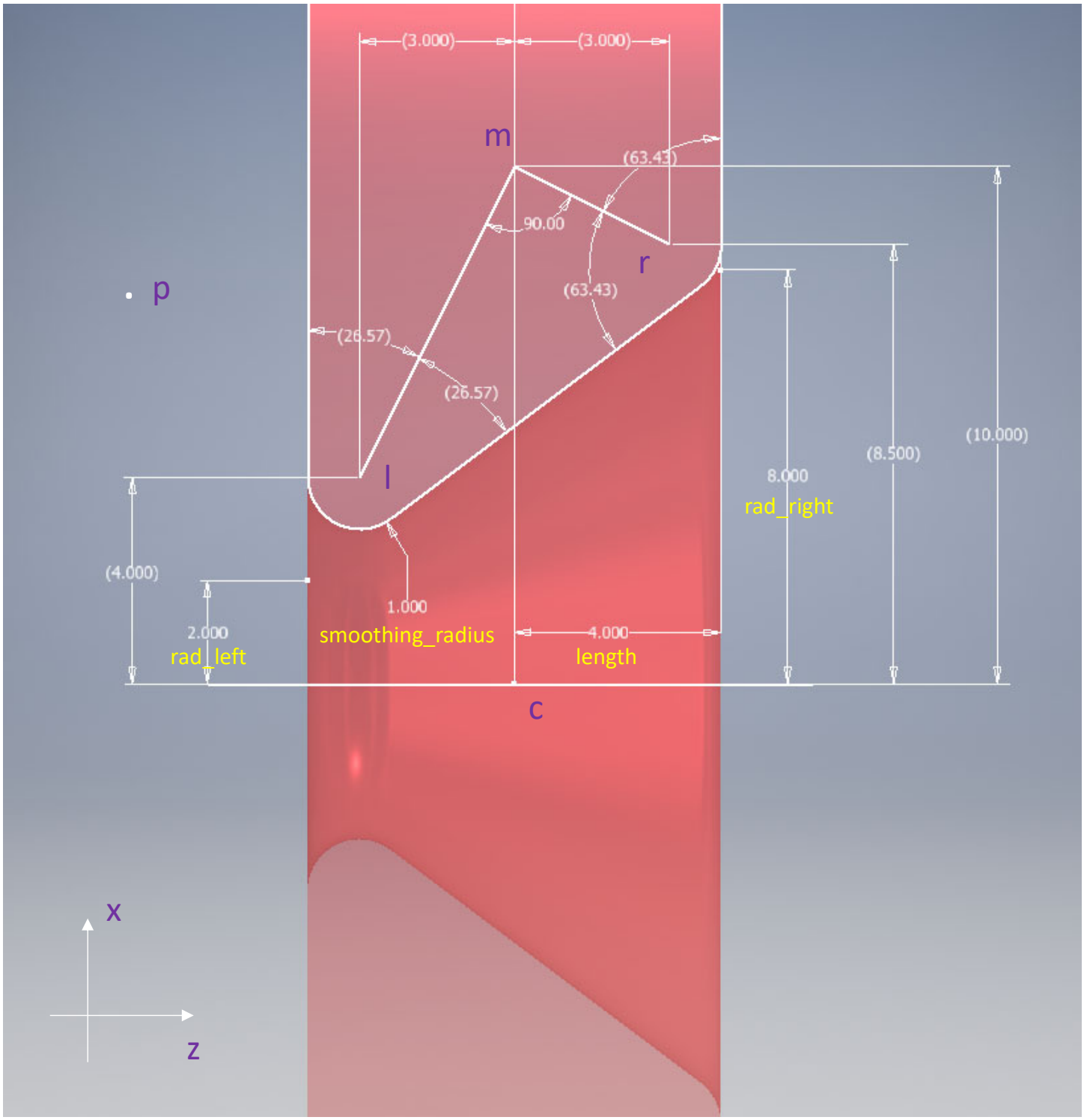
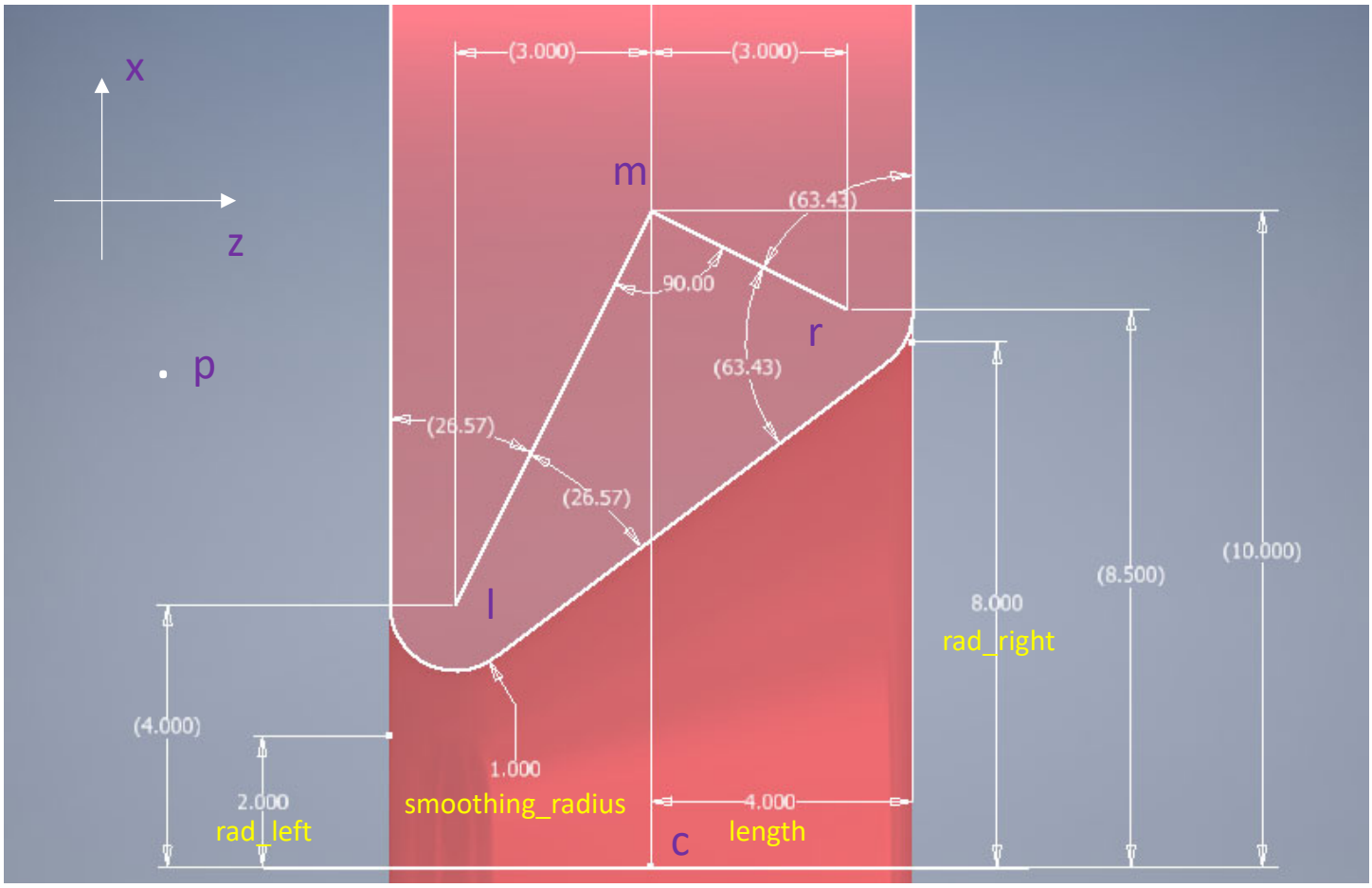
```
void calculate_pore_dist(Particle *p1, double ppos[3], Particle
*c_p, Constraint_pore *c, double *dist, double *vec)
```

