Simulation of pickoff beams using Zemax

Richard DAY

European Gravitational Observatory

Warning !

In this talk I am presenting the tool.

Simulation results have not been validated or cross-checked

Motivation

- Need for 3D optical CAD software for Virgo
- Need for understanding where all pickoffs go in the interferometer

Note: none of the software discussed here is a replacement for DarkF, Oscar etc…

Optocad ideal tool for CAD design of interferometer

But only 2D → little chance of becoming 3D in near future

Zemax is a commercial tool with two modes of operation

Non-sequential mode:

Place all optics in 3D. Watch what happens when you send in rays

Limited functionality for gaussian beam propagation and aberration analysis

Sequential mode:

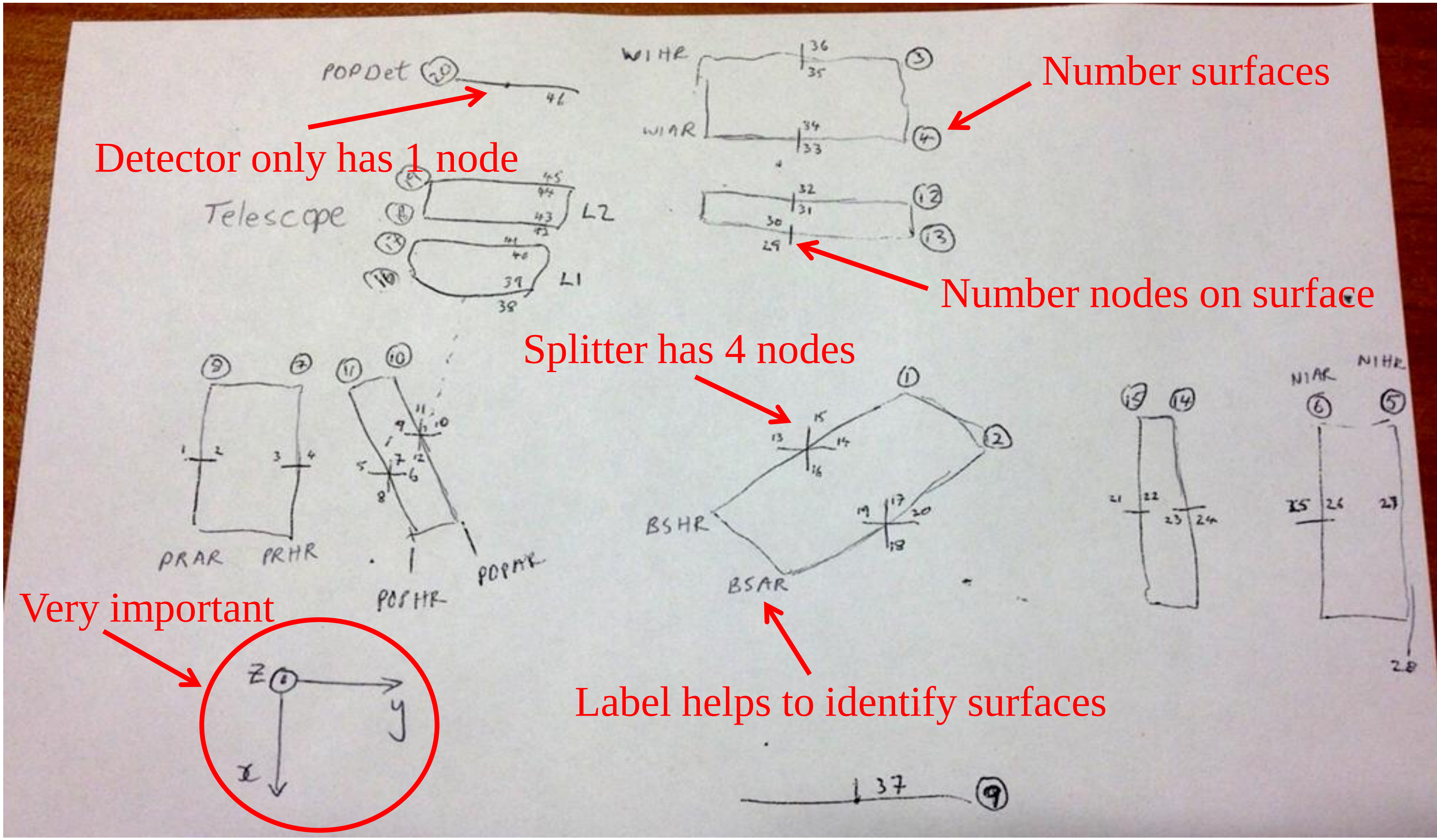
Place surfaces in order that beam will encounter them

