[ϕbi180 + 90];
```

Out[358]=
```
{29, 21, 149, 41, 46, 36, 123, 126, 40, 31, 136, 50, 41, 44, 139, 145, 40,
 116, 34, 149, 38, 49, 138, 119, 146, 137, 134, 154, 51, 130, 128, 142, 128,
 39, 131, 141, 132, 141, 138, 145, 34, 134, 130, 120, 45, 34, 40, 143, 48,
 146, 39, 139, 138, 45, 46, 28, 32, 124, 46, 37, 35, 142, 43, 37, 134, 132,
 143, 35, 29, 56, 121, 129, 132, 34, 48, 39, 145, 130, 41, 134, 45, 126, 52,
 138, 127, 22, 139, 136, 123, 53, 113, 51, 154, 128, 31, 41, 42, 146, 135, 41}
```

E. Light properties after the 100 CNC organizations - Loop1 for each organization

In[362]:=
```
ϕi = 0;(* the initial polarization direction of the incident beam w.r.t. y-axis *)
(*0 deg was chosen as an example - Supplementary Figure 8 *)
ϕi = ϕi + ϵ;(* ϵ is a small number, 10^-9 *)
```

1) Cross-polylamellate

Storage

Loop

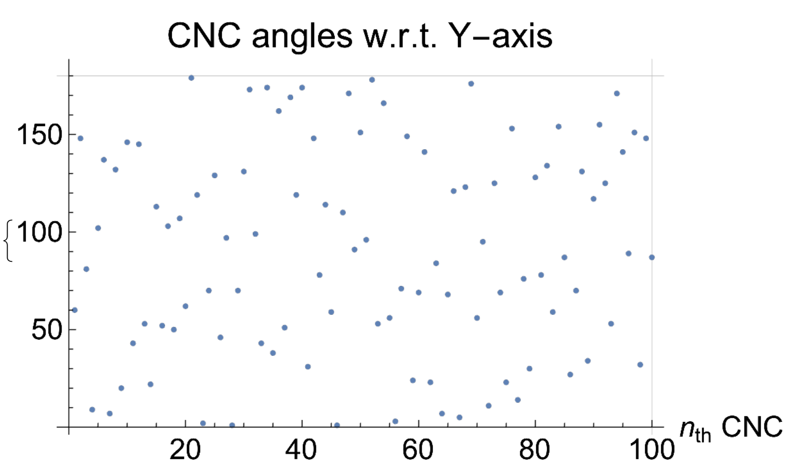
In[380]:=
```
For[p = 1, p < n + 1, p++,

 ϕbtn[[p]] = ψy[[p]] - ϕcl180Norm[[p]]; (* convert axis: From y-axis to CNC normal *)
 ψres[[p]] = ψ[ϕbtn[[p]]]; (* result of rotation in CNC normal axis*)
 ψy[[p + 1]] = ψres[[p]] + ϕcl180Norm[[p]];
 (* back to y-axis system*) (*F180 is for *)

 dψ[[p]] = ψres[[p]] - ϕbtn[[p]]; (* (ψafter - ψbefore) in CNC axis *)

 ttot[[p]] = tt[ϕbtn[[p]]];
 Etot[[p + 1]] = Etot[[p]] * ttot[[p]]; (* energy loss as passing through *)

 dψ2[[p]] = ψy[[p + 1]] - ψy[[p]];
 (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)
]
```

$$PI = \left(Etot[[n+1]] * Sin\left[(\psi y[[n+1]] - \phi 1) * \frac{\pi}{180}\right]\right)^2; \text{(* } I = (|E_{total}| * \sin\psi)^2 \text{ - Equation 5 *)}$$

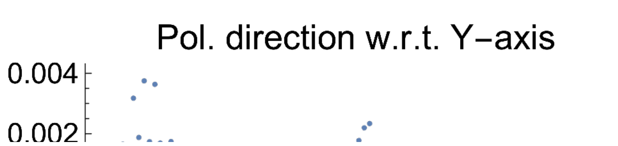
Check plots and final intensity (Supplementary Figure 8e,f here)

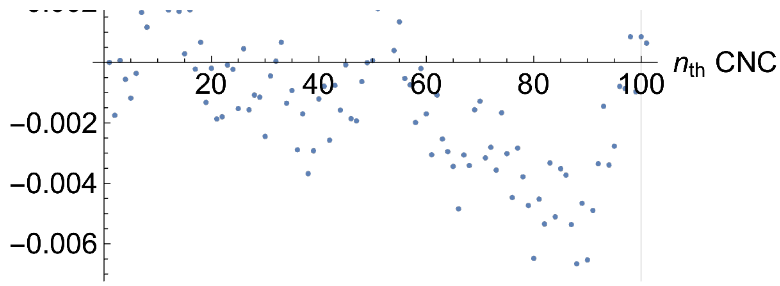
In[382]:=
```
{ListPlot[ϕcl180, ImageSize → 400, PlotRange → All, AxesLabel → {"n_th CNC", None},
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  GridLines → {{100}, {180}}, AxesOrigin → {0, 0},
  PlotLabel → "CNC angles w.r.t. Y-axis"](* Supplementary Figure 8e *),
 ListPlot[ψy, ImageSize → 400, PlotRange → All,
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  AxesLabel → {"n_th CNC", None}, PlotLabel → "Pol. direction w.r.t. Y-axis",
  GridLines → {{100}, None}](* Supplementary Figure 8f *),
 ListPlot[Etot, ImageSize → 400, PlotRange → All,
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  AxesLabel → {"n_th CNC", None}, PlotLabel → "|E_total|", GridLines → {{100}, None}],
 ListPlot[(dψ), ImageSize → 400, PlotRange → All, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"n_th CNC", None},
  PlotLabel → "Rotation (ψ-ψ0)", GridLines → {{100}, None}]}
{ListPlot[ϕbtn, ImageSize → 400, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"n_th CNC", None},
  PlotLabel → "Pol. direction w.r.t. CNC normal (before)",
  GridLines → {{100}, None}],
 ListPlot[ψres, ImageSize → 400, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"n_th CNC", None},
  PlotLabel → "Pol. direction w.r.t. CNC normal (after)",
  GridLines → {{100}, None}]}

