Measuring short molecules: problems with non-linear traps

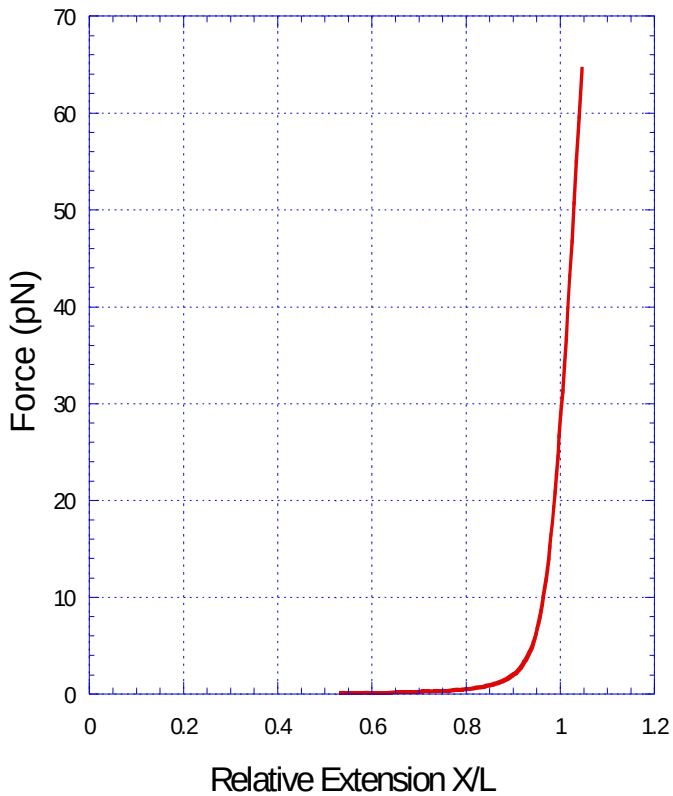
Steve Smith

Bustamante Lab Group Meeting

Aug. 15, 2008

dsDNA modeled as extensible WLC

Brochard & Croquette, Biophysical J, 1998

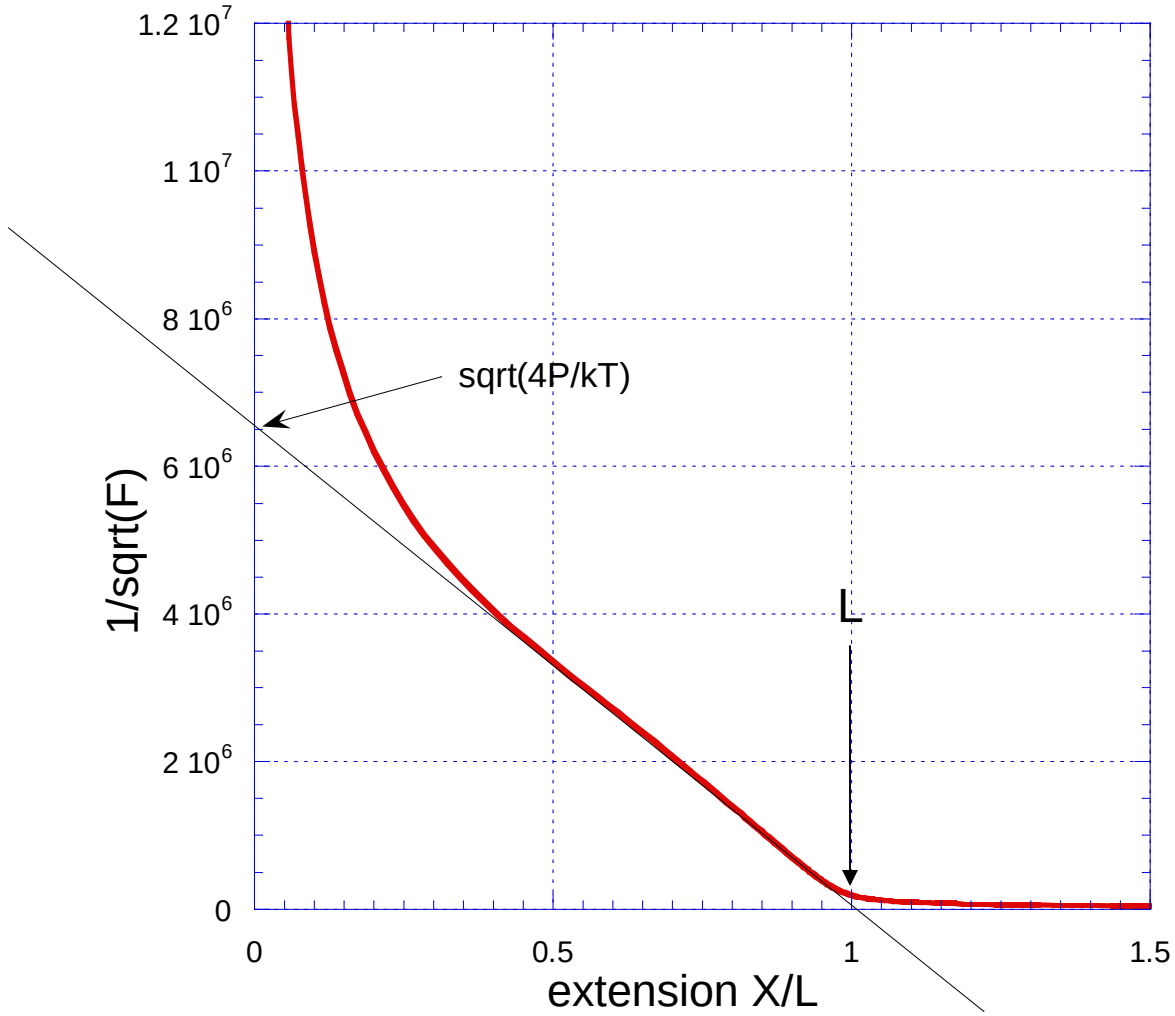


Persistence length P = 50nm
Contour Length L = 0.34 nm/bp
Stretch modulus S = 1000 pN