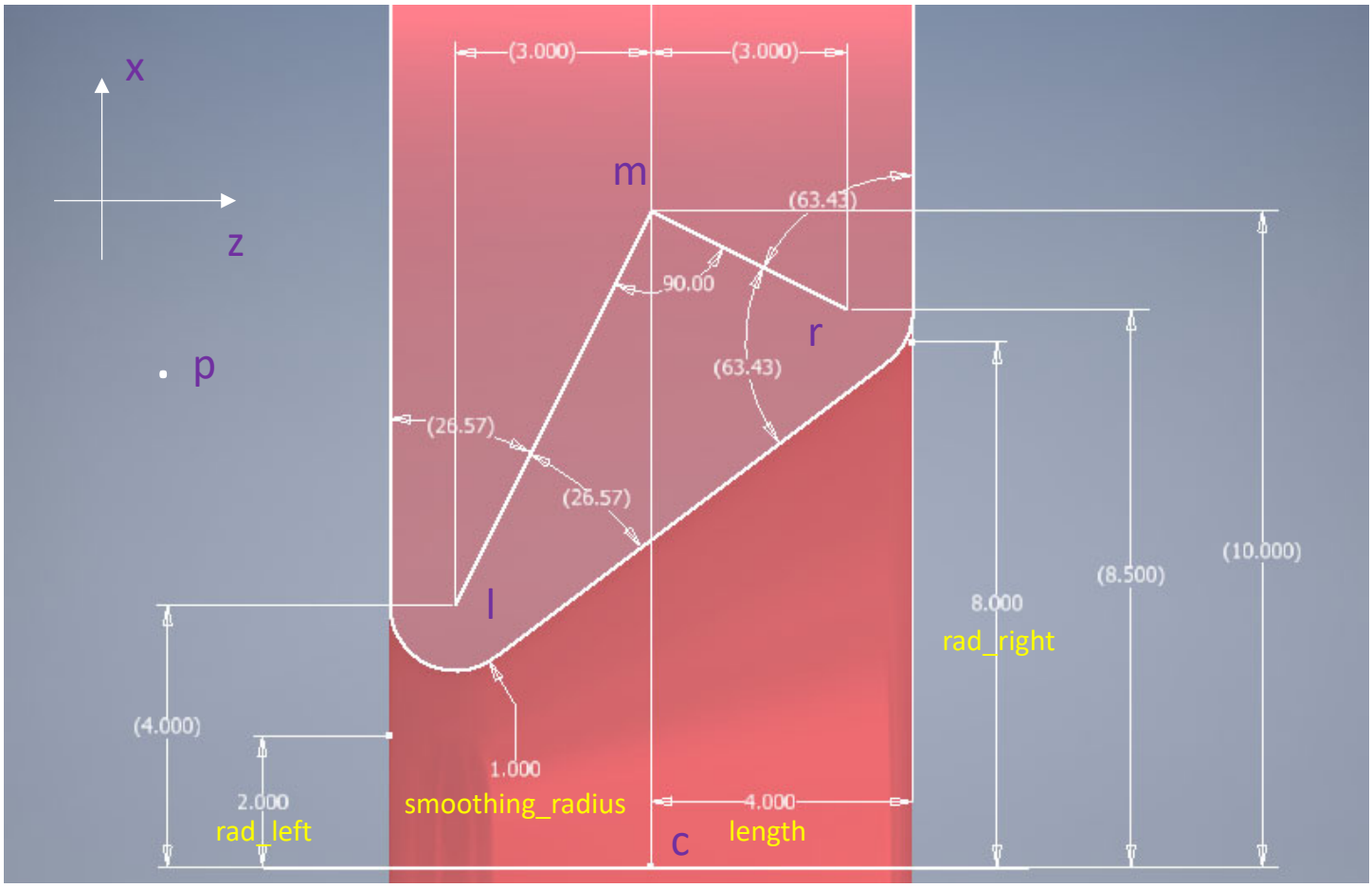


All happens in the upper half of this plane.
l and r are the centers of the smoothing circles
m is the intersection of those two angle bisectors



In the z-x coordinate system, we have

- k = (rad_right – rad_left) / 2. / length;
- sec_k = sqrt(1+k*k);
- cm_z = 0
- cm_x = (rad_right + rad_left) / 2. + length * sec_slope;
- cl_z = smoothing_radius - length;
- cl_x = rad_left + smoothing_radius * (sec_k + k);
- tan_lm = (cl_x - cm_x) / cl_z;
- tan_lp = (x - cl_x) / (z - cl_z);