Print["E_total after 100th CNC=", Etot[[n + 1]]]
Print["Final intensity after the 2nd polarizer=", PI]
```

Out[382]= {

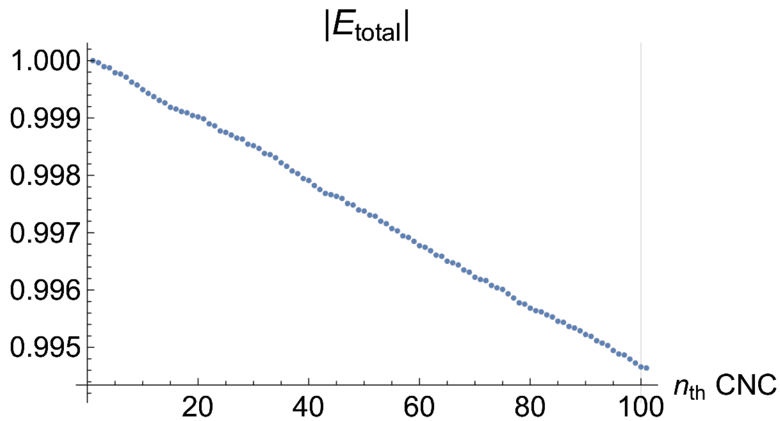


,

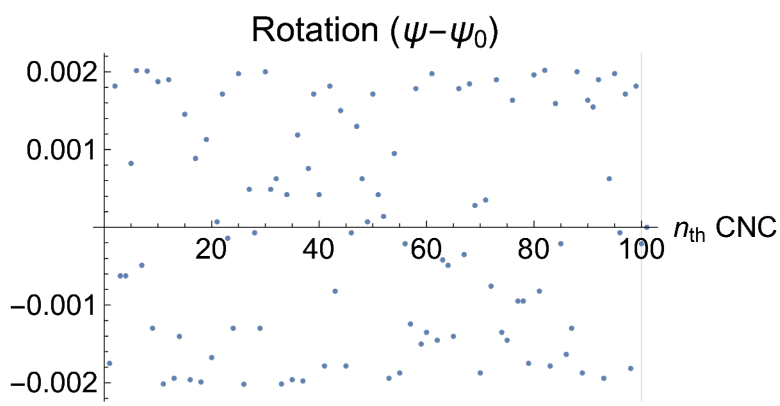



−0.002
−0.004
−0.006
20
40
60
80
100
n_{th} CNC

,

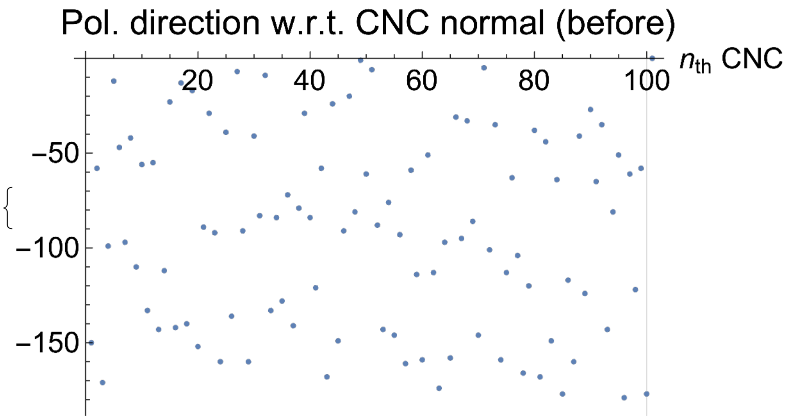

$|E_{total}|$
1.000
0.999
0.998
0.997
0.996
0.995
20
40
60
80
100
n_{th} CNC

,

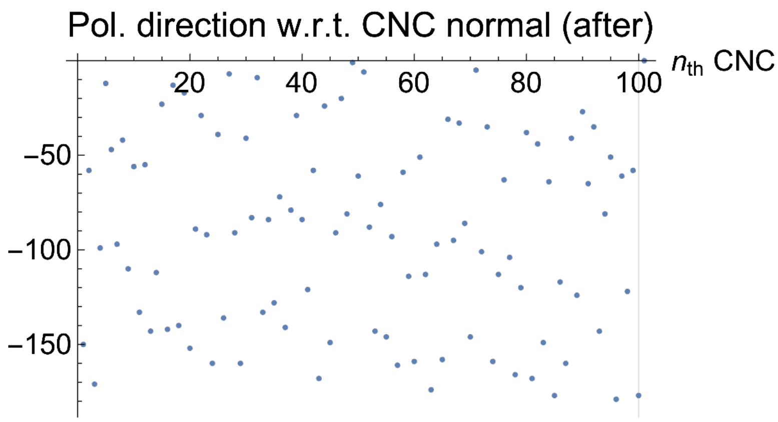

Rotation $(\psi-\psi_0)$
0.002
0.001
−0.001
−0.002
20
40
60
80
100
n_{th} CNC

}

Out[383]= {



,



}

E_{total} after 100th CNC=0.994641 + 0. i

Final intensity after the 2nd polarizer=0.741974 + 0. i

Check numbers after each CNC

2) Random

Storage

Loop

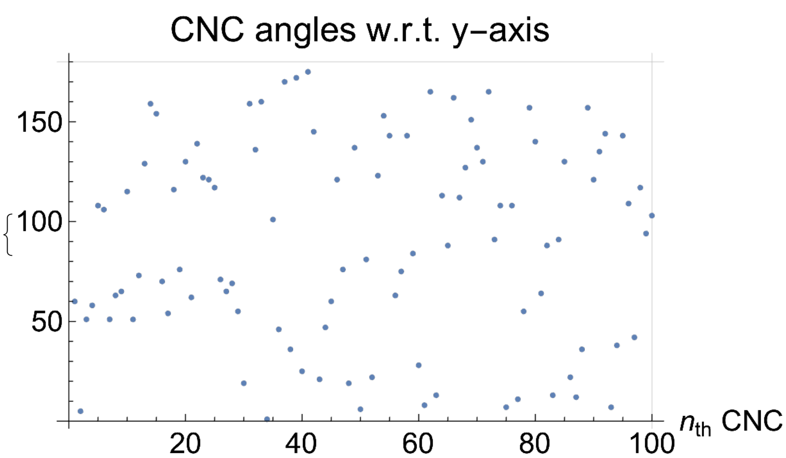
In[407]:=
```
For[p = 1, p < n + 1, p++,

 ϕbtn⟦p⟧ = ψy⟦p⟧ - ϕra180Norm⟦p⟧; (* convert axis: From y-axis to CNC normal *)
 ψres⟦p⟧ = ψ[ϕbtn⟦p⟧]; (* result of rotation in CNC normal axis*)
 ψy⟦p + 1⟧ = ψres⟦p⟧ + ϕra180Norm⟦p⟧;
 (* back to y-axis system*) (*F180 is for *)

 dψ⟦p⟧ = ψres⟦p⟧ - ϕbtn⟦p⟧; (* (ψafter - ψbefore) in CNC axis *)

 ttot⟦p⟧ = tt[ϕbtn⟦p⟧];
 Etot⟦p + 1⟧ = Etot⟦p⟧ * ttot⟦p⟧; (* energy loss as passing through *)

 dψ2⟦p⟧ = ψy⟦p + 1⟧ - ψy⟦p⟧;
 (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)
]
```

$$PI = \left(Etot[[n+1]] * Sin\left[(\psi y[[n+1]] - \phi 1) * \frac{\pi}{180}\right]\right)^2;$$ (* $I = (|E_{total}| * \sin\psi)^2$ - Equation 5 *)

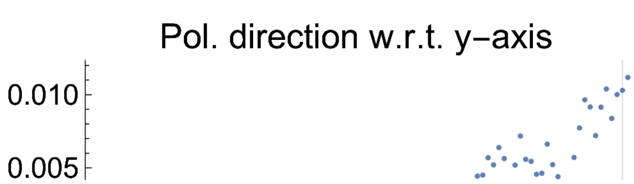
Check plots and final intensity

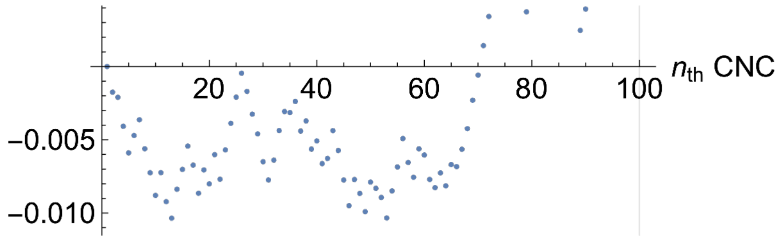
In[409]:=
```
{ListPlot[ϕra180, ImageSize → 400, PlotRange → All, AxesLabel → {"nth CNC", None},
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  GridLines → {{100}, {180}}, AxesOrigin → {0, 0},
  PlotLabel → "CNC angles w.r.t. y-axis"],
 ListPlot[ψy, ImageSize → 400, PlotRange → All, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
  PlotLabel → "Pol. direction w.r.t. y-axis", GridLines → {{100}, None}],
 ListPlot[Etot, ImageSize → 400, PlotRange → All,
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  AxesLabel → {"nth CNC", None}, PlotLabel → "|Etotal|", GridLines → {{100}, None}],
 ListPlot[(dψ), ImageSize → 400, PlotRange → All, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
  PlotLabel → "Pol. rotation", GridLines → {{100}, None}]}
{ListPlot[ϕbtn, ImageSize → 400, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
  PlotLabel → "Pol. direction w.r.t. CNC normal (before)",
  GridLines → {{100}, None}],
 ListPlot[ψres, ImageSize → 400, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
  PlotLabel → "Pol. direction w.r.t. CNC normal (after)",
  GridLines → {{100}, None}]}