Very very difficult to design interferometer with pickoffs

Solution

Use Matlab to interface with Zemax in sequential mode to make configurations and extract needed data

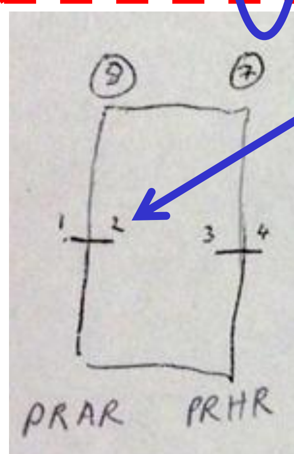
First design what you want to simulate



Next we connect nodes and surfaces all together using look-up tables

```
ts = [ 1  2  3 4 5   6  7  8   9  10 11  12 13  14 15  16  17 18  19 20 21  22  23 24 25  26  27 28 29  30  31 32 33  34  35 36 37 38  39  40  41 42  43  44 45 46]; % node index (not used)
re = [ 1  2  3 4 8   7  6  5   12 11 10  9  15  16 13  14  19 20  17 18 21  22  23 24 25  26  27 28 29  30  31 32 33  34  35 36 0  38  39  40  41 42  43  44 45 0 ]; % node reflected to
tr = [ 2  1  4 3 6   5  8  7   10 9  12  11 14  13 16  15  18 17  20 19 22  21  24 23 26  25  28 27 30  29  32 31 34  33  36 35 0  39  38  41  40 43  42  45 44 0 ]; % node transmitted to
pr = [ 0  3  2 5 4   9  12 0   6  13 38  7  10  19 29  17  16 37  14 21 20  23  22 25 24  27  26 0  15  31  30 33 32  35  34 0  18 11  40  39  42 41  44  43 46 45]; % node propagated to
s  = [ 8 -8 -7 7 11 -11 -11 11 -10 10 10 -10 1 -1  1 -1  -2  2  -2  2  15 -15 -14 14 6  -6  -5  5  13 -13 -12 12 4  -4  -3  3   9  16 -16 -17  17 18 -18 -19 19 20]; % surface correspondance
```

```
ts = [ 1  2  3 4 5   6    % node index (not used)
re = [ 1  2  3 4 8   7    % node reflected to
tr = [ 2  1  4 3 6   5    % node transmitted to
pr = [ 0  3  2 5 4   9    % node propagated to
s  = [ 8 -8 -7 7 11 -11   % surface correspondance
```



Example:

Node 2 reflects to node 2

Node 2 transmits to node 1

Node 2 propagates to node 3

Node 2 on surface 8 (-ve because in glass)