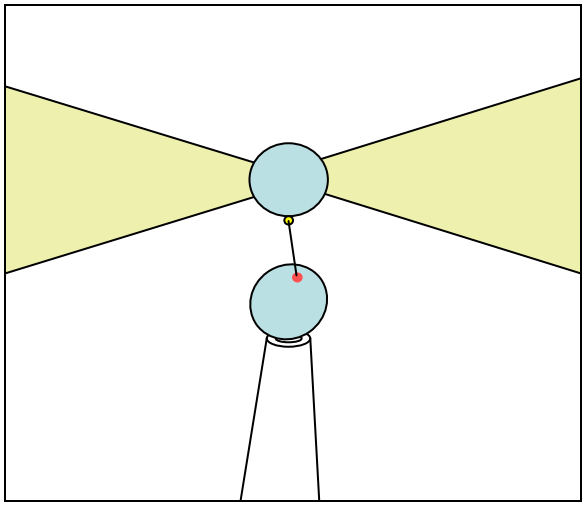
$$F = \frac{k_B T}{P}\left[\frac{1}{4(1-x/L)^2} + \frac{x}{L} - \frac{1}{4}\right]$$

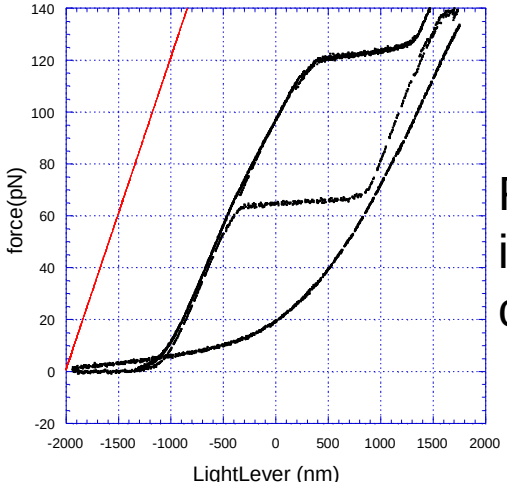
$$F \rightarrow \frac{1}{\left(1-\frac{x}{L}\right)^2}$$

Extracting P and L with 1/sqrt(F) plot



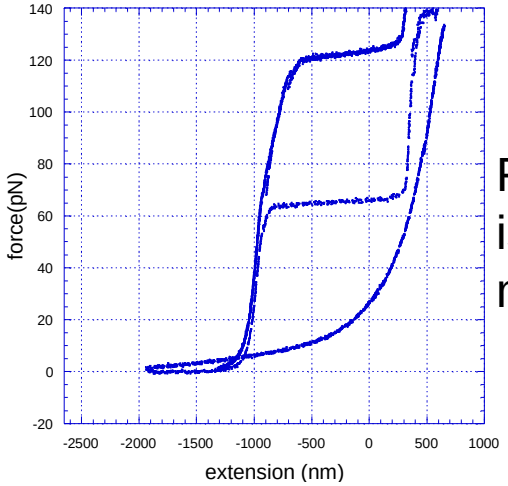
Measuring force vs. extension





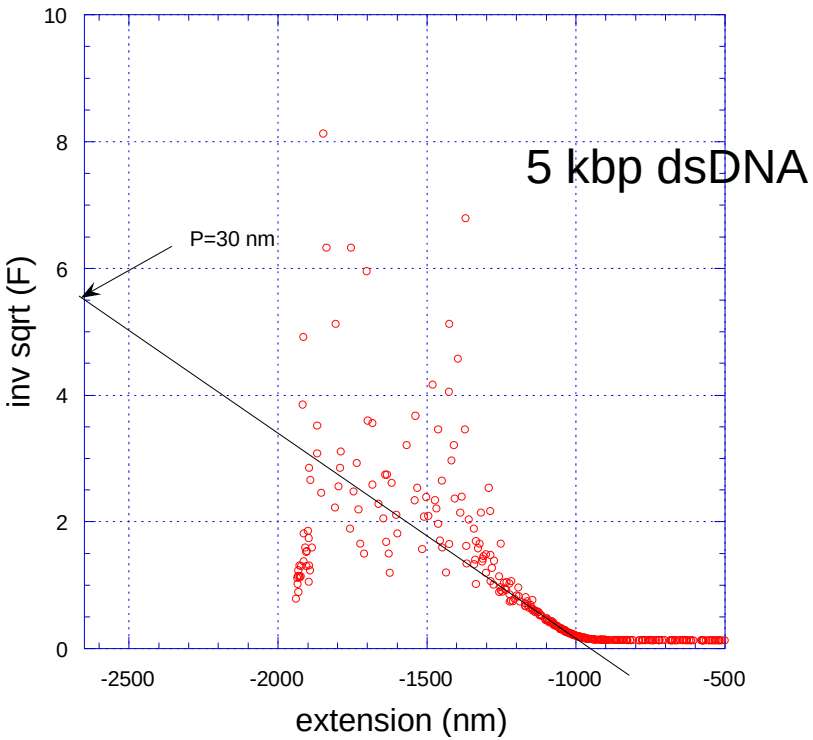
Force vs position includes trap/bead compliance

Same data with compliance removed



Force vs extension is basis for WLC model

Real data in 1/sqrt(F) plot show thermal noise



Note the persistence length (Y-intercept) seems too low for dsDNA

Everybody has the same problem for short molecules

Biophysical Journal Volume 93 December 2007 4099

New and Notable

The Great Hunt For Extra Compliance

Jan Liphardt
Physics Department, University of California, Berkeley, California

Every field has skeletons in their closet. One of the skeletons of single molecule biophysics is the persistence length of short DNA molecules. Loosely speaking, the persistence length, P, is a statement about the elasticity of a polymer; an infinitely stiff polymer has an infinite P. Knowledge of the persistence length of DNA and RNA is essential to design and interpret single-molecule experiments. Two models of polymer elas-

Nonetheless, one thing was sacred: the persistence length of DNA. The persistence length certainly did depend on ionic strength, pH, and intercalators (3–5), but incoming graduate students were told that P captured an intrinsic property of DNA and did not depend on the polymer's length. Most practitioners suspected that this was not true, since the concept of a persistence length becomes ill-defined when the polymer is not much longer than P. Moreover, many experimental studies of short pieces of DNA revealed that it was more compliant than predicted from the classic WLC parameterized with the canonical value of P, 50 nm.