Print["Etotal after 100th CNC=", Etot[[n + 1]]]
Print["Final intensity after the 2nd polarizer=", PI]
```

Out[409]= {



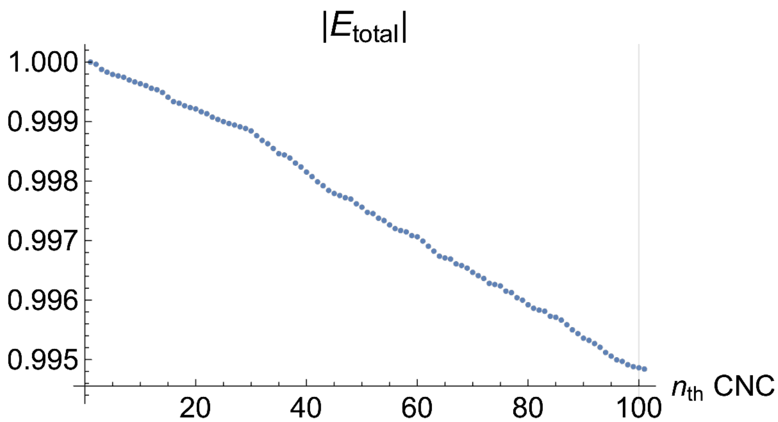
,





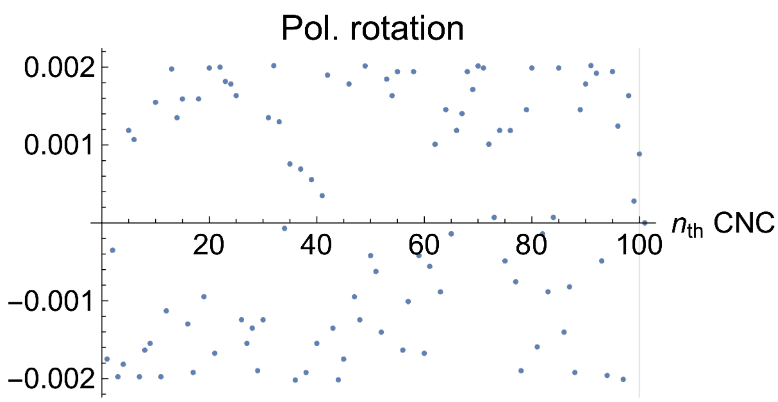
-0.005
-0.010
20
40
60
80
100
n_{th} CNC

,



$|E_{total}|$
1.000
0.999
0.998
0.997
0.996
0.995
20
40
60
80
100
n_{th} CNC

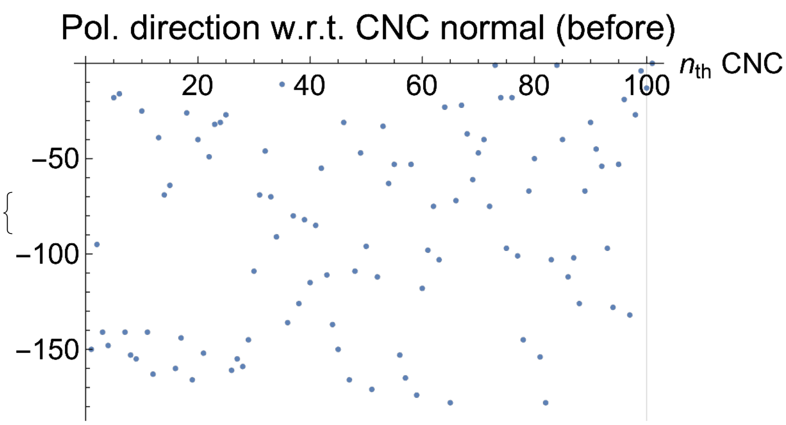
,



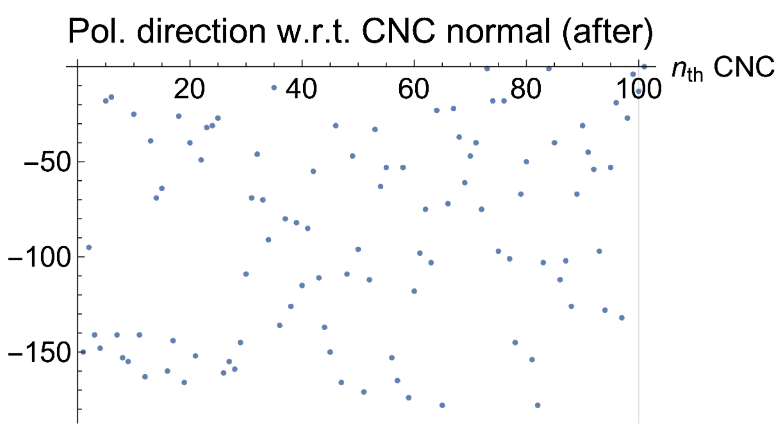
Pol. rotation
0.002
0.001
-0.001
-0.002
20
40
60
80
100
n_{th} CNC

}

Out[410]= {



,



}

E_{total} after 100th CNC=0.994841 + 0. i

Final intensity after the 2nd polarizer=0.742114 + 0. i

Check numbers after each CNC

3) Helicoidal

Storage

Loop

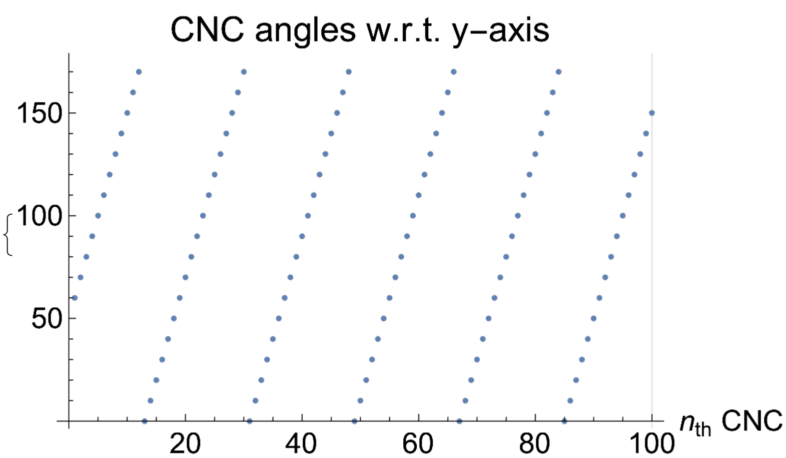
In[434]:=
```
For[p = 1, p < n + 1, p++,

 ϕbtn[[p]] = ψy[[p]] - ϕhe180Norm[[p]]; (* convert axis: From y-axis to CNC normal *)
 ψres[[p]] = ψ[ϕbtn[[p]]]; (* result of rotation in CNC normal axis*)
 ψy[[p + 1]] = ψres[[p]] + ϕhe180Norm[[p]];
 (* back to y-axis system*) (*F180 is for *)

 dψ[[p]] = ψres[[p]] - ϕbtn[[p]]; (* (ψafter - ψbefore) in CNC axis *)

 ttot[[p]] = tt[ϕbtn[[p]]];
 Etot[[p + 1]] = Etot[[p]] * ttot[[p]]; (* energy loss as passing through *)

 dψ2[[p]] = ψy[[p + 1]] - ψy[[p]];
 (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)
]
```

$$PI = \left(Etot[[n+1]] * Sin\left[(\psi y[[n+1]] - \phi 1) * \frac{\pi}{180}\right]\right)^2;$$ (* $I = (|E_{total}| * \sin\psi)^2$ - Equation 5 *)

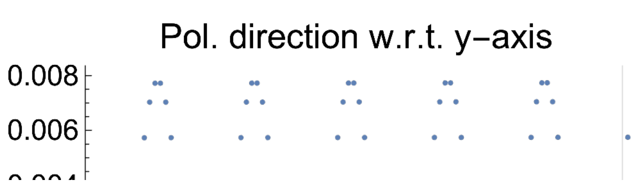
Check plots and final intensity

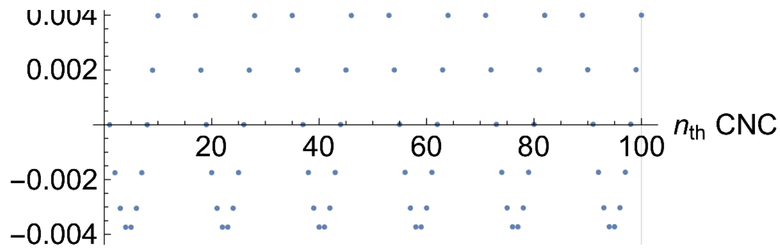
In[436]:=
```
{ListPlot[ϕhe180, ImageSize → 400, PlotRange → All, AxesLabel → {"nth CNC", None},
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  GridLines → {{100}, {180}}, AxesOrigin → {0, 0},
  PlotLabel → "CNC angles w.r.t. y-axis"],
 ListPlot[ψy, ImageSize → 400, PlotRange → All, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
  PlotLabel → "Pol. direction w.r.t. y-axis", GridLines → {{100}, None}],
 ListPlot[Etot, ImageSize → 400, PlotRange → All,
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  AxesLabel → {"nth CNC", None}, PlotLabel → "|Etotal|", GridLines → {{100}, None}],
 ListPlot[(dψ), ImageSize → 400, PlotRange → All, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
  PlotLabel → "Pol. rotation", GridLines → {{100}, None}]}
{ListPlot[ϕbtn, ImageSize → 400, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
  PlotLabel → "Pol. direction w.r.t. CNC normal (before)",
  GridLines → {{100}, None}],
 ListPlot[ψres, ImageSize → 400, LabelStyle →
   {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
  PlotLabel → "Pol. direction w.r.t. CNC normal (after)",
  GridLines → {{100}, None}]}