Surface properties

Next we define the properties of the surface by building a struct for each surface

```
Sdata(1)  = struct('com', 'BS_HR',  'pos', BSHR , 'dir',  BSHRdir  , 'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,     'ref', 0.5,   'Type', 'S', 'diam', 0.550);
Sdata(2)  = struct('com', 'BS_AR',  'pos', BSAR , 'dir',  BSARdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,     'ref', 100e-6,'Type', 'A', 'diam', 0.550);
Sdata(3)  = struct('com', 'WI_HR',  'pos', WIHR , 'dir',  WIHRdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc', -1420,   'ref', 0.986,  'Type', 'R', 'diam', 0.350);
Sdata(4)  = struct('com', 'WI_AR',  'pos', WIAR , 'dir',  WIARdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  1420.2, 'ref', 100e-6, 'Type', 'A', 'diam', 0.350);
Sdata(5)  = struct('com', 'NI_HR',  'pos', NIHR , 'dir',  NIHRdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc', -1420,   'ref', 0.986,  'Type', 'R', 'diam', 0.350);
Sdata(6)  = struct('com', 'NI_AR',  'pos', NIAR , 'dir',  NIARdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  1420.2, 'ref', 100e-6, 'Type', 'A', 'diam', 0.350);
Sdata(7)  = struct('com', 'PR_HR',  'pos', PRHR , 'dir',  PRHRdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc', -1431.4, 'ref', 0.95,   'Type', 'S', 'diam', 0.350);
Sdata(8)  = struct('com', 'PR_AR',  'pos', PRAR , 'dir',  PRARdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  3.620,  'ref', 100e-6, 'Type', 'A', 'diam', 0.350);
Sdata(9)  = struct('com', 'D_DF',   'pos', DDF ,  'dir',  DDFdir ,   'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,    'ref', 0,      'Type', 'B', 'diam', 0.1);
Sdata(10) = struct('com', 'POP_AR', 'pos', POPAR ,'dir',  POPARdir , 'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,    'ref', 100e-6, 'Type', 'A', 'diam', 0.350);
Sdata(11) = struct('com', 'POP_HR', 'pos', POPHR ,'dir',  POPHRdir , 'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,    'ref', 300e-6, 'Type', 'S', 'diam', 0.350);
Sdata(12) = struct('com', 'CPW_W',  'pos', CPWW , 'dir',  CPWWdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,    'ref', 100e-6, 'Type', 'A', 'diam', 0.350);
Sdata(13) = struct('com', 'CPW_BS', 'pos', CPWBS ,'dir',  CPWBSdir , 'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,    'ref', 100e-6, 'Type', 'A', 'diam', 0.350);
Sdata(14) = struct('com', 'CPN_N',  'pos', CPNN , 'dir',  CPNNdir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,    'ref', 100e-6, 'Type', 'A', 'diam', 0.350);
Sdata(15) = struct('com', 'CPN_BS', 'pos', CPNBS ,'dir',  CPNBSdir , 'ori',  Allori  ,'Mp', '', 'Mm', GlassType1, 'roc',  Inf,    'ref', 100e-6, 'Type', 'A', 'diam', 0.350);
Sdata(16) = struct('com', 'L1_POP', 'pos', L1POP ,'dir',  L1L2dir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType2, 'roc',  2.188,  'ref', 0,      'Type', 'T', 'diam', 0.23);
Sdata(17) = struct('com', 'L1_Det', 'pos', L1Det ,'dir',  L1L2dir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType2, 'roc',  7.3345, 'ref', 0,      'Type', 'T', 'diam', 0.23);
Sdata(18) = struct('com', 'L2_POP', 'pos', L2POP ,'dir',  L1L2dir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType2, 'roc', -2.979,  'ref', 0,      'Type', 'T', 'diam', 0.23);
Sdata(19) = struct('com', 'L2_Det', 'pos', L2Det ,'dir',  L1L2dir ,  'ori',  Allori  ,'Mp', '', 'Mm', GlassType2, 'roc',  4.500,  'ref', 0,      'Type', 'T', 'diam', 0.23);
Sdata(20) = struct('com', 'POP_Det','pos', POPDet,'dir',  POPDetdir ,'ori',  Allori  ,'Mp', '', 'Mm', '',         'roc',  Inf,    'ref', 0,      'Type', 'D', 'diam', 0.1);
```

Comment appearing in Zemax

3D position of centre

Direction vector normal to surface

2nd vector that defines plane with direction vector giving orientation

Glass used in Zemax

Super important !
Defines how surfaces behave

```
Stype = ['H' 'A' 'S' 'D' 'T' 'R' 'L' 'B']; % don't change
% H=high reflectivity surface. Main beam reflected. Pickoff beam transmitted
% A=antireflection coating.    Pickoff beam reflected. Main beam transmitted
% S=beamsplitter.              Main beam reflected. Main beam transmitted
% D=detector
% T=perfect transmitter.       No beam reflected. Main beam transmitted
% R=perfect reflector.         Main beam reflected. No beam transmitted
% L=Laser.                     Laser source. No reflection of transmission
% B=Beam dump.                 Beam dump. No reflection of transmission
```

Find beam paths

```
L = 4; % which node has laser
Beam.lambda = 1064e-9;
Beam.pos = PRHR;
Beam.dir = PRHRdir;
Beam.ori = Allori;
Beam.radius = 49.06e-3;
Beam.roc = -1431.4; % positive RoC is diverging from node
```

Finally we define the beam properties

Choose beam leaving PR HR

```
P = 1; % pick off number: transmission through H, reflection off A.
       % e.g. P=0 no pickoffs considered
       %      P-1 first pickoff considered
       %      P=2 pickoff AND pickoff of pickoff considered
```