What we did not know was how P

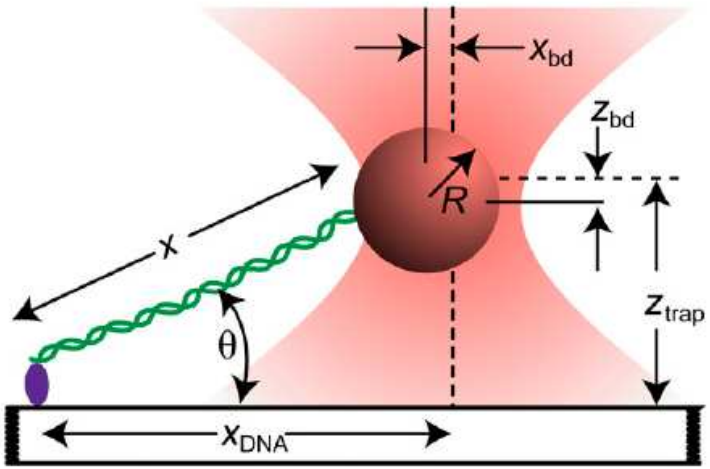
finite-WLC does rather well. T
of Seol et al. contains extensi
imental results that establish
formance of their finite-WLC

Is the finite-WLC the end of
Fortunately not—a multitude
tial biological processes take
length scales far below one p
length, and we are only beg
understand the intricacies of
RNA elasticity on short and
scales. Nonetheless, the finite-W
us a glimpse of what future
DNA elasticity will look like.
that, but Seol et al. also gives
print for constructing elasticit
for specific experimental geom
length scales.

Biophysical Journal Volume 93 December 2007 4360–4373

Elasticity of Short DNA Molecules: Theory and Experiment for Contour Lengths of 0.6–7 μm

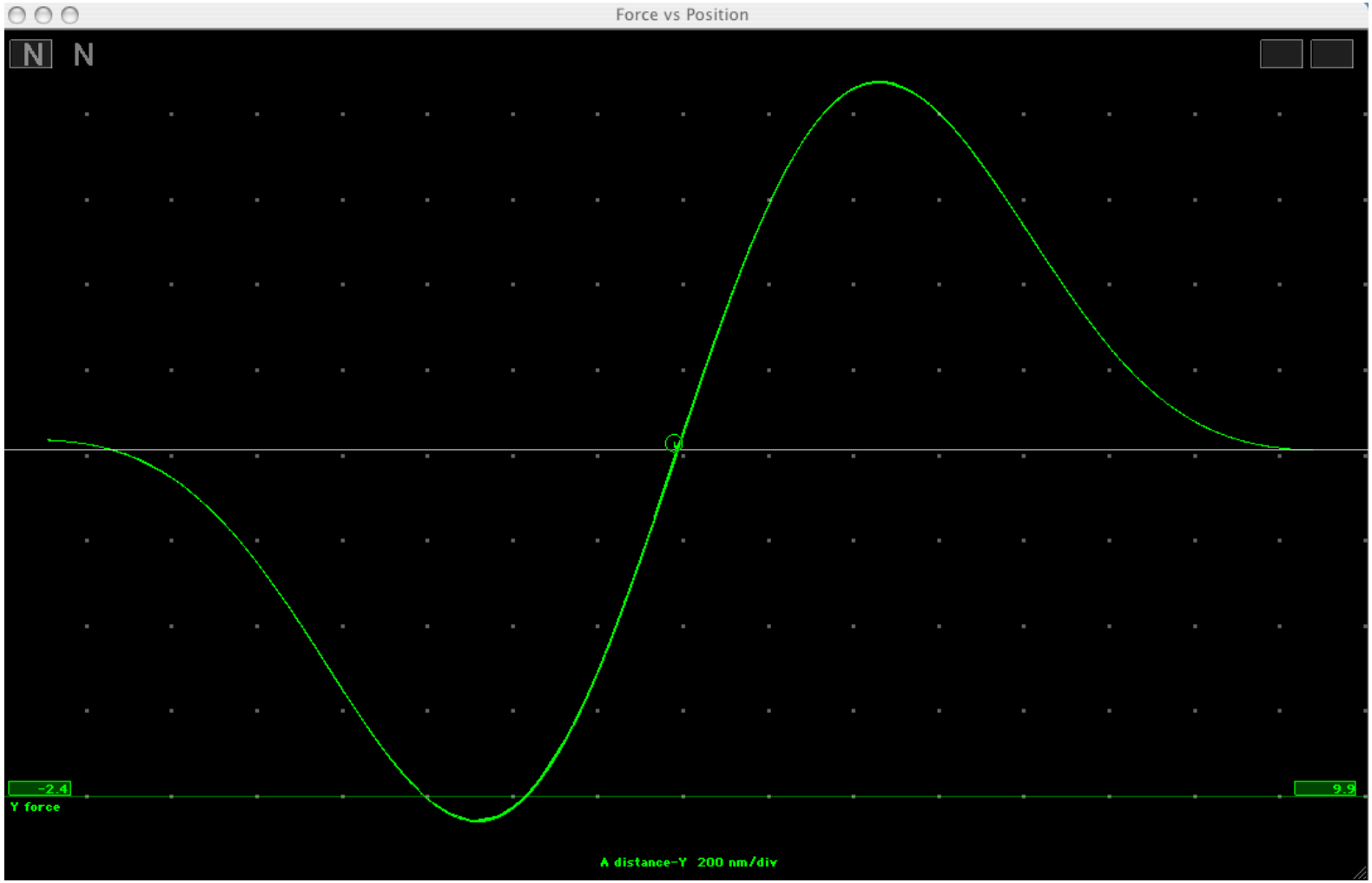
Yeonee Seol,* Jinyu Li,† Philip C. Nelson,¶ Thomas T. Perkins,*‡ and M. D. Betterton§

