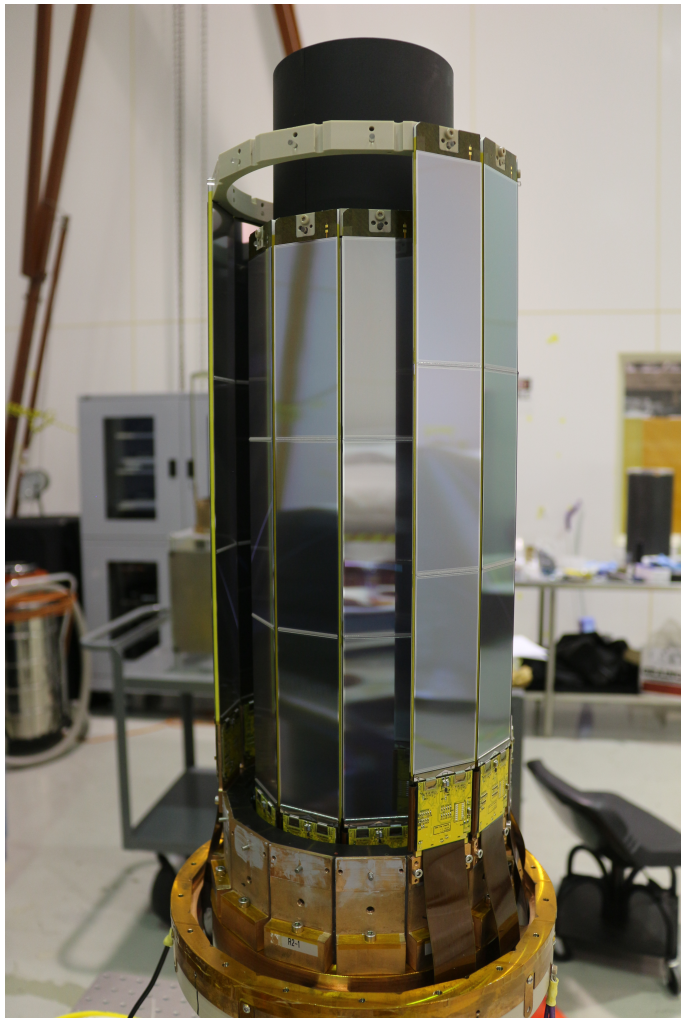
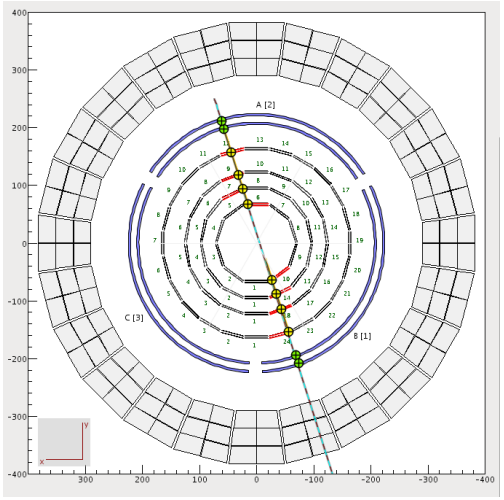
SVT Track-Based Alignment

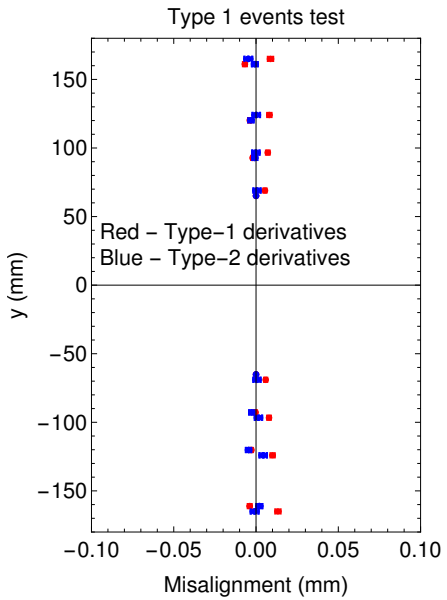
1. Goal: Align the SVT to reach the resolution design specification of $\approx 65\ \mu m$.
2. Build accurate and complete representation of the SVT geometry and materials as part of the CLAS12 Common Tools.
3. Provide the geometry for the *gemc* simulation and the CLAS12 reconstruction from a common set of parameters.
4. Develop algorithms to measure and correct misalignments in the SVT.
5. Document it.