Now we decide how many pickoffs we want to consider

Now algorithm chooses all possible beam paths in optical configuration

If P = 0 is chosen there are just 2 beam paths: reflection of WI and reflection off NI

```
Paths =

  Columns 1 through 12

    'PR_HR'    'POP_HR'    'POP_AR'    'BS_HR'    'CPW_BS'    'CPW_W'    'WI_AR'    'WI_HR'    'WI_AR'    'CPW_W'    'CPW_BS'    'BS_HR'
    'PR_HR'    'POP_HR'    'POP_AR'    'BS_HR'    'BS_AR'     'CPN_BS'   'CPN_N'    'NI_AR'    'NI_HR'    'NI_AR'    'CPN_N'     'CPN_BS'

  Columns 13 through 22

    'POP_AR'    'POP_HR'    'POP_AR'    'L1_POP'    'L1_Det'    'L2_POP'    'L2_Det'    'POP_Det'            []            []
    'BS_AR'     'BS_HR'     'POP_AR'    'POP_HR'    'POP_AR'    'L1_POP'    'L1_Det'    'L2_POP'    'L2_Det'    'POP_Det'
```

If P = 1 is chosen there are 24 beam paths:

Paths =

Columns 1 through 13

'PR_HR'	'POP_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'
'PR_HR'	'POP_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'L1_POP'	'L1_Det'	'L2_POP'	'L2_Det'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'CPW_BS'	'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'L1_POP'	'L1_Det'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'CPW_W'	'CPW_BS'	'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'	'CPW_W'	'WI_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'	'BS_HR'	'POP_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'	'BS_HR'	'POP_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'	'BS_HR'	'POP_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'	'BS_HR'	'BS_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'BS_HR'	'CPW_BS'	'CPW_W'	'WI_AR'	'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'BS_AR'	'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'L1_POP'	'L1_Det'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'CPN_BS'	'BS_AR'	'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'CPN_N'	'CPN_BS'	'BS_AR'	'BS_HR'	'POP_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'	'NI_AR'	'NI_HR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'	'CPN_BS'	'CPN_N'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'	'CPN_BS'	'BS_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'	'CPN_BS'	'BS_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'	'CPN_BS'	'BS_AR'
'PR_HR'	'POP_HR'	'POP_AR'	'BS_HR'	'BS_AR'	'CPN_BS'	'CPN_N'	'NI_AR'	'NI_HR'	'NI_AR'	'CPN_N'	'CPN_BS'	'BS_AR'

Columns 14 through 25

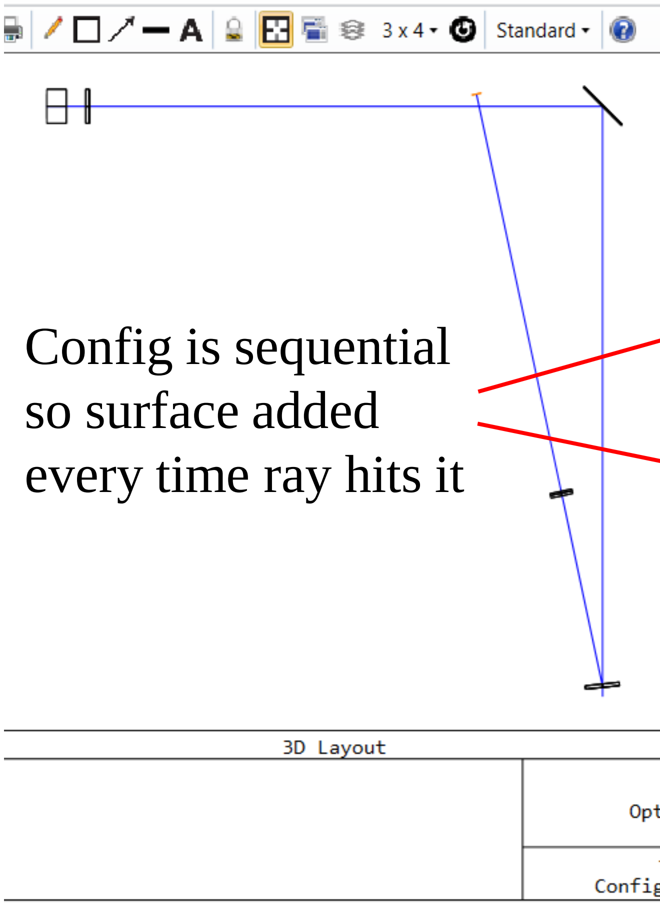
'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'L1_POP'	'L1_Det'	'L2_POP'	'L2_Det'	'POP_Det'	[]	[]	[]
'CPN_BS'	'BS_AR'	'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'L1_POP'	'L1_Det'	'L2_POP'	'L2_Det'	'POP_Det'	[]
'POP_Det'	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]	[]
'L2_POP'	'L2_Det'	'POP_Det'	[]	[]	[]	[]	[]	[]	[]	[]	[]
'L1_POP'	'L1_Det'	'L2_POP'	'L2_Det'	'POP_Det'	[]	[]	[]	[]	[]	[]	[]
'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'L1_POP'	'L1_Det'	'L2_POP'	'L2_Det'	'POP_Det'	[]	[]	[]
'CPW_W'	'CPW_BS'	'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'L1_POP'	'L1_Det'	'L2_POP'	'L2_Det'	'POP_Det'	[]
'WI_HR'	'WI_AR'	'CPW_W'	'CPW_BS'	'BS_HR'	'POP_AR'	'POP_HR'	'POP_AR'	'L1_POP'	'L1_Det'	'L2_POP'	'L2_Det'