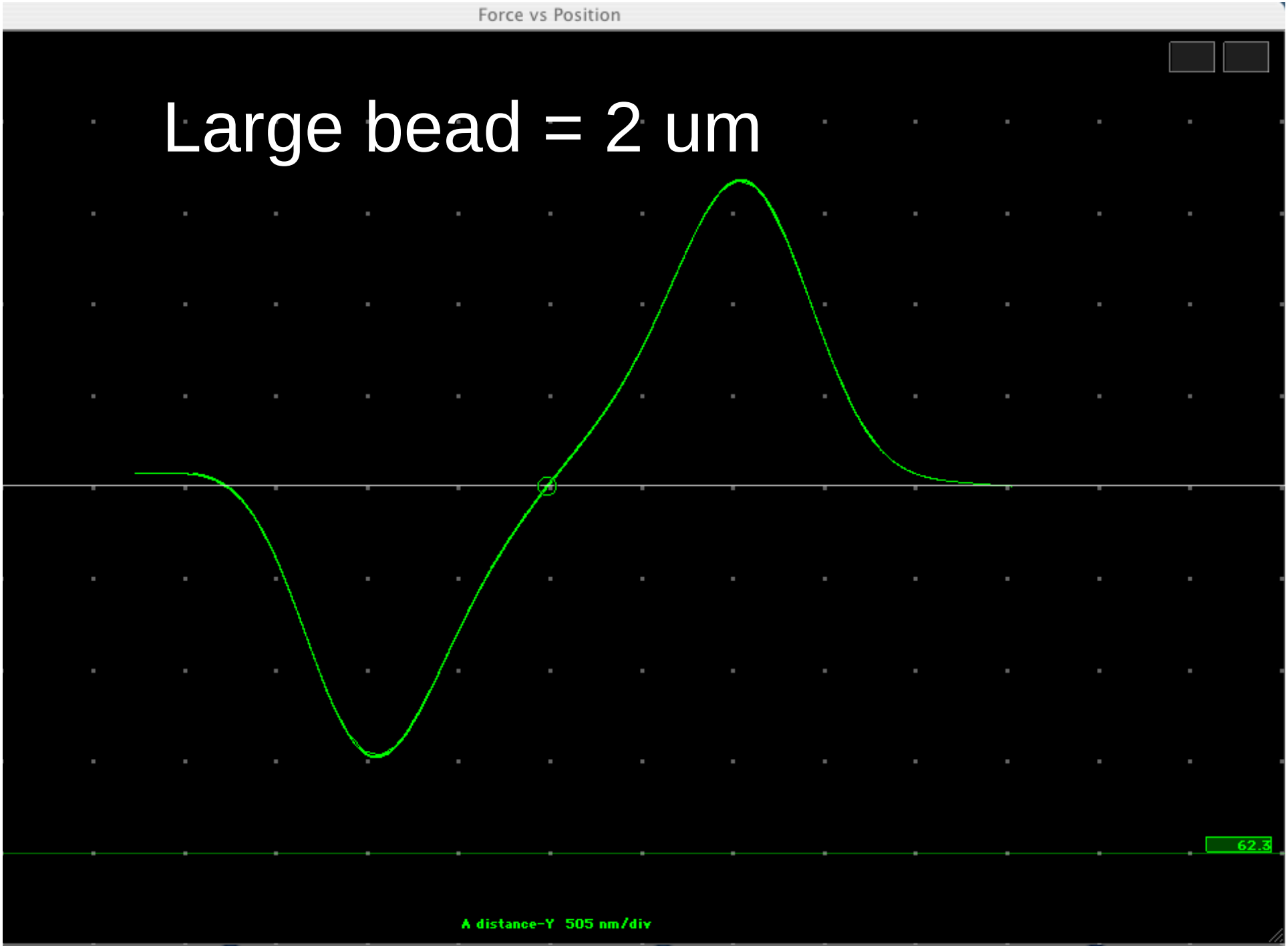


For instance see Seol et al.
Is the trap stiffness linear?
Yes, position clamp avoids problems with trap linearity.

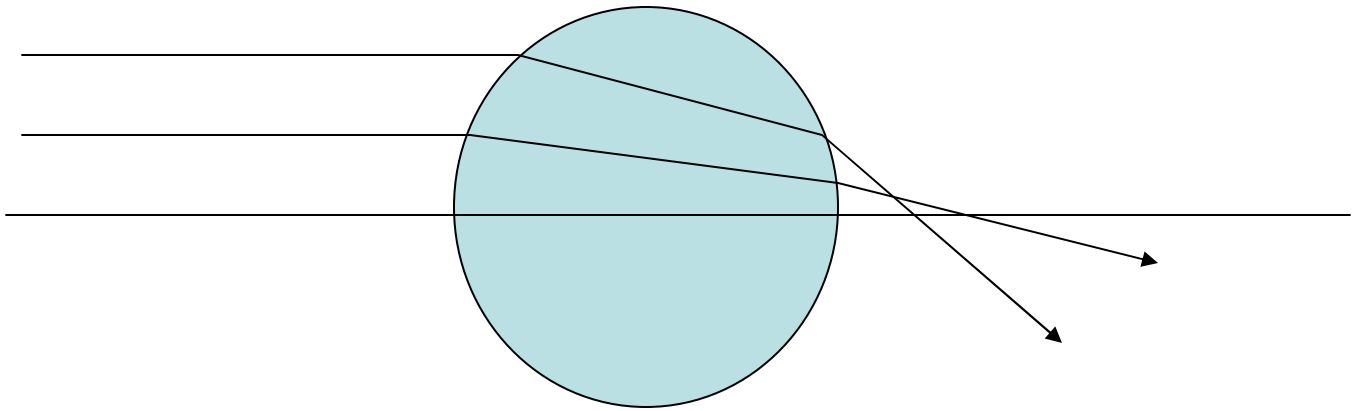
How linear is our trap?