Status of SVT alignment study

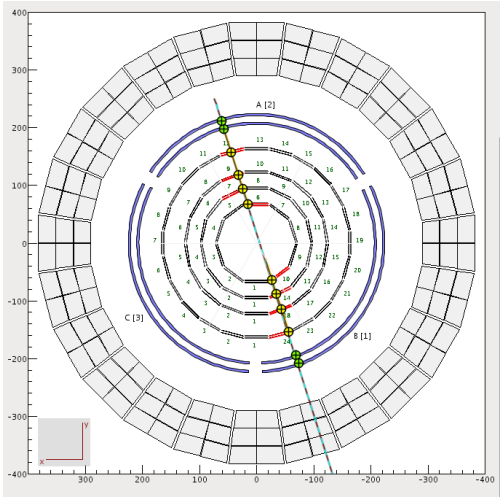
1. Type 2 code written and being tested. Using Type-1 events.
2. Comparison of Type-1 events analyzed with Type-2 code useful for identifying bugs, picking signs of derivatives, *etc*.
3. *gemc* version 4a.1.0 in use, Java/Groovy scripts at coatjava 4a.5.5.
4. Applied to simulated, Type-1 cosmic rays with the ideal geometry.
5. Misalignments extracted from `millepede` close to zero within uncertainties (blue points in plot).
6. Misalignments extracted from `millepede` using old Type-1 code show clear bias. Effect of new geometry in *gemc*?

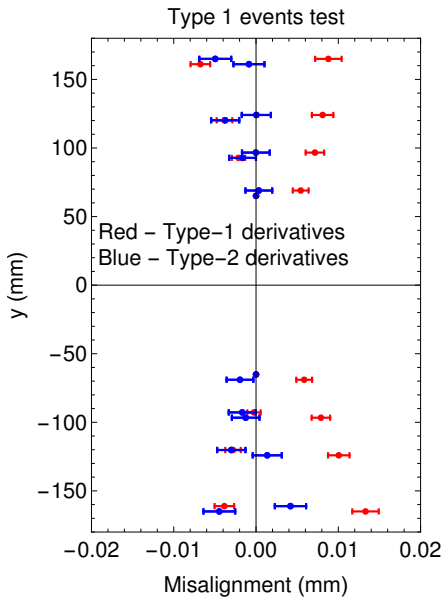




Status of SVT alignment study

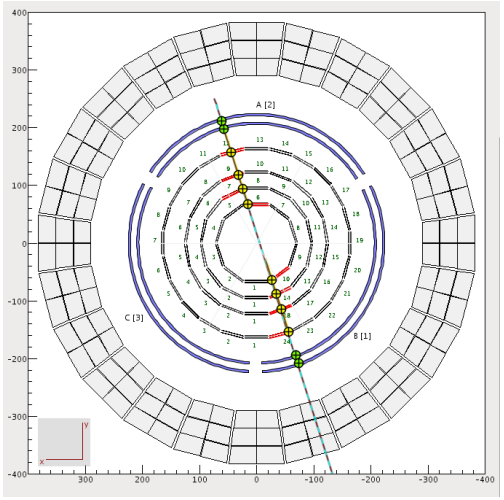
1. Type 2 code written and being tested. Using Type-1 events.
2. Comparison of Type-1 events analyzed with Type-2 code useful for identifying bugs, picking signs of derivatives, *etc*.
3. *gemc* version 4a.1.0 in use, Java/Groovy scripts at coatjava 4a.5.5.
4. Applied to simulated, Type-1 cosmic rays with the ideal geometry.
5. Misalignments extracted from `millepede` close to zero within uncertainties (blue points in plot).
6. Misalignments extracted from `millepede` using old Type-1 code show clear bias. Effect of new geometry in *gemc*?