If P = 2 is chosen there are 140 beam paths:

Build configurations in Zemax

Matlab opens a session in Zemax and builds sequential config for each path

Matlab saves configs so user can interact with design using Zemax interface



Config is sequential so surface added every time ray hits it

Lens Data

Surface 24 Properties — Configuration 1/1

	Surf:Type		Comment	Radius	Thickness	Material	Coating	Semi-Diameter
5		Standard	Beam Start	Infinity	95.116			2.885E-017
6	(aper and tilts)	Standard	POP_HR	Infinity	35.000	SUPR...		175.000 U
7	(aper and tilts)	Standard	POP_AR	Infinity	5905.093			175.000 U
8	(aper and tilts)	Standard	BS_HR	Infinity	-5248.102	MIRR...		275.000 U
9	(aper and tilts)	Standard	CPW_BS	Infinity	-35.000	SUPR...		175.000 U
10	(aper and tilts)	Standard	CPW_W	Infinity	-200.000			175.000 U
11	(aper and tilts)	Standard	WI_AR	-1.420E+...	-200.000	SUPR...		175.000 U
12	(aper)	Standard	WI_HR	-1.420E+...	200.000	MIRR...		175.000 U
13	(aper and tilts)	Standard	WI_AR	-1.420E+...	200.000			175.000 U
14	(aper and tilts)	Standard	CPW_W	Infinity	35.000	SUPR...		175.000 U
15	(aper and tilts)	Standard	CPW_BS	Infinity	5248.102			175.000 U
16	(aper and tilts)	Standard	BS_HR	Infinity	-5905.093	MIRR...		275.000 U
17	(aper and tilts)	Standard	POP_AR	Infinity	-35.000	SUPR...		175.000 U
18	(aper)	Standard	POP_HR	Infinity	35.000	MIRR...		175.000 U
19	(aper and tilts)	Standard	POP_AR	Infinity	1965.195			175.000 U
20	(aper and tilts)	Standard	L1_POP	2188.000	30.000	BK7		115.000 U
21	(aper)	Standard	L1_Det	-7334.500	1.319			115.000 U
22	(aper)	Standard	L2_POP	-2979.000	19.000	BK7		115.000 U
23	(aper)	Standard	L2_Det	-4500.000	4150.000			115.000 U
24		Standard	POP_Det	Infinity	0.000			50.000 U
25	IMAGE	Standard		Infinity	-			7.875E-005

Analysis from Matlab

Real power of this approach is capability to automate analysis with Matlab

Easy to access ray data from Zemax for each configuration

- Fine tune tilts and positions of all optics in interferometer using simple loops in Matlab
- Full aberration analysis