Small bead = 0.27 um




Force vs Position
Large bead = 2 um
62.3
A distance-Y 505 nm/div

Spherical aberration means marginal rays fall short, thus exerting extra force



Sin(theta) > offset distance
Force > offset distance

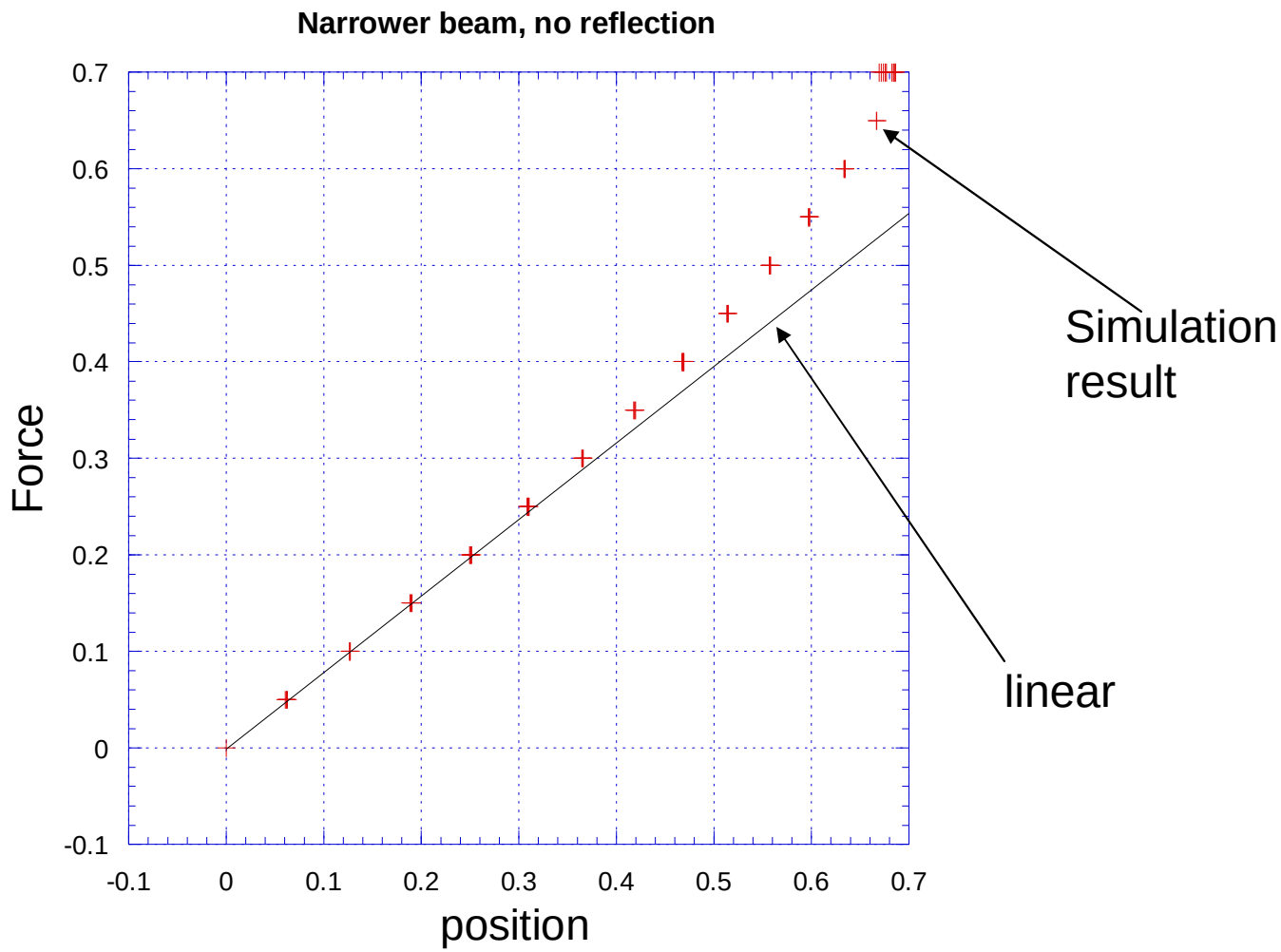
Ray-optics simulation uses Snell's law plus momentum changes to demonstrate force versus offset

```
if (photoCount mod transmitRate) = 0 then
    begin {photon is reflected}
    Px:=tangPx-normPx;
    Py:=tangPy-normPy;
    color:=white;  inc(reflCtr);
    end
  else
    begin {photon is refracted}   inc(refrCtr);
    inHiIndex:=insideBead; {adjust to new index}
    {tangPx and tangPy stay the same by Snell's law}
    if inHiIndex {but magnitude of P increases from 1 to refInd}
      then begin
      normPx:=normX * sqrt(sqr(relInd)-sqr(tangPx)-sqr(tangPy));
      normPy:=normY * sqrt(sqr(relInd)-sqr(tangPx)-sqr(tangPy));
      Px:=-normPx+tangPx;
      Py:=-normPy+tangPy;
      end;
    if (not inHiIndex) and ((sqr(tangPx)+sqr(tangPy))<1)
      then begin   {magnitude of P decreases to one}
      normPx:=normX * sqrt(1-sqr(tangPx)-sqr(tangPy));
      normPy:=normY * sqrt(1-sqr(tangPx)-sqr(tangPy));
      Px:= normPx+tangPx;
      Py:= normPy+tangPy;
      end;
    end;
  targetX:=targetX+LightUmph*(oldPx-Px);{apply light force to bead}
  targetY:=targetY+LightUmph*(oldPy-Py);{due to momentum change}
  end;{if photon hits interface}
 end;  {for photon[ray]}
end;
```

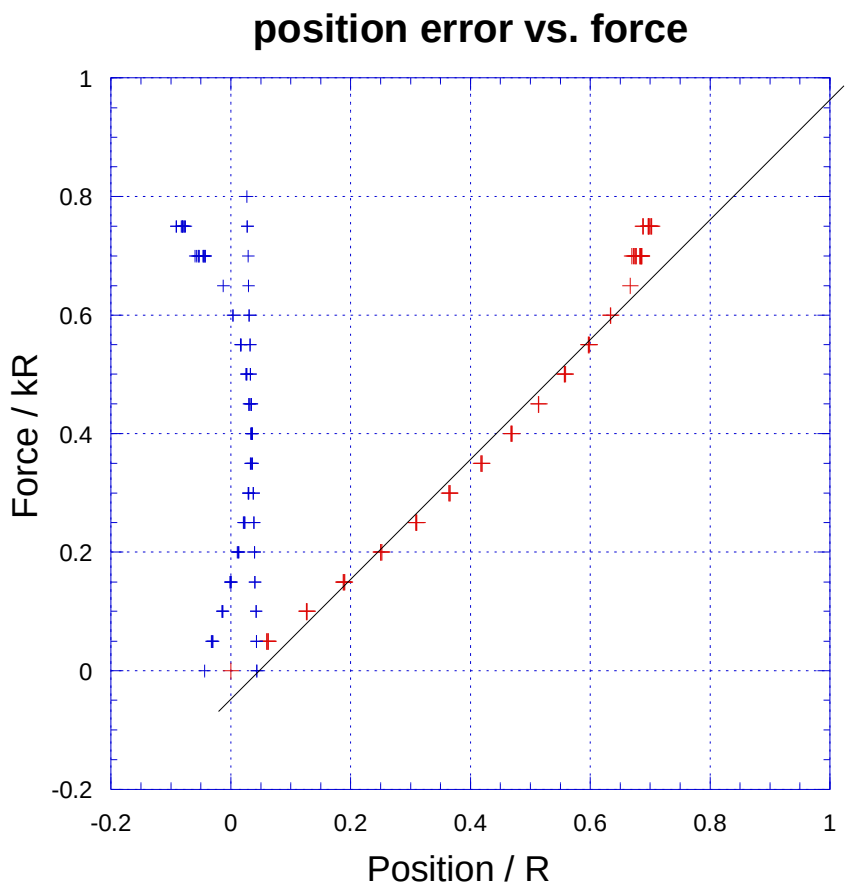
C:\Documents and Settings\Steven Smith\My Documents\Group\PHOTONS8.EXE