Print["Etotal after 100th CNC=", Etot[[n + 1]]]
Print["Final intensity after the 2nd polarizer=", PI]
```

Out[436]= {

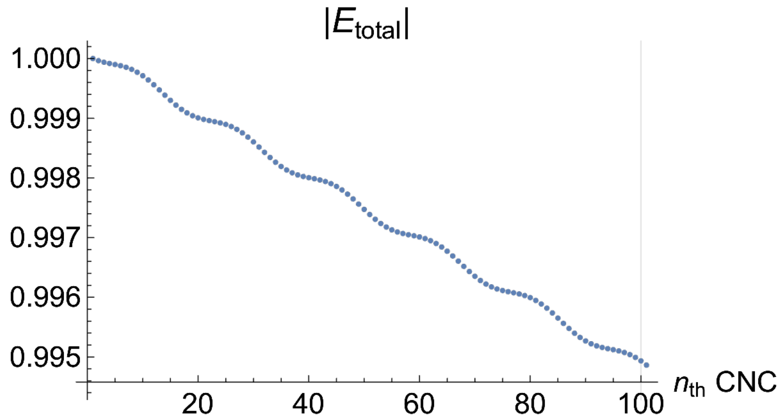


,

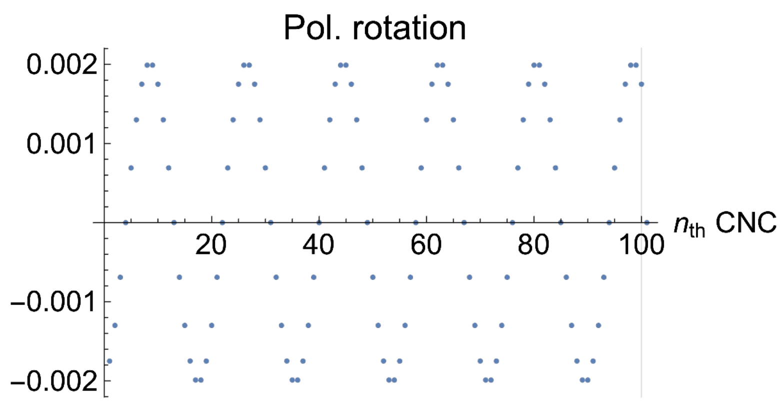



0.004
0.002
-0.002
-0.004
20
40
60
80
100
n_{th} CNC

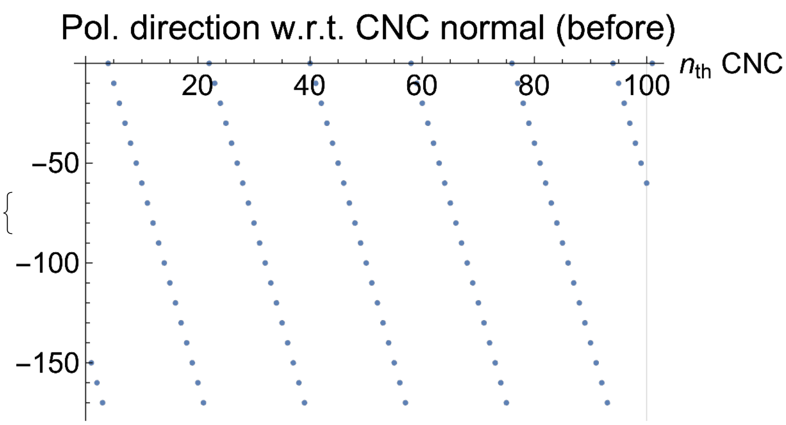
,

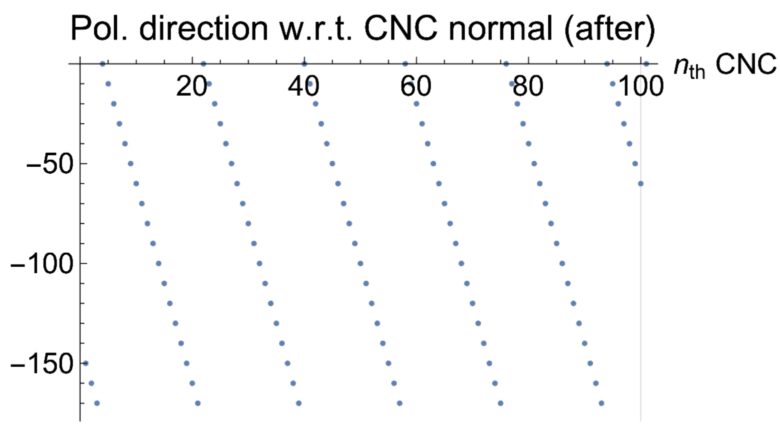

$|E_{total}|$
1.000
0.999
0.998
0.997
0.996
0.995
20
40
60
80
100
n_{th} CNC

,


Pol. rotation
0.002
0.001
-0.001
-0.002
20
40
60
80
100
n_{th} CNC

}

Out[437]= {

,


}

E_{total} after 100th CNC=0.994863 + 0. i

Final intensity after the 2nd polarizer=0.742228 + 0. i

Check numbers after each CNC

4) Uniaxial

Storage

Loop

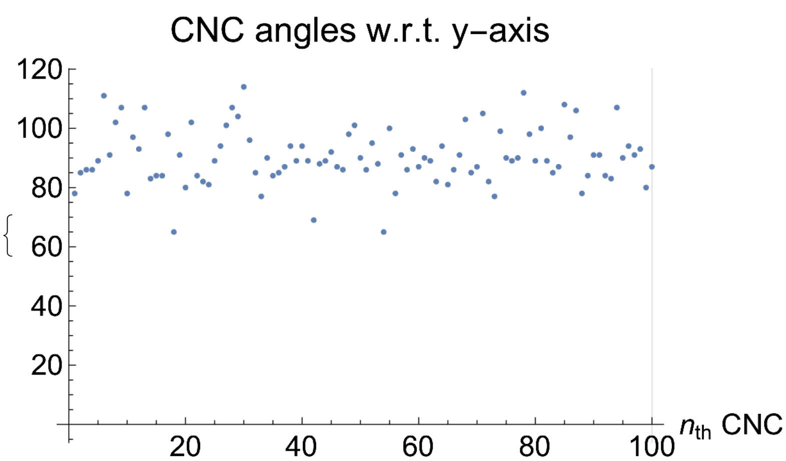
In[461]:=
```
For[p = 1, p < n + 1, p++,

 ϕbtn[[p]] = ψy[[p]] - ϕun180Norm[[p]]; (* convert axis: From y-axis to CNC normal *)
 ψres[[p]] = ψ[ϕbtn[[p]]]; (* result of rotation in CNC normal axis*)
 ψy[[p + 1]] = ψres[[p]] + ϕun180Norm[[p]];
 (* back to y-axis system*) (*F180 is for *)

 dψ[[p]] = ψres[[p]] - ϕbtn[[p]]; (* (ψafter - ψbefore) in CNC axis *)

 ttot[[p]] = tt[ϕbtn[[p]]];
 Etot[[p + 1]] = Etot[[p]] * ttot[[p]]; (* energy loss as passing through *)

 dψ2[[p]] = ψy[[p + 1]] - ψy[[p]];
 (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)