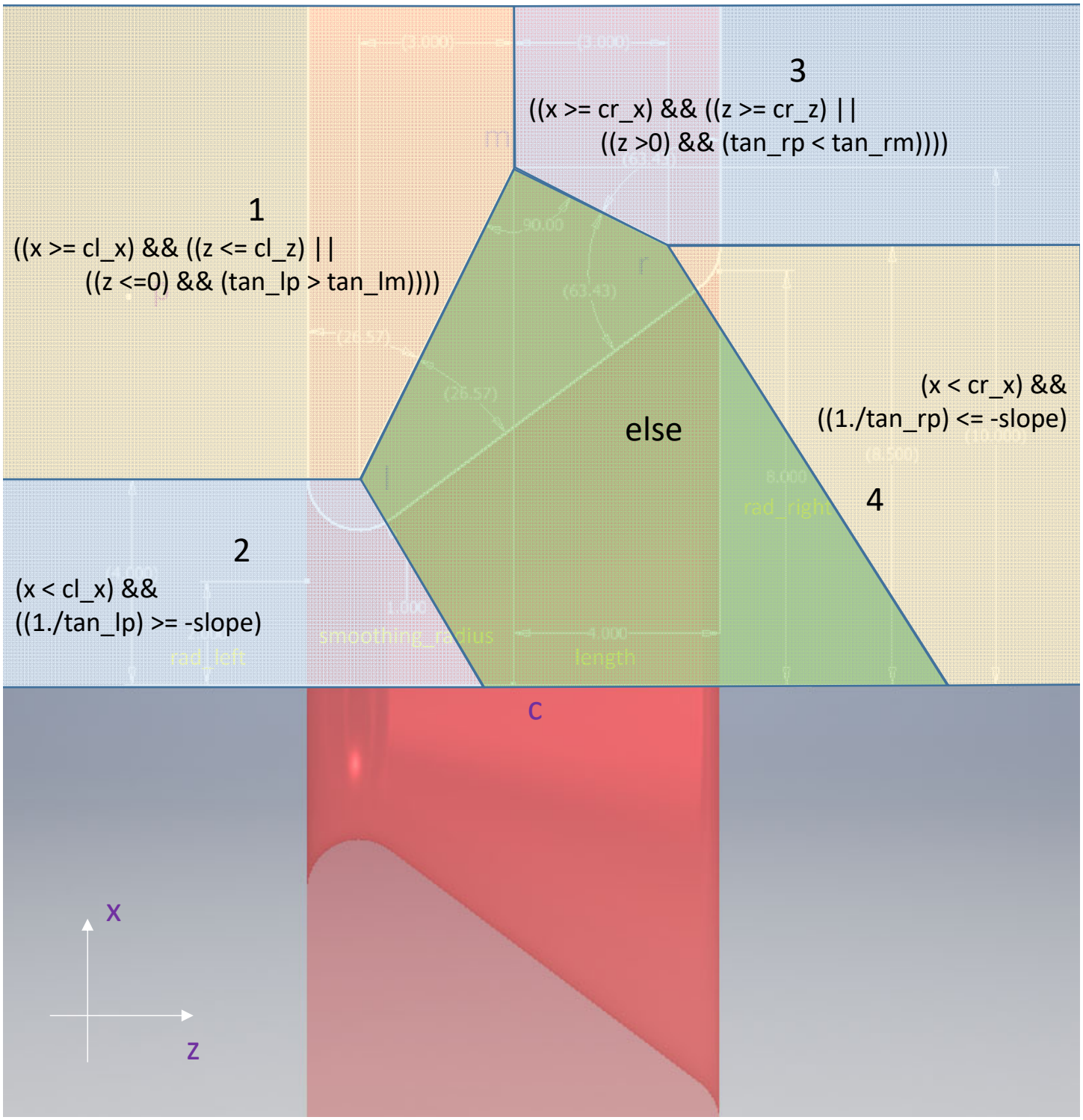
Similarly, on the right side , we have

- cr_z = length - smoothing_radius;
- cr_x = rad_right + smoothing_radius * (sec_k - k);
- tan_rm = (cr_x - cm_x) / cr_z;
- tan_rp = (x - cr_x) / (z - cr_z);

else, we have

1. *dist = (k*z - x + rad_middle) * cos_k;
2. vec[i] = *dist * (sin_k * e_z[i] - cos_k * e_x[i]);

```
void calculate_pore_dist(Particle *p1, double ppos[3], Particle
*c_p, Constraint_pore *c, double *dist, double *vec)
```