See "Photons" program in software section of TweezersLab web site

Simulation result

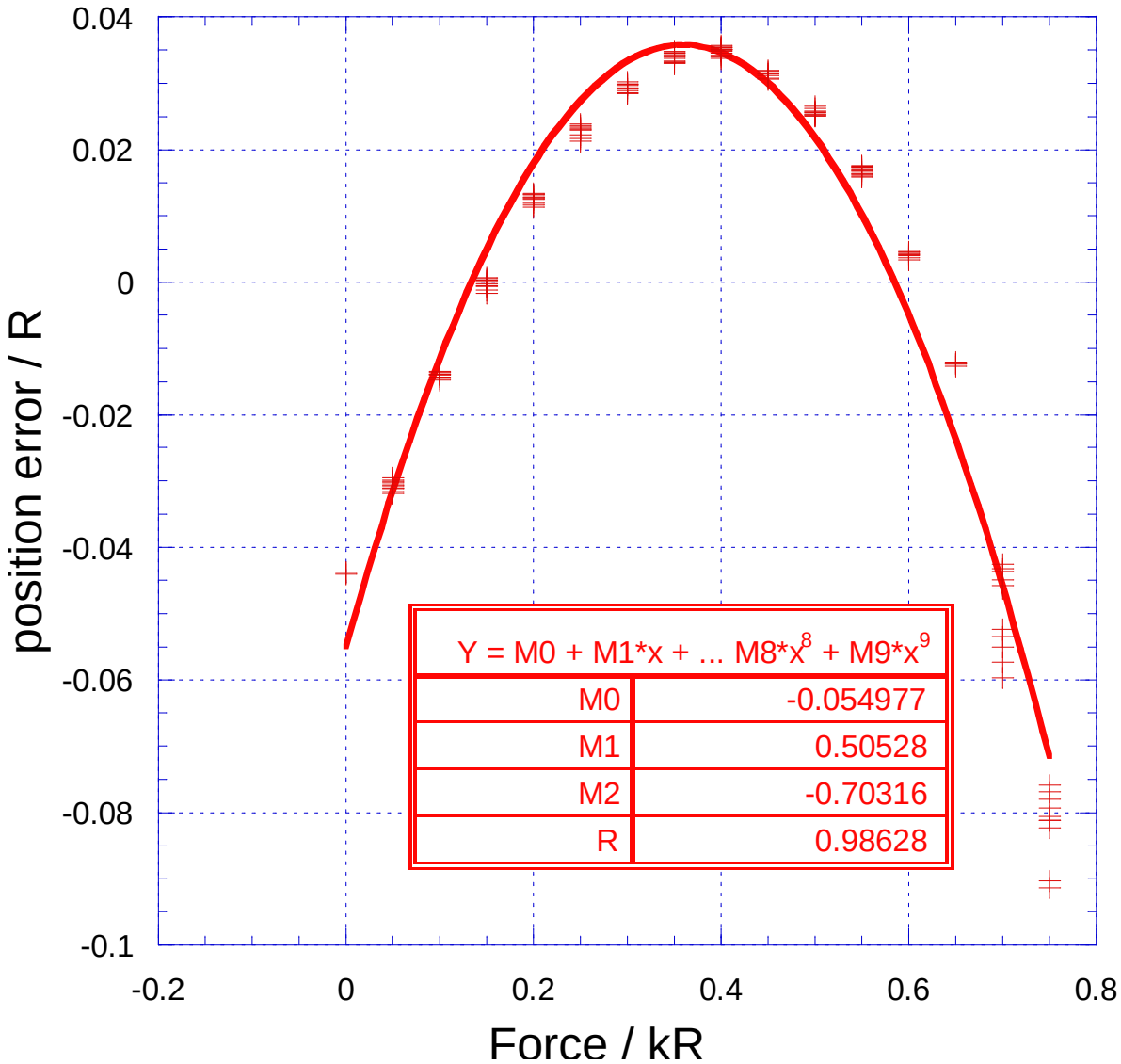


Simulation result

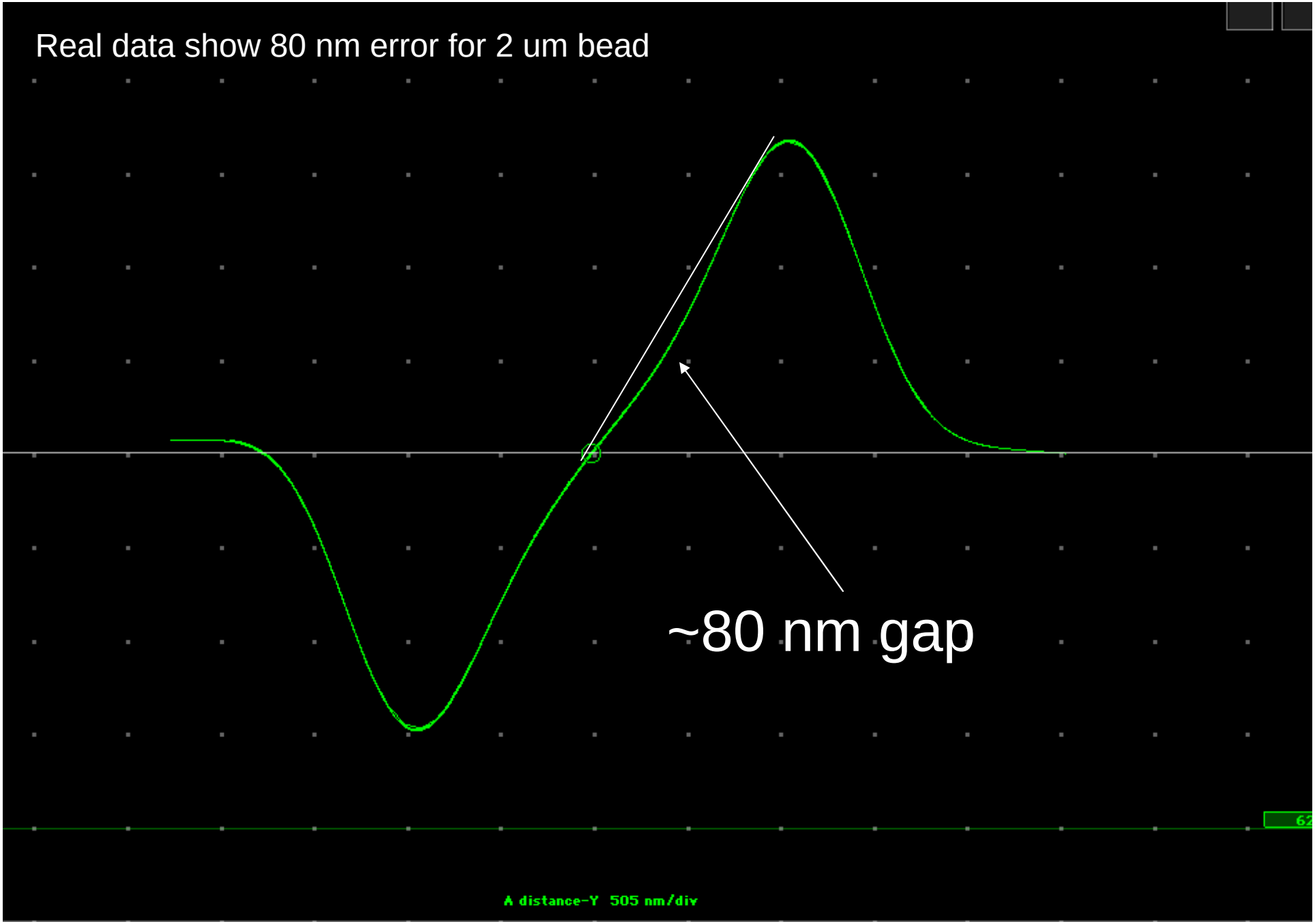


Assuming linear trap stiffness gives error in trap/bead compliance, thus leading to error in end-to-end distance