]
```

$$PI = \left(Etot[[n+1]] * Sin\left[(\psi y[[n+1]] - \phi 1) * \frac{\pi}{180}\right]\right)^2; \text{(* } I=(|E_{total}|*sin\psi)^2 \text{ - Equation 5 *)}$$

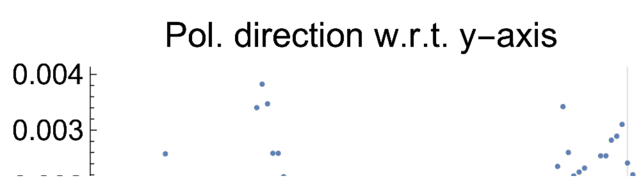
Check plots and final intensity

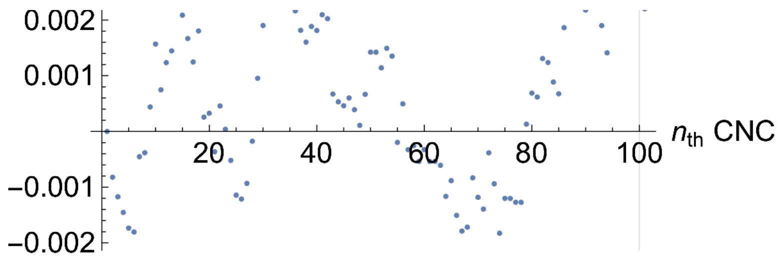
In[463]:=
```
{ListPlot[ϕun180, ImageSize → 400, PlotRange → All, AxesLabel → {"nth CNC", None},
   LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
   GridLines → {{100}, {180}}, AxesOrigin → {0, 0},
   PlotLabel → "CNC angles w.r.t. y-axis"],
 ListPlot[ψy, ImageSize → 400, PlotRange → All, LabelStyle →
    {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
   PlotLabel → "Pol. direction w.r.t. y-axis", GridLines → {{100}, None}],
 ListPlot[Etot, ImageSize → 400, PlotRange → All,
   LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
   AxesLabel → {"nth CNC", None}, PlotLabel → "|Etotal|", GridLines → {{100}, None}],
 ListPlot[(dψ), ImageSize → 400, PlotRange → All, LabelStyle →
    {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
   PlotLabel → "Pol. rotation", GridLines → {{100}, None}]}
{ListPlot[ϕbtn, ImageSize → 400, LabelStyle →
    {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
   PlotLabel → "Pol. direction w.r.t. CNC normal (before)",
   GridLines → {{100}, None}],
 ListPlot[ψres, ImageSize → 400, LabelStyle →
    {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"nth CNC", None},
   PlotLabel → "Pol. direction w.r.t. CNC normal (after)",
   GridLines → {{100}, None}]}

Print["Etotal after 100th CNC=", Etot[[n + 1]]]
Print["Final intensity after the 2nd polarizer=", PI]
```

Out[463]= {

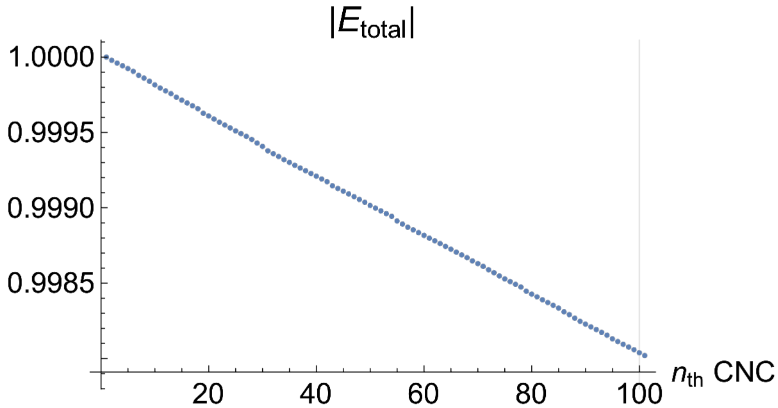


,

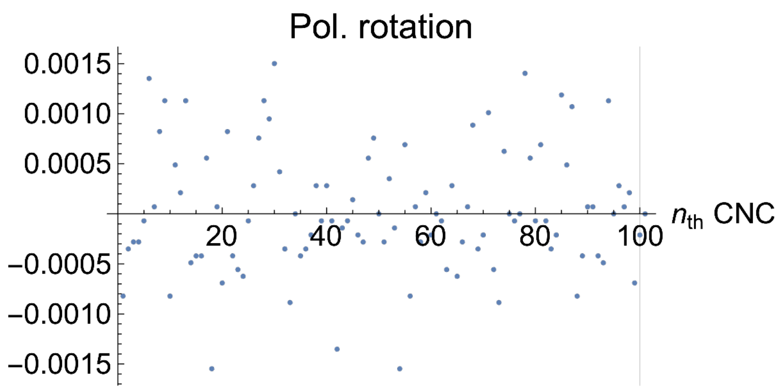



0.002
0.001
−0.001
−0.002
20
40
60
80
100
n_{th} CNC

,

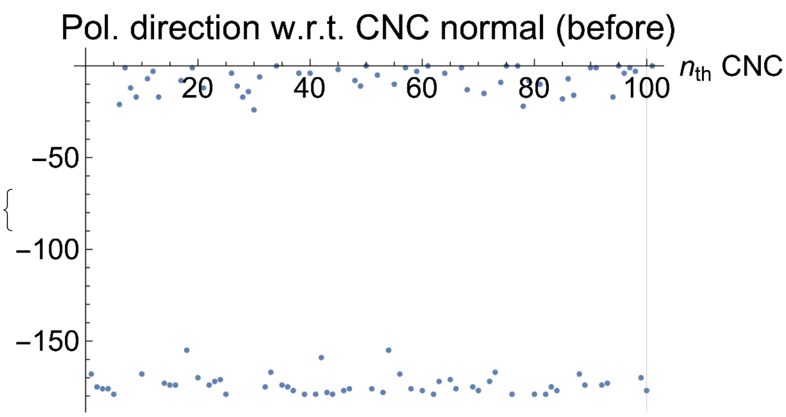

$|E_{\text{total}}|$
1.0000
0.9995
0.9990
0.9985
20
40
60
80
100
n_{th} CNC

,

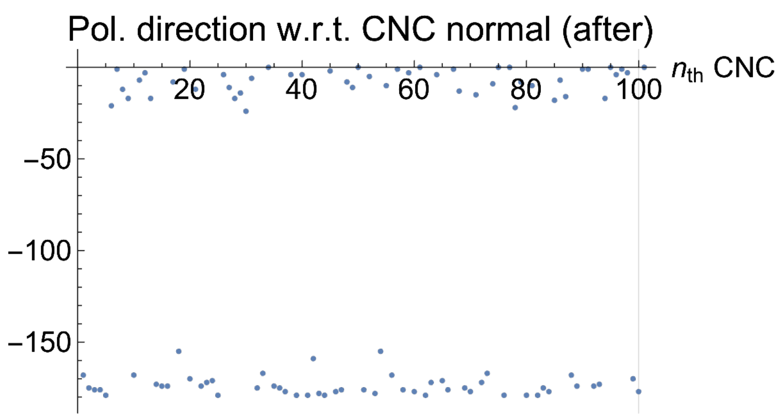

Pol. rotation
0.0015
0.0010
0.0005
−0.0005
−0.0010
−0.0015
20
40
60
80
100
n_{th} CNC

}

Out[464]= {



,



}

E_{total} after 100th CNC=0.99802 + 0. i

Final intensity after the 2nd polarizer=0.747 + 0. i

Check numbers after each CNC

5) Bimodal

Storage

Loop

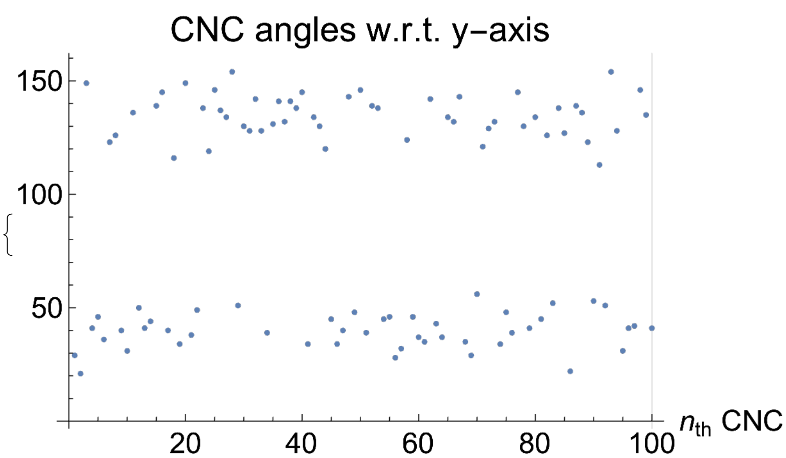
In[488]:=
```
For[p = 1, p < n + 1, p++,