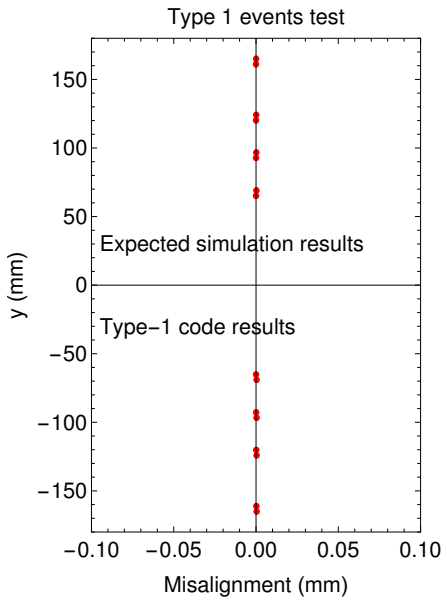




Status of SVT alignment study

1. Type 2 code written and being tested. Using Type-1 events.
2. Comparison of Type-1 events analyzed with Type-2 code useful for identifying bugs, picking signs of derivatives, *etc*.
3. *gemc* version 4a.1.0 in use, Java/Groovy scripts at coatjava 4a.5.5.
4. Applied to simulated, Type-1 cosmic rays with the ideal geometry.
5. Misalignments extracted from `millepede` close to zero within uncertainties (blue points in plot).
6. Misalignments extracted from `millepede` using old Type-1 code show clear bias. Effect of new geometry in *gemc*?