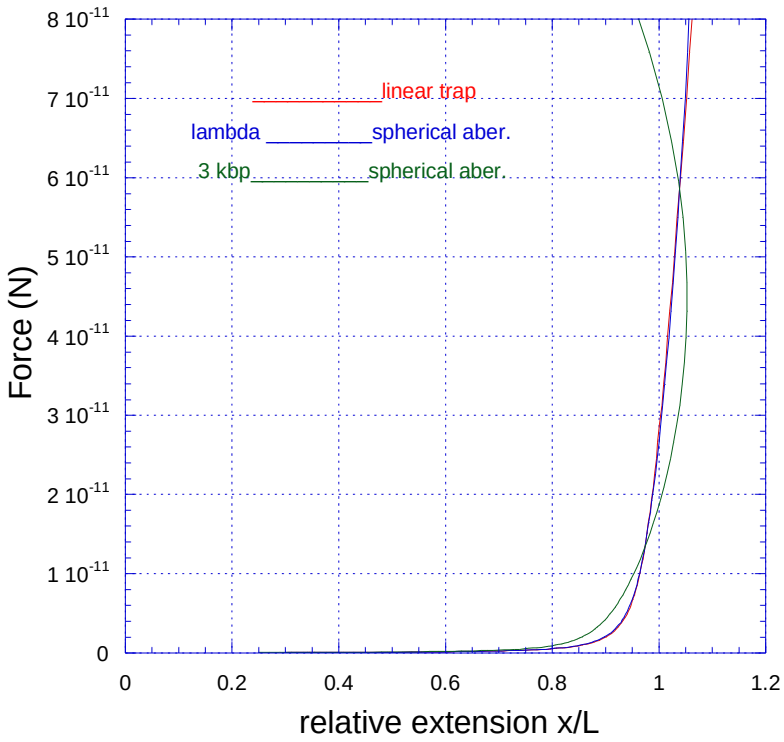
Compliance error plotted as fraction of bead radius



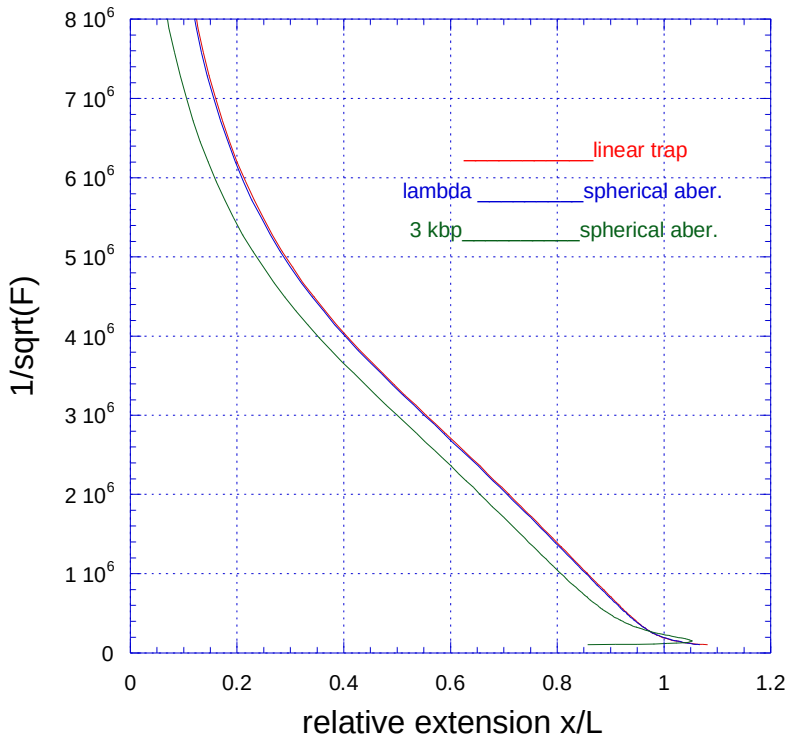
Real data show 80 nm error for 2 um bead



Simulated 80 nm bulge added to force/extension for lambda DNA makes little error. However 80 nm bulge added to curve for 3 kbp molecule makes retrograde slope.