All happens in the upper half of this plane.
l and r are the centers of the smoothing circles
m is the intersection of those two angle bisectors

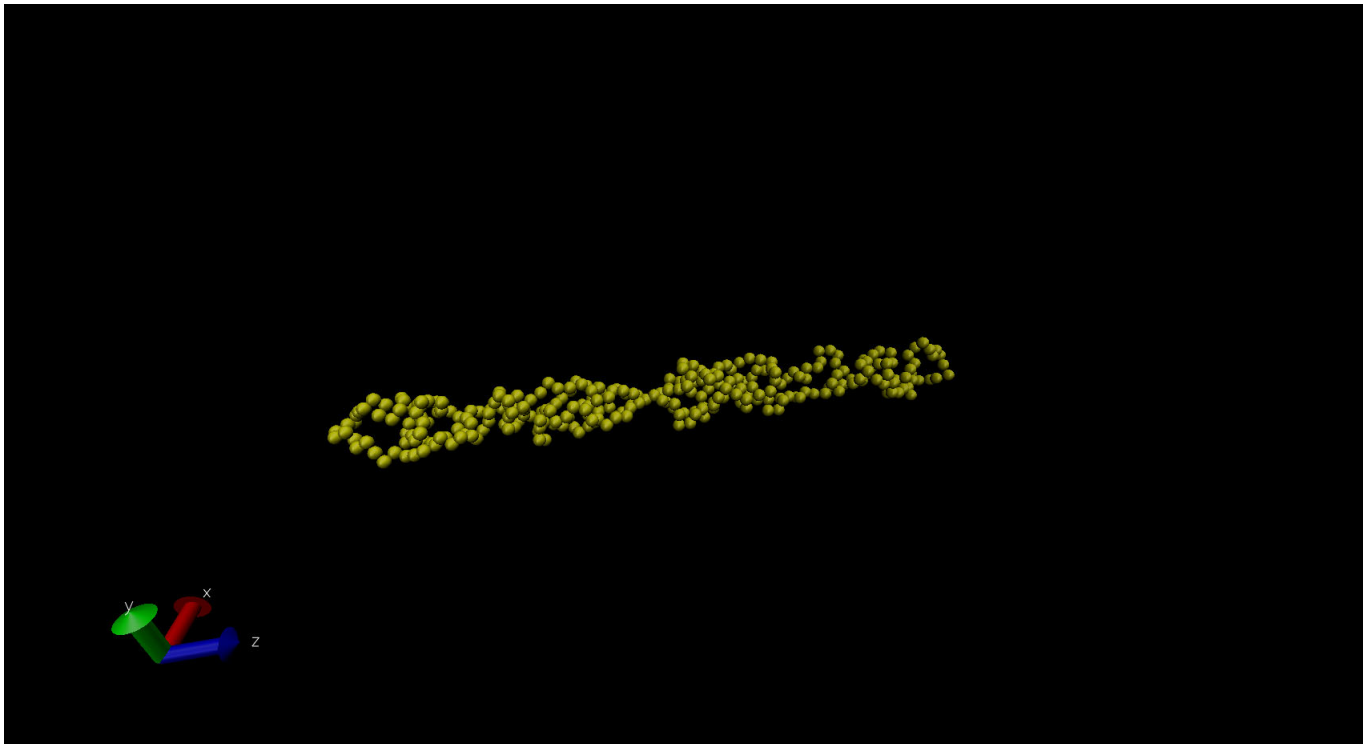
Test run with

constraint cylinder center $cyl_x $cyl_y $cyl_z axis 0 0 1 radius $cyl_outer_rad length $cyl_outer_l direction -1 type 3 penetrable 0 reflecting 1

constraint pore center $cyl_x $cyl_y $cyl_z axis 0 0 1 radius $cyl_inner_rad length $cyl_inner_l type 3 reflecting 1 smoothing_radius $psr

psr = 1.0

cyl_inner_rad = 3.5 initially



Failed at pore radius = 0.83998;