 ϕbtn⟦p⟧ = ψy⟦p⟧ - ϕbi180Norm⟦p⟧; (* convert axis: From y-axis to CNC normal *)
 ψres⟦p⟧ = ψ[ϕbtn⟦p⟧]; (* result of rotation in CNC normal axis*)
 ψy⟦p + 1⟧ = ψres⟦p⟧ + ϕbi180Norm⟦p⟧;
 (* back to y-axis system*) (*F180 is for *)

 dψ⟦p⟧ = ψres⟦p⟧ - ϕbtn⟦p⟧; (* (ψafter - ψbefore) in CNC axis *)

 ttot⟦p⟧ = tt[ϕbtn⟦p⟧];
 Etot⟦p + 1⟧ = Etot⟦p⟧ * ttot⟦p⟧; (* energy loss as passing through *)

 dψ2⟦p⟧ = ψy⟦p + 1⟧ - ψy⟦p⟧;
 (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)
]
```

$$PI = \left(Etot[[n+1]] * Sin\left[(\psi y[[n+1]] - \phi 1) * \frac{\pi}{180}\right]\right)^2;$$ (* I=(|E_{total}|*sinψ)^2 - Equation 5 *)

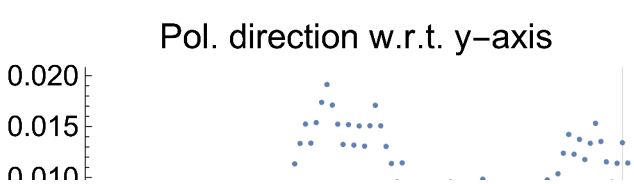
Check plots and final intensity

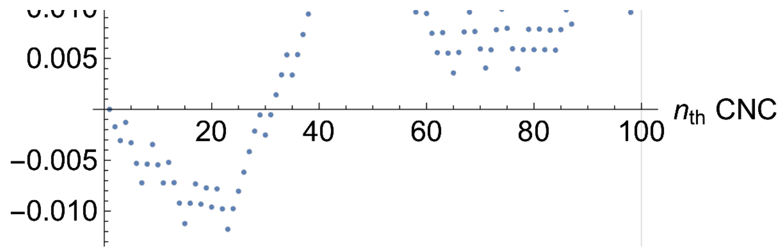
In[490]:=
```
{ListPlot[ϕbi180, ImageSize → 400, PlotRange → All, AxesLabel → {"n_th CNC", None},
   LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
   GridLines → {{100}, {180}}, AxesOrigin → {0, 0},
   PlotLabel → "CNC angles w.r.t. y-axis"],
  ListPlot[ψy, ImageSize → 400, PlotRange → All, LabelStyle →
    {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"n_th CNC", None},
   PlotLabel → "Pol. direction w.r.t. y-axis", GridLines → {{100}, None}],
  ListPlot[Etot, ImageSize → 400, PlotRange → All,
   LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
   AxesLabel → {"n_th CNC", None}, PlotLabel → "|E_total|", GridLines → {{100}, None}],
  ListPlot[(dψ), ImageSize → 400, PlotRange → All, LabelStyle →
    {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"n_th CNC", None},
   PlotLabel → "Pol. rotation", GridLines → {{100}, None}]}
{ListPlot[ϕbtn, ImageSize → 400, LabelStyle →
    {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"n_th CNC", None},
   PlotLabel → "Pol. direction w.r.t. CNC normal (before)",
   GridLines → {{100}, None}],
  ListPlot[ψres, ImageSize → 400, LabelStyle →
    {FontFamily → "Arial", FontSize → 15, Black}, AxesLabel → {"n_th CNC", None},
   PlotLabel → "Pol. direction w.r.t. CNC normal (after)",
   GridLines → {{100}, None}]}

Print["E_total after 100th CNC=", Etot[[n + 1]]]
Print["Final intensity after the 2nd polarizer=", PI]
```

Out[490]= {

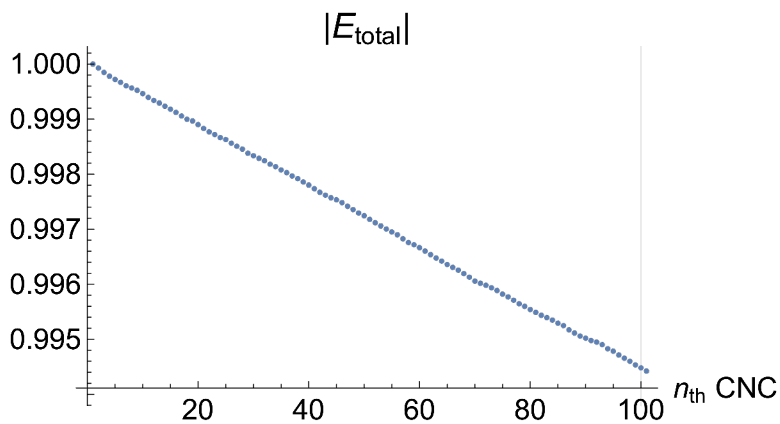


,

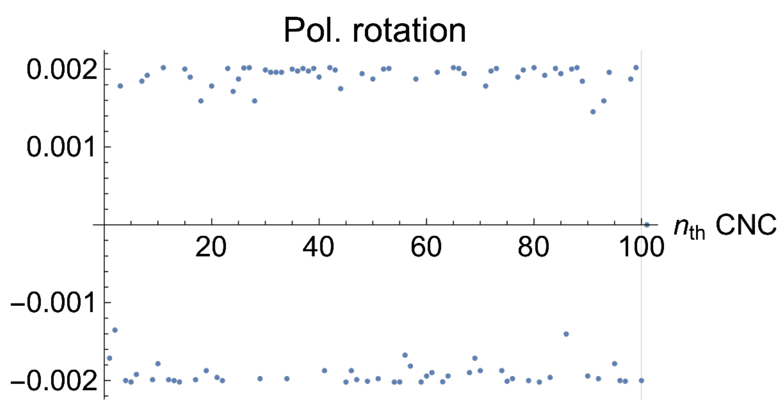



0.005
−0.005
−0.010
20
40
60
80
100
n_{th} CNC

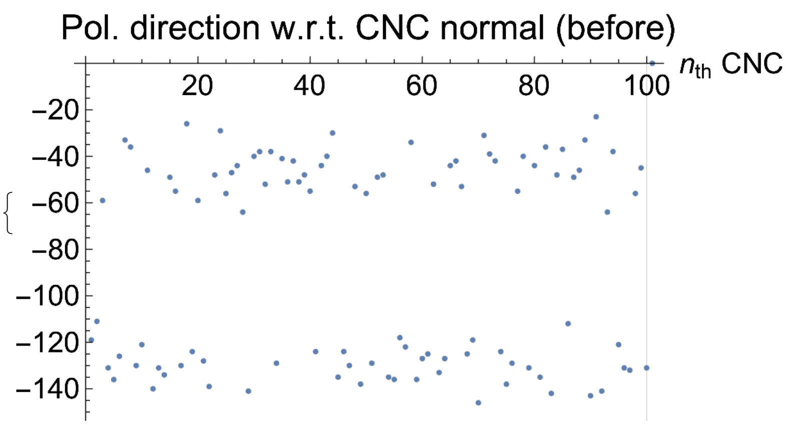
,

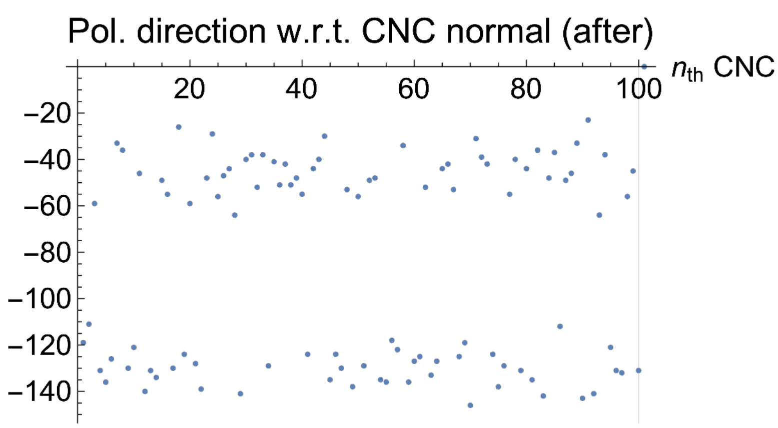

$|E_{total}|$
1.000
0.999
0.998
0.997
0.996
0.995
20
40
60
80
100
n_{th} CNC

,


Pol. rotation
0.002
0.001
−0.001
−0.002
20
40
60
80
100
n_{th} CNC

}

Out[491]= {

,


}

E_{total} after 100th CNC=0.994422 + 0. ⅈ

Final intensity after the 2nd polarizer=0.741486 + 0. ⅈ

Check numbers after each CNC

G. Final intensity as a function of the incident polarization angle

Note: the same calculation with various incident polarization angle (no explanation included). Data exportation included.

In[499]:=
```
(*Crossed-polylamellate*)
PIcl[x_] := Module[{},
  ϕi = x + ϵ; (* Polarization direction w.r.t. y-axis *)
  ψy = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)
  ψy[[1]] = N[ϕi];