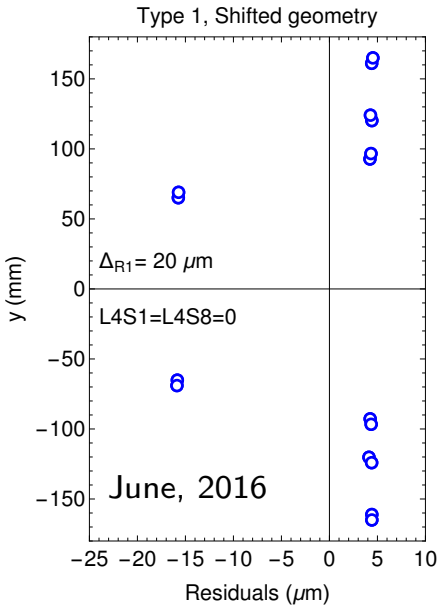


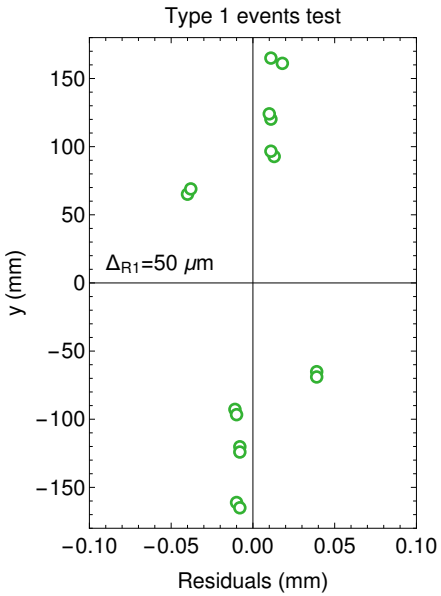
Status of SVT alignment study

1. Apply known shift to region 1 in *gemc* - $\Delta x = 50\ \mu m$.
2. *gemc* syntax:

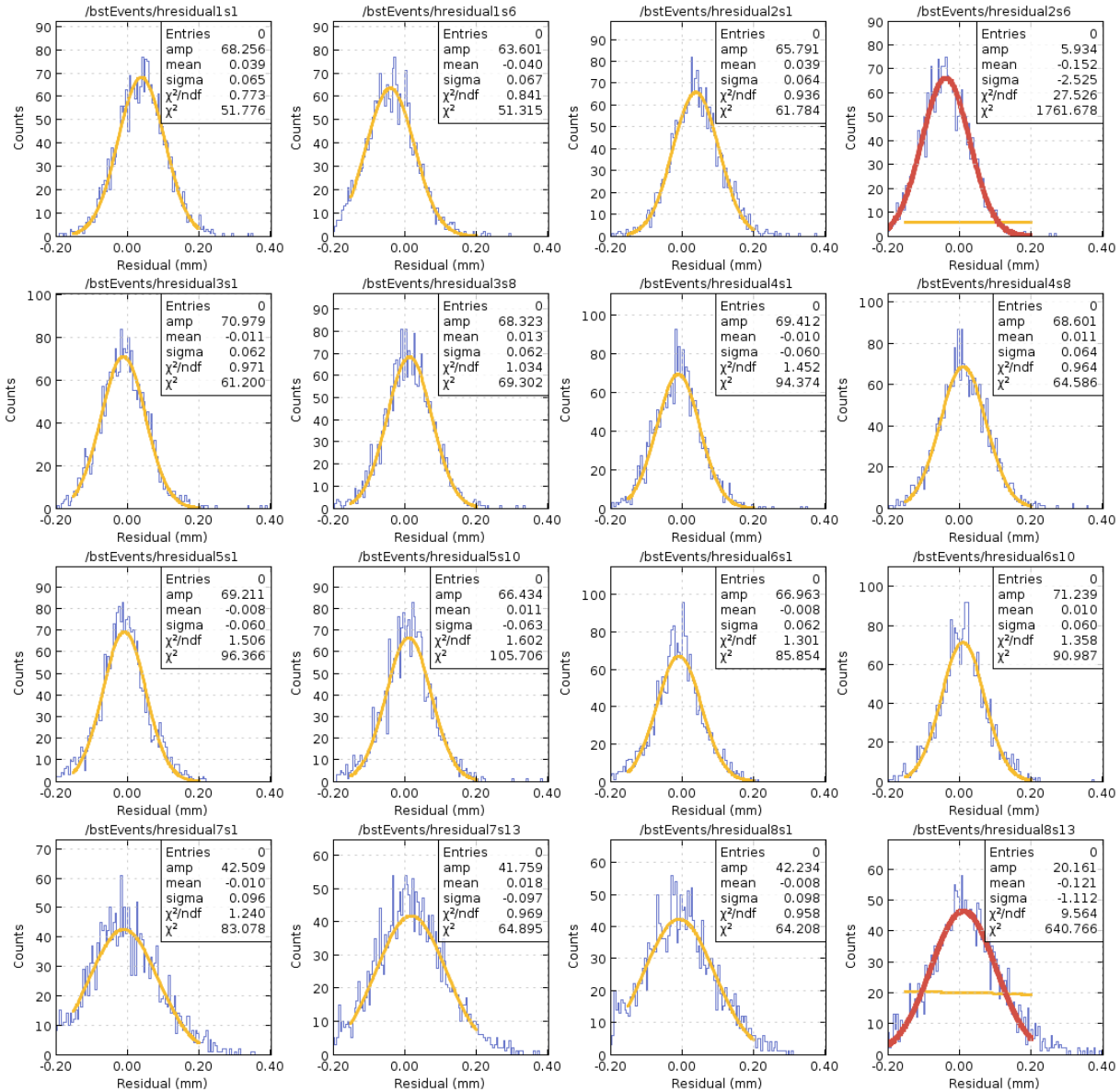
```
<detector name="region1">
  <position x="0.050*mm" y="0*cm" z="0*cm" />
</detector>
```

3. Change in definition of residual sign in BSTRec::Hits bank?





Residuals for Type-1 Events, 50-micron shift



Status of SVT alignment study

1. First application of code to Type-2 events.
2. Type-1 events have small residuals in BSTRec::Hits bank.
3. Type-2 events have large residuals in BSTRec::Hits bank even for horizontal modules.

```
small residuals on type-1 events. event 32

Choose (n=next,p=previous, q=quit), Type Bank Name or id : BSTRec::Hits   for type 1 event
 SHOWING BANK
------------------------+--------------------------+
>>>> GROUP (group=     1) (name=BSTRec::Hits):
------------------------+--------------------------+
          ID (     INT) :        12       13       14       15       11       10        9
       layer (     INT) :         1        1        1        2        2        3        3
      sector (     INT) :         6        6        1        1        6        8        8
       strip (     INT) :        84       85      127       63      106       92       93
 fitResidual (   FLOAT) :    -0.079    0.131    0.026   -0.009    0.006   -0.064    0.138
  trkingStat (     INT) :        -1       -1       -1       -1       -1       -1       -1
   clusterID (     INT) :         1        1        2        3        4        5        5
       trkID (     INT) :         1        1        1        1        1        1        1
```