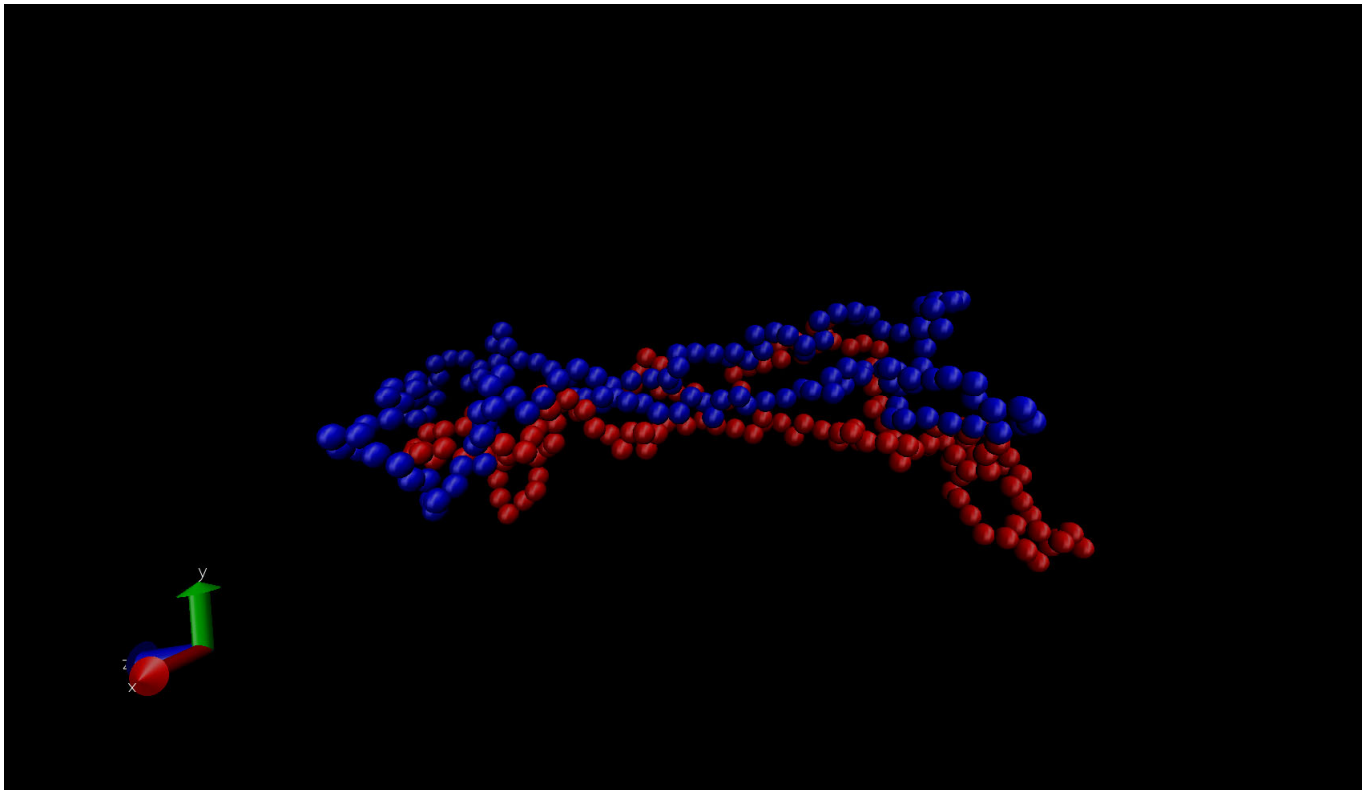
Would not run at smoothing_radius < length

Test run with

constraint pore center $cyl_x $cyl_y $cyl_z axis 0 0 1 radius $cyl_inner_rad length $cyl_inner_l type 3 reflecting 1 smoothing_radius $psr

psr = 15.0

cyl_inner_rad = 3.5 initially



Failed at pore radius = 3.396;

Would not run at smoothing_radius < length