Access to gaussian beam analysis tools

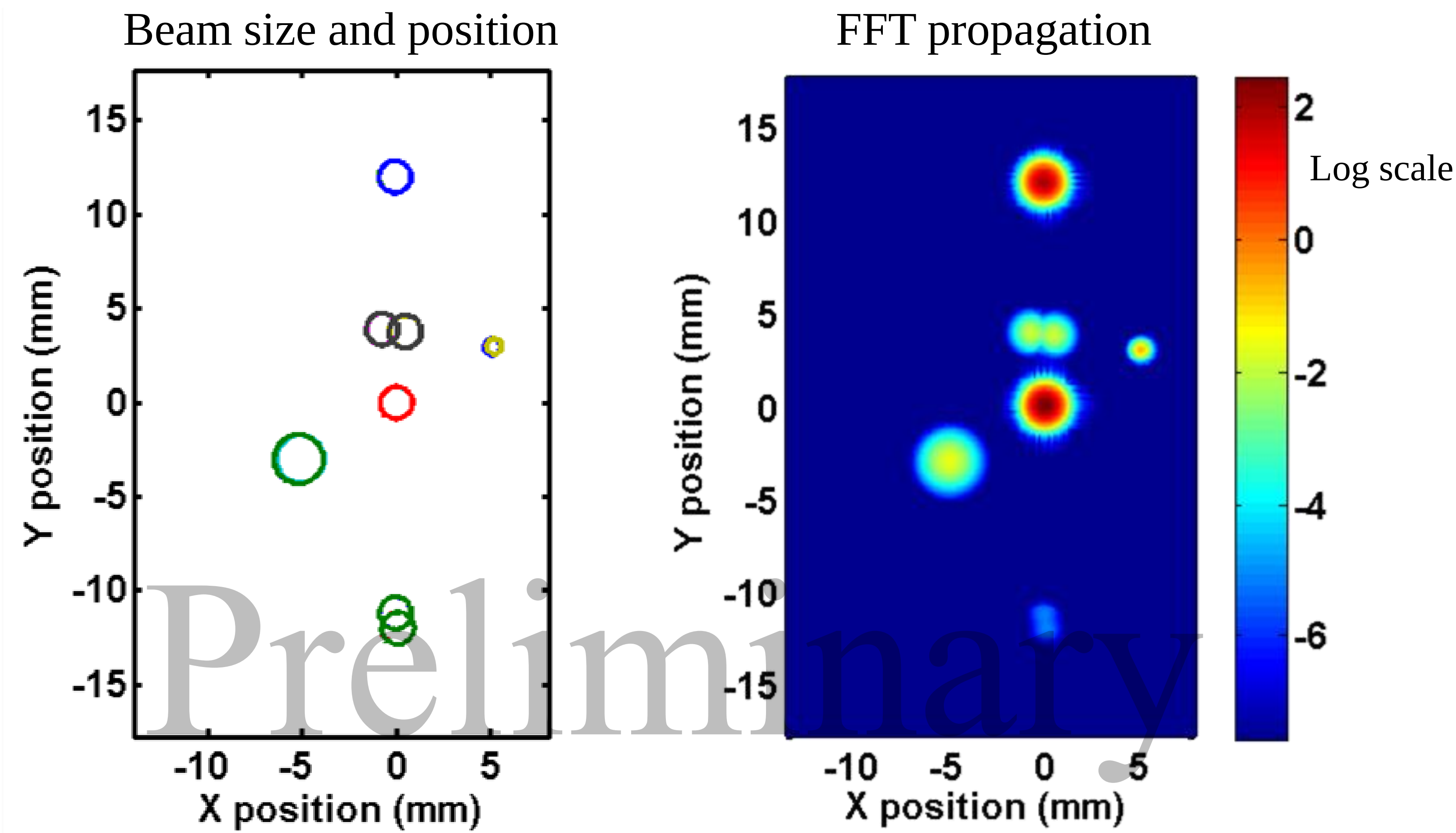
- Paraxial gaussian beams
- FFT propagation taking into account geometric aberrations of real optics

Combination of these tools allows us to simulate the pickoffs in CITF

Pick offs in CITF

Use telescope on POP (like Virgo+ end bench telescope)

Build 24 configurations and for each one run analysis



Conclusion

Developed Matlab script to build sequential configurations in Zemax

Simple interface to define optical configuration in 3D

Algorithm to determine all possible beam paths

Automated configuration analysis allows modeling of pickoffs

But:

Validation of current configurations required

Parabolic mirrors not yet integrated

I will make a version of the script available.

Antonino Chiummo has accepted to be the reference person for the scripts


So Long,
and thanks for all the Fish.