Type 1

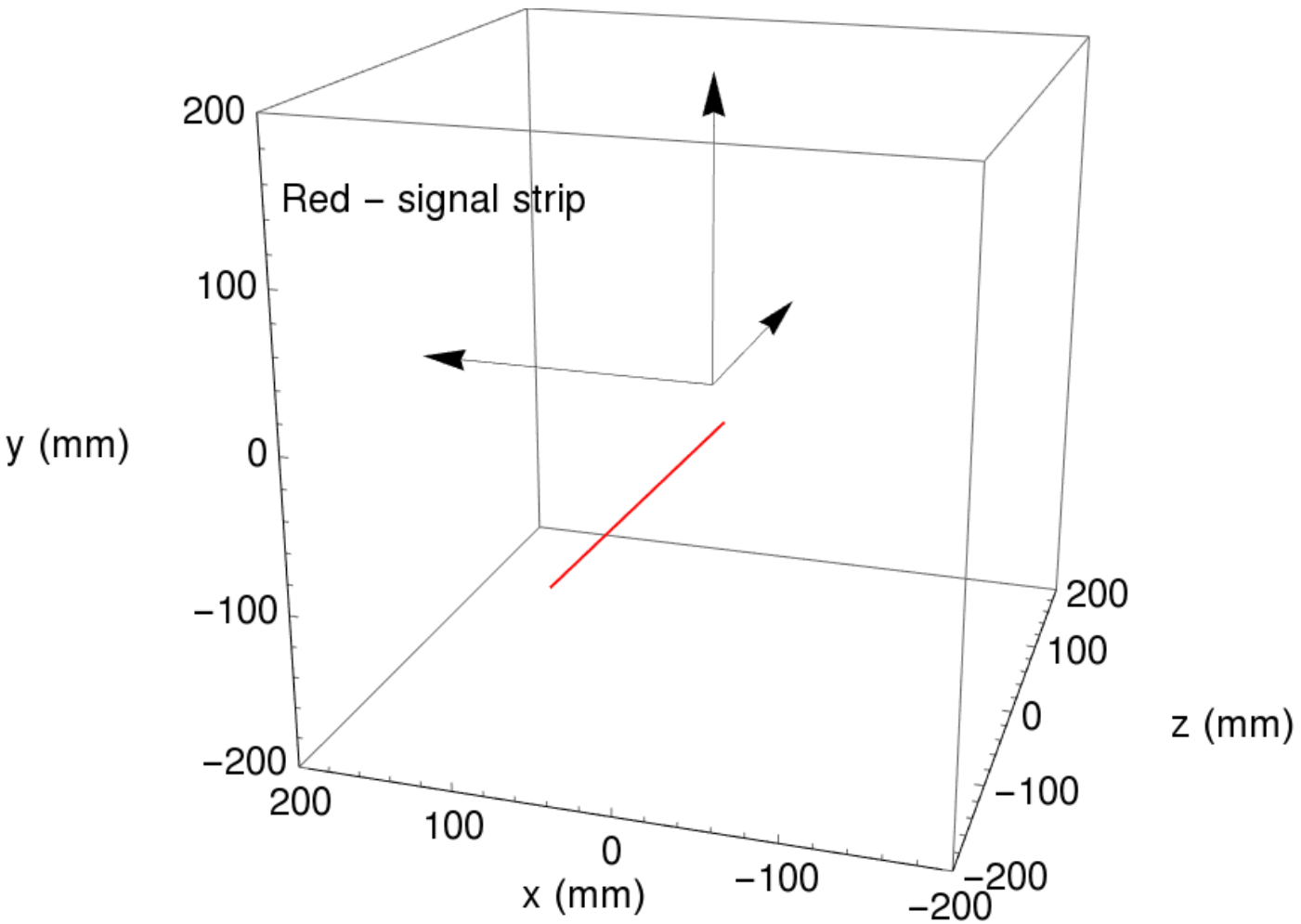
```
large residuals on type-2 eventss. event 17

Choose (n=next,p=previous, q=quit), Type Bank Name or id : BSTRec::Hits   for type 2 event
 SHOWING BANK
------------------------+--------------------------+
>>>> GROUP (group=     1) (name=BSTRec::Hits):
------------------------+--------------------------+
          ID (     INT) :        13       12       14       15       11       10        9
       layer (     INT) :         1        1        1        2        2        2        3
      sector (     INT) :         5        5        1        1        5        5        7
       strip (     INT) :       102      103      189       22      101      102       85
 fitResidual (   FLOAT) :     0.270    0.459   -0.440   -0.395    0.503    0.315    0.236
  trkingStat (     INT) :        -1       -1       -1       -1       -1       -1       -1
   clusterID (     INT) :         1        1        2        3        4        4        5
       trkID (     INT) :         1        1        1        1        1        1        1
------------------------+--------------------------+
```