  ϕbtn = Table[0, {i, 1, n + 1}]; (*angle between Pol. direction & CNC normal*)

  ψres = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)

  Etot = Table[0, {i, 1, n + 1}];
  Etot[[1]] = 1;
  ttot = Table[0, {i, 1, n + 1}];

  dψ = Table[0, {i, 1, n + 1}];
  dψ2 = Table[0, {i, 1, n + 1}];

  (**)
  For[p = 1, p < n + 1, p++,

   ϕbtn[[p]] = ψy[[p]] - ϕcl180Norm[[p]]; (* convert axis: From y-axis to CNC normal *)
   ψres[[p]] = ψ[ϕbtn[[p]]]; (* result of rotation in CNC normal axis*)
   ψy[[p + 1]] = ψres[[p]] + ϕcl180Norm[[p]];
   (* back to y-axis system*) (*F180 is for *)

   dψ[[p]] = ψres[[p]] - ϕbtn[[p]]; (* (ψafter - ψbefore) in CNC axis *)

   ttot[[p]] = tt[ϕbtn[[p]]];
   Etot[[p + 1]] = Etot[[p]] * ttot[[p]]; (* energy loss as passing through *)

   dψ2[[p]] = ψy[[p + 1]] - ψy[[p]];
   (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)

  ];
  PI = (Etot[[n + 1]] * Sin[(ψy[[n + 1]] - ϕi) * π/180])^2
 ]
clAng = Table[{i, PIcl[i]}, {i, 0, 180, 5}];
```

In[501]:=
```
(*Random*)
PIra[x_] := Module[{},
  ϕi = x + ϵ; (* Polarization direction w.r.t. y-axis *)
  ψy = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)
  ψy[[1]] = N[ϕi];

  ϕbtn = Table[0, {i, 1, n + 1}]; (*angle between Pol. direction & CNC normal*)

  ψres = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)

  Etot = Table[0, {i, 1, n + 1}];
  Etot[[1]] = 1;
  ttot = Table[0, {i, 1, n + 1}];

  dψ = Table[0, {i, 1, n + 1}];
  dψ2 = Table[0, {i, 1, n + 1}];

  (**)
  For[p = 1, p < n + 1, p++,

   ϕbtn[[p]] = ψy[[p]] - ϕra180Norm[[p]]; (* convert axis: From y-axis to CNC normal *)
   ψres[[p]] = ψ[ϕbtn[[p]]]; (* result of rotation in CNC normal axis*)
   ψy[[p + 1]] = ψres[[p]] + ϕra180Norm[[p]];
   (* back to y-axis system*) (*F180 is for *)

   dψ[[p]] = ψres[[p]] - ϕbtn[[p]]; (* (ψafter - ψbefore) in CNC axis *)

   ttot[[p]] = tt[ϕbtn[[p]]];
   Etot[[p + 1]] = Etot[[p]] * ttot[[p]]; (* energy loss as passing through *)

   dψ2[[p]] = ψy[[p + 1]] - ψy[[p]];
   (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)

  ];
  PI = (Etot[[n + 1]] * Sin[(ψy[[n + 1]] - ϕi) * π/180])^2
 ]
raAng = Table[{i, PIra[i]}, {i, 0, 180, 5}];
```

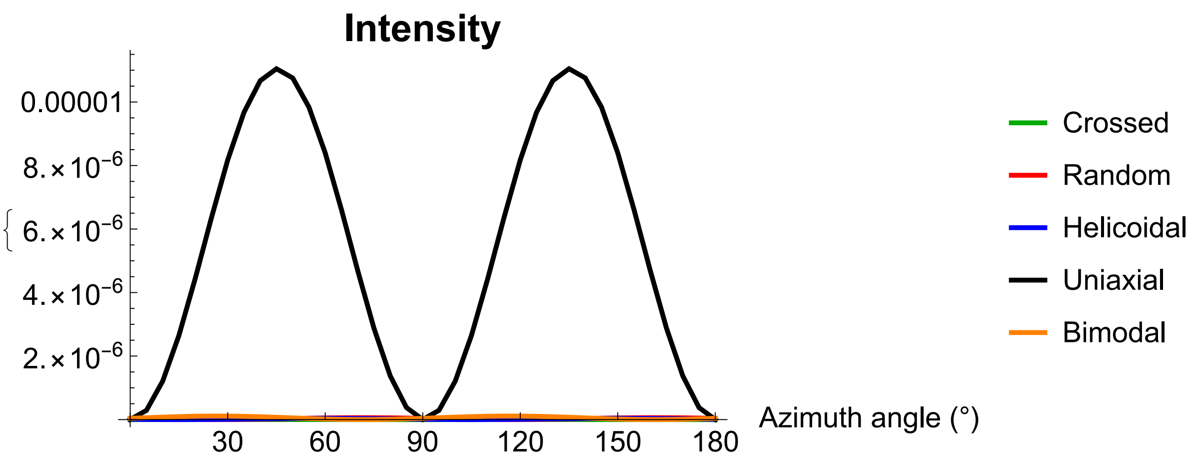
In[503]:=
```
(*Helicoidal*)
PIhe[x_] := Module[{},
  ϕi = x + ϵ; (* Polarization direction w.r.t. y-axis *)
  ψy = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)
  ψy[[1]] = N[ϕi];

  ϕbtn = Table[0, {i, 1, n + 1}]; (*angle between Pol. direction & CNC normal*)

  ψres = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)

  Etot = Table[0, {i, 1, n + 1}];
  Etot[[1]] = 1;
  ttot = Table[0, {i, 1, n + 1}];

  dψ = Table[0, {i, 1, n + 1}];
  dψ2 = Table[0, {i, 1, n + 1}];

  (**)
  For[p = 1, p < n + 1, p++,

   ϕbtn[[p]] = ψy[[p]] - ϕhe180Norm[[p]]; (* convert axis: From y-axis to CNC normal *)
   ψres[[p]] = ψ[ϕbtn[[p]]]; (* result of rotation in CNC normal axis*)
   ψy[[p + 1]] = ψres[[p]] + ϕhe180Norm[[p]];
   (* back to y-axis system*) (*F180 is for *)

   dψ[[p]] = ψres[[p]] - ϕbtn[[p]]; (* (ψafter - ψbefore) in CNC axis *)

   ttot[[p]] = tt[ϕbtn[[p]]];
   Etot[[p + 1]] = Etot[[p]] * ttot[[p]]; (* energy loss as passing through *)

   dψ2[[p]] = ψy[[p + 1]] - ψy[[p]];
   (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)

  ];
  PI = (Etot[[n + 1]] * Sin[(ψy[[n + 1]] - ϕi) * π/180])^2
 ]
heAng = Table[{i, PIhe[i]}, {i, 0, 180, 5}];
```

In[505]:=
```
(*Uniaxial*)
PIun[x_] := Module[{},
  ϕi = x + ϵ; (* Polarization direction w.r.t. y-axis *)
  ψy = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)
  ψy⟦1⟧ = N[ϕi];