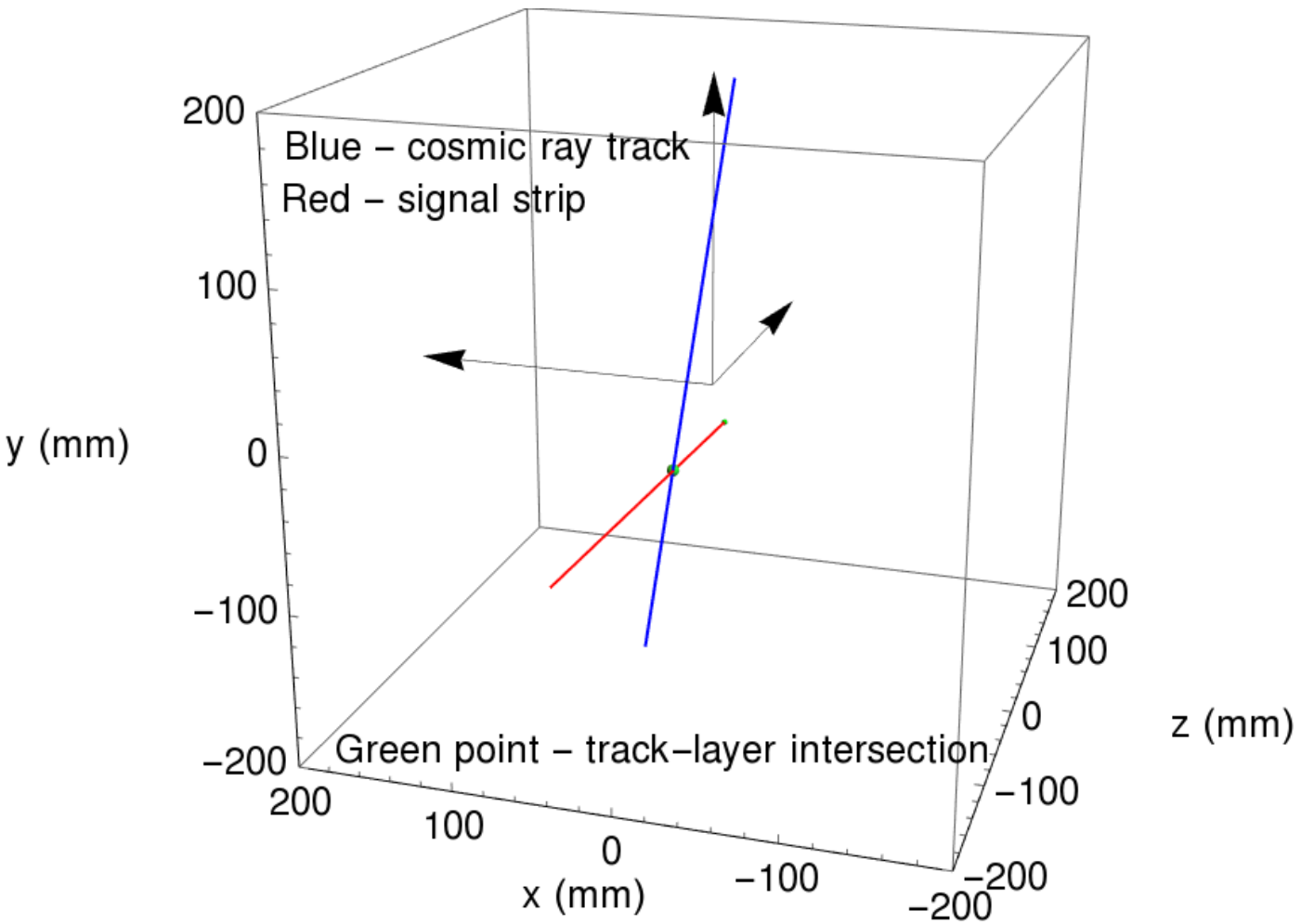
Type 2

Additional Slides

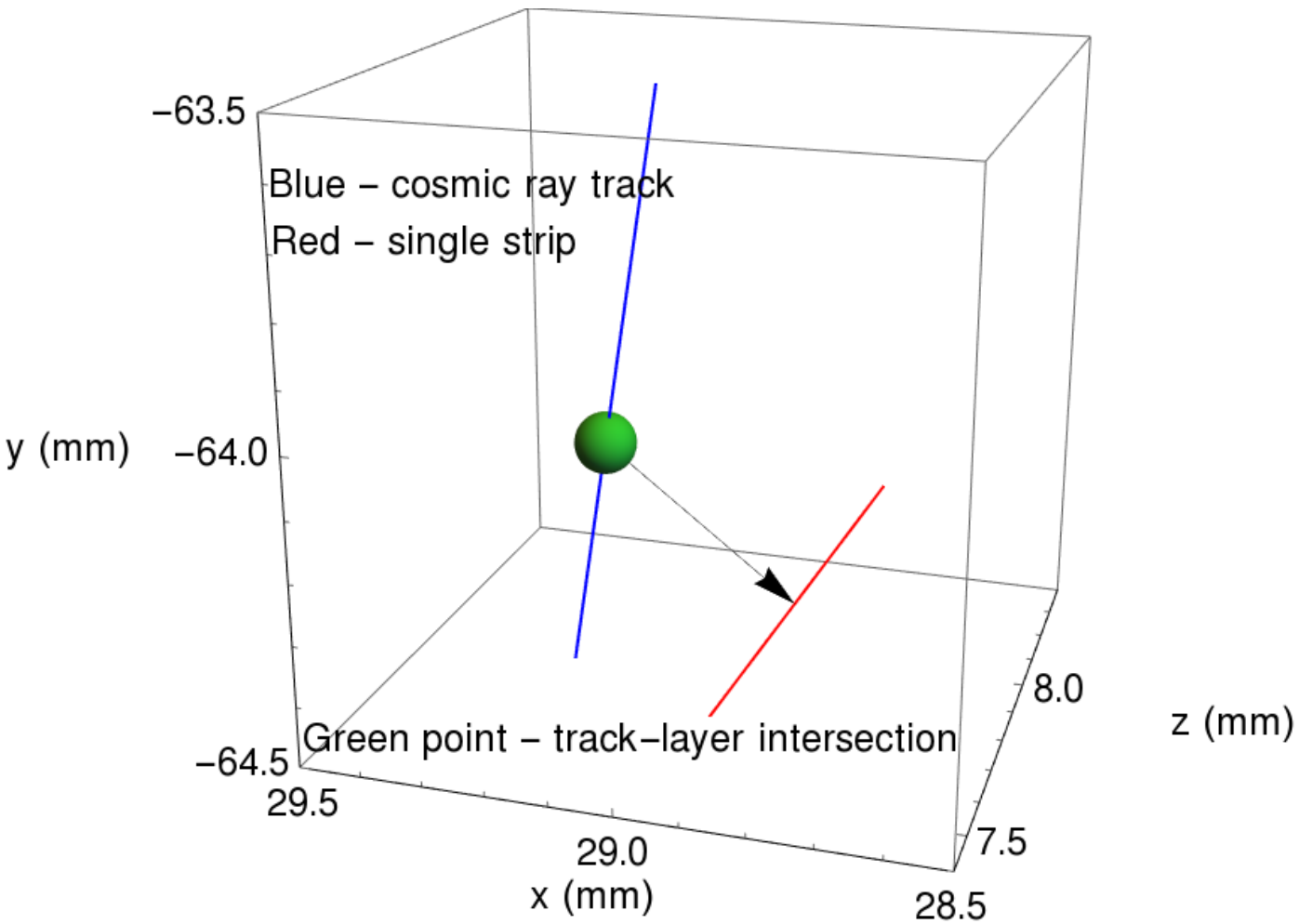
Visualization/Validation



Visualization



Visualization



Fun with *Mathematica*

$$\frac{dDOCAQ}{dmyz}$$

DOCA - distance of closest approach, *myz* - slope in $y - z$ plane.

```
double dDOCAdmyz=(2*(-((((dmyx + myx)*(-((((dmyx + myx)*(dmyz + myz)*Math.cos(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5)) - ((dmyz + myz)*Math.sin(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - ((dmyx + myx)*(dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - ((dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2),1.5)) - ((dmyx + myx)*(dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyz + myz)*Math.sin(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5) - ((dmyz + myz)*Math.sin(phis))/Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - ((dmyx + myx)*(dmyz + myz)*Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - (Math.pow(dmyz + myz,2)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))) + (Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))))))/Math.sqrt(Math.pow(xd - xu,2) + Math.pow(yd - yu,2) + Math.pow(zd - zu,2))) + ((xd - xu)*(-((((dmyx + myx)*(-((((dmyx + myx)*(dmyz + myz)*Math.cos(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5)) - ((dmyz + myz)*Math.sin(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - ((dmyx + myx)*(dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - (Math.pow(dmyz + myz,2)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))) + (Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))))))/Math.sqrt(Math.pow(xd - xu,2) + Math.pow(yd - yu,2) + Math.pow(zd - zu,2))) + ((zd - zu)*(-((((dmyx + myx)*(-((((dmyx + myx)*(dmyz + myz)*Math.cos(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5)) - ((dmyz + myz)*Math.sin(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - ((dmyx + myx)*(dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - (Math.pow(dmyz + myz,2)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))) + (Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))))))/Math.sqrt(Math.pow(xd - xu,2) + Math.pow(yd - yu,2) + Math.pow(zd - zu,2))) + 2*(-((((dmyz + myz)*(-((((dmyx + myx)*(dmyz + myz)*Math.cos(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5)) - ((dmyz + myz)*Math.sin(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - ((dmyx + myx)*(dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - (Math.pow(dmyz + myz,2)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))) + (Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))))))/Math.sqrt(Math.pow(xd - xu,2) + Math.pow(yd - yu,2) + Math.pow(zd - zu,2))) + ((dmyx + myx)*(dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - (Math.pow(dmyz + myz,2)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))) + (Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))))))/Math.sqrt(Math.pow(xd - xu,2) + Math.pow(yd - yu,2) + Math.pow(zd - zu,2))) + 2*((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(-((((dmyx + myx)*(-((((dmyx + myx)*(dmyz + myz)*Math.cos(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5)) - ((dmyz + myz)*Math.sin(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - ((dmyx + myx)*(dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - (Math.pow(dmyz + myz,2)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))) + (Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))))))/Math.sqrt(Math.pow(xd - xu,2) + Math.pow(yd - yu,2) + Math.pow(zd - zu,2))) + 2*(-((((dmyz + myz)*(-((((dmyx + myx)*(dmyz + myz)*Math.cos(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5)) - ((dmyz + myz)*Math.sin(phis))/Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - ((dmyx + myx)*(dmyz + myz)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*Math.pow(((((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))/2)) - (Math.pow(dmyz + myz,2)*(Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2)))/(Math.pow(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2),1.5))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)))) + (Math.cos(phis)*(byx + dbyx - Rl*Math.cos(phis)) - Rl*Math.pow(Math.sin(phis),2))/(Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))*(((dmyx + myx)*Math.cos(phis))/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2)) + Math.sin(phis)/Math.sqrt(1 + Math.pow(dmyx + myx,2) + Math.pow(dmyz + myz,2))))))/Math.sqrt(Math.pow(xd - xu,2) + Math.pow(yd - yu,2) + Math.pow(zd - zu,2)));
```