  ϕbtn = Table[0, {i, 1, n + 1}]; (*angle between Pol. direction & CNC normal*)

  ψres = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)

  Etot = Table[0, {i, 1, n + 1}];
  Etot⟦1⟧ = 1;
  ttot = Table[0, {i, 1, n + 1}];

  dψ = Table[0, {i, 1, n + 1}];
  dψ2 = Table[0, {i, 1, n + 1}];

  (**)
  For[p = 1, p < n + 1, p++,

   ϕbtn⟦p⟧ = ψy⟦p⟧ - ϕun180Norm⟦p⟧; (* convert axis: From y-axis to CNC normal *)
   ψres⟦p⟧ = ψ[ϕbtn⟦p⟧]; (* result of rotation in CNC normal axis*)
   ψy⟦p + 1⟧ = ψres⟦p⟧ + ϕun180Norm⟦p⟧;
   (* back to y-axis system*) (*F180 is for *)

   dψ⟦p⟧ = ψres⟦p⟧ - ϕbtn⟦p⟧; (* (ψafter - ψbefore) in CNC axis *)

   ttot⟦p⟧ = tt[ϕbtn⟦p⟧];
   Etot⟦p + 1⟧ = Etot⟦p⟧ * ttot⟦p⟧; (* energy loss as passing through *)

   dψ2⟦p⟧ = ψy⟦p + 1⟧ - ψy⟦p⟧;
   (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)

  ];
  PI = (Etot⟦n + 1⟧ * Sin[(ψy⟦n + 1⟧ - ϕi) * π/180])^2
 ]
unAng = Table[{i, PIun[i]}, {i, 0, 180, 5}];
```

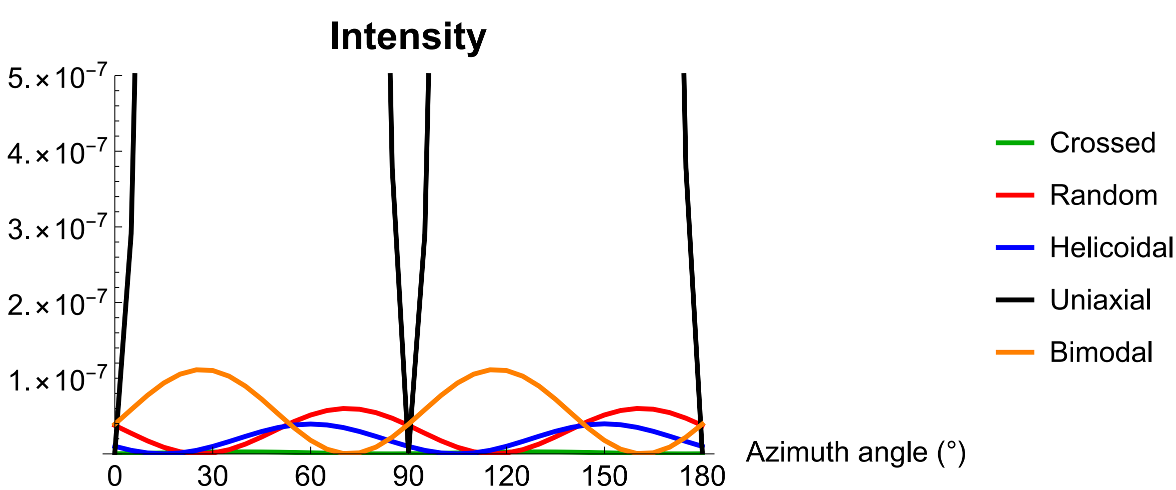
In[507]:=
```
(*Bimodal*)
PIbi[x_] := Module[{},
  ϕi = x + ϵ; (* Polarization direction w.r.t. y-axis *)
  ψy = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)
  ψy[[1]] = N[ϕi];

  ϕbtn = Table[0, {i, 1, n + 1}]; (*angle between Pol. direction & CNC normal*)

  ψres = Table[0, {i, 1, n + 1}]; (* Azimuth angle after passing a CNC*)

  Etot = Table[0, {i, 1, n + 1}];
  Etot[[1]] = 1;
  ttot = Table[0, {i, 1, n + 1}];

  dψ = Table[0, {i, 1, n + 1}];
  dψ2 = Table[0, {i, 1, n + 1}];

  (**)
  For[p = 1, p < n + 1, p++,

   ϕbtn[[p]] = ψy[[p]] - ϕbi180Norm[[p]]; (* convert axis: From y-axis to CNC normal *)
   ψres[[p]] = ψ[ϕbtn[[p]]]; (* result of rotation in CNC normal axis*)
   ψy[[p + 1]] = ψres[[p]] + ϕbi180Norm[[p]];
   (* back to y-axis system*) (*F180 is for *)

   dψ[[p]] = ψres[[p]] - ϕbtn[[p]]; (* (ψafter - ψbefore) in CNC axis *)

   ttot[[p]] = tt[ϕbtn[[p]]];
   Etot[[p + 1]] = Etot[[p]] * ttot[[p]]; (* energy loss as passing through *)

   dψ2[[p]] = ψy[[p + 1]] - ψy[[p]];
   (* (ψafter - ψbefore) in CNC axis. How much rotation @ each CNC*)

  ];
  PI = (Etot[[n + 1]] * Sin[(ψy[[n + 1]] - ϕi) * π/180])^2
 ]
biAng = Table[{i, PIbi[i]}, {i, 0, 180, 5}];
```

In[509]:=
```
(* Check plots *)
{ListPlot[{clAng, raAng, heAng, unAng, biAng},
  PlotStyle → {{Darker[Green], Thickness[0.008]}, {Red, Thickness[0.008]}, {Blue,
     Thickness[0.008]}, {Black, Thickness[0.008]}, {Orange, Thickness[0.008]}},
  Ticks → {Table[30 i, {i, 0, 6}], Automatic}, PlotRange → All,
  ImageSize → 500, AxesLabel → {"  Azimuth angle (°)", None},
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  PlotLabel → Style["Intensity", 20, Black, Bold], Joined → True,
  PlotLegends → {"Crossed", "Random", "Helicoidal", "Uniaxial", "Bimodal"}],
 ListPlot[{clAng, raAng, heAng, unAng, biAng},
  PlotStyle → {{Darker[Green], Thickness[0.008]}, {Red, Thickness[0.008]}, {Blue,
     Thickness[0.008]}, {Black, Thickness[0.008]}, {Orange, Thickness[0.008]}},
  Ticks → {Table[30 i, {i, 0, 6}], Automatic}, PlotRange → {All, {0, 5 * 10^-7}},
  ImageSize → 500, AxesLabel → {"  Azimuth angle (°)", None},
  LabelStyle → {FontFamily → "Arial", FontSize → 15, Black},
  PlotLabel → Style["Intensity", 20, Black, Bold], Joined → True,
  PlotLegends → {"Crossed", "Random", "Helicoidal", "Uniaxial", "Bimodal"}]}
```

Out[509]= {



,



}

In[510]:=
```
(* Combine generated data *)
clAng2 = Transpose[clAng];
raAng2 = Transpose[raAng];
heAng2 = Transpose[heAng];
unAng2 = Transpose[unAng];
biAng2 = Transpose[biAng];
b1 = {clAng2⟦1⟧, clAng2⟦2⟧, raAng2⟦2⟧, heAng2⟦2⟧, unAng2⟦2⟧, biAng2⟦2⟧};
b1 = Abs[b1];
b1 = b1 // Transpose;
b1 // MatrixForm;

(* Data exportation *)
name = CurrentValue[EvaluationNotebook[], {"NotebookFileName"}];
name = StringDrop[name, -3];
time = TextString[TimeObject[]];
text1 = StringTake[time, 2];
text2 = StringTake[time, {4, 5}];
text3 = StringTake[time, -2];
time2 = ToExpression["text1"] <> ToString["."] <>
  ToExpression["text2"] <> ToString["."] <> ToExpression["text3"]
Export[NotebookDirectory[] <> ToExpression["name"] <>
   ToExpression["time2"] <> ToString["_"] <> "CRHUB.xlsx", b1];
```

Out[525]= 22.27.42