Effect of bulge on 1/sqrt(F) curve

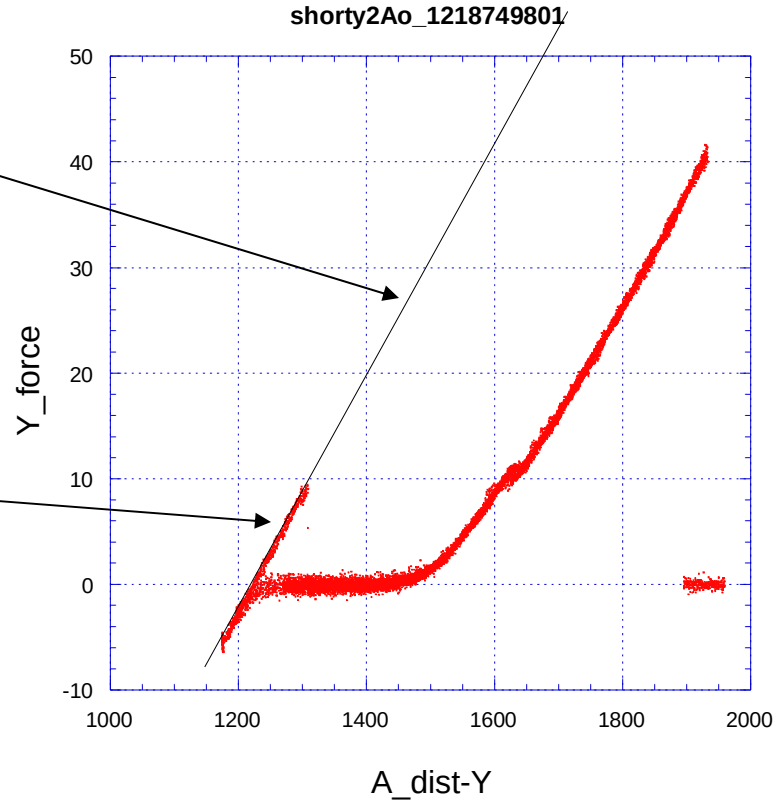


Real 1 kbp dsDNA

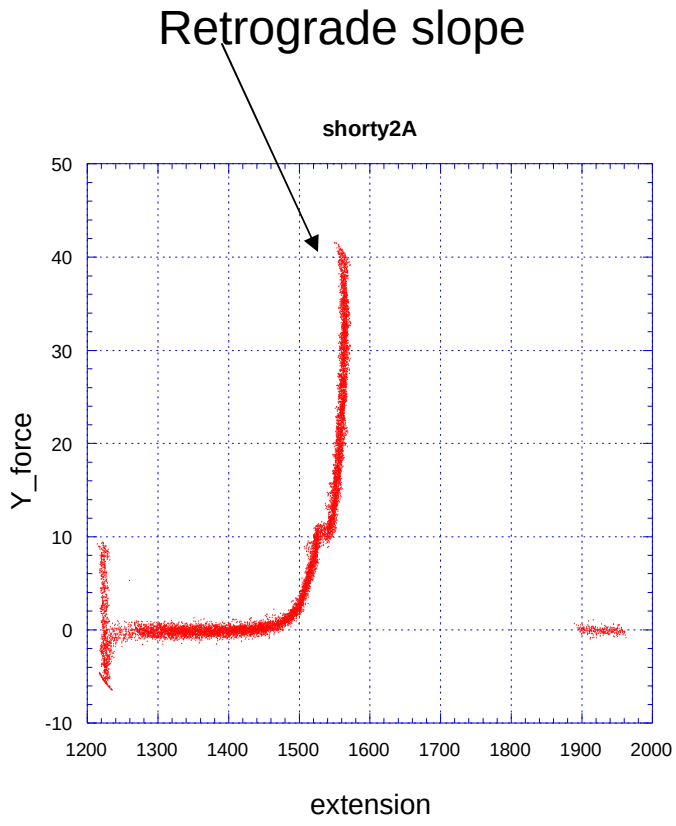
thanks, Rodrigo Maillard for providing molecule

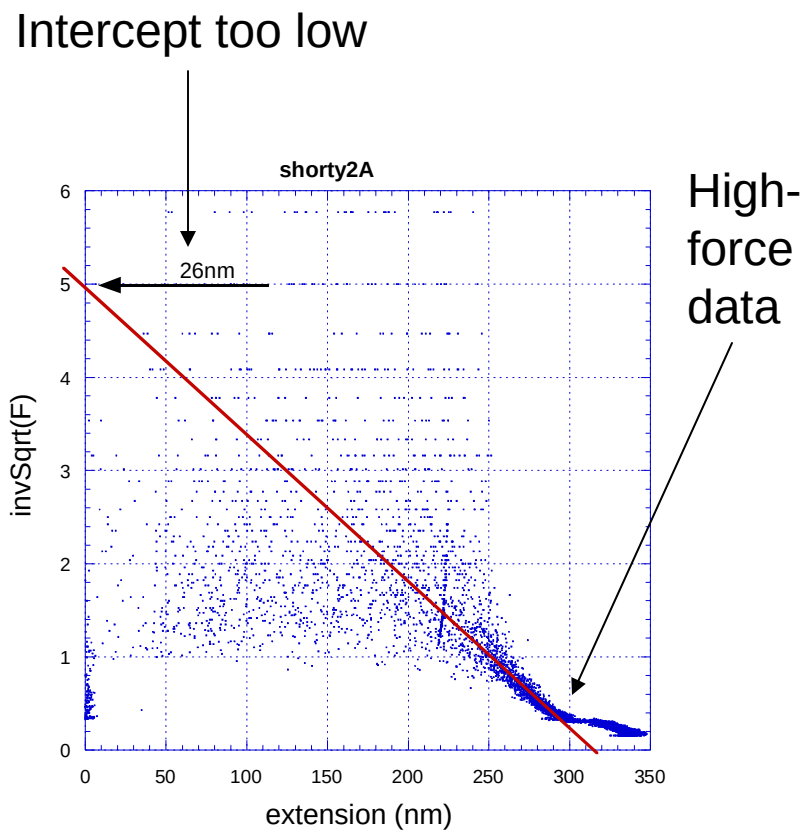
Linear extrapolation of stiffness

Beads were stuck together here

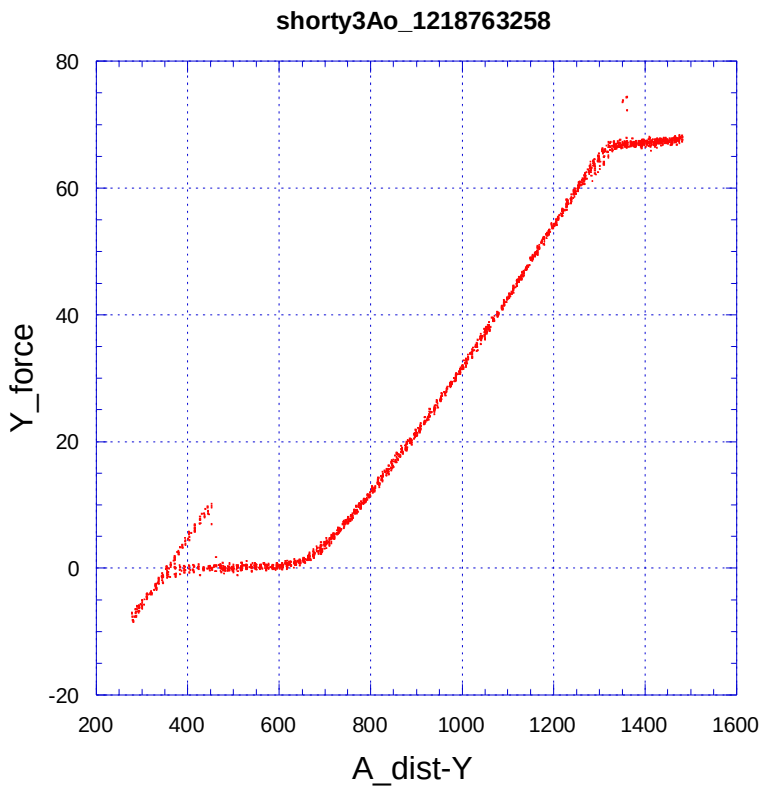


using linear stiffness makes force/extension curve double back at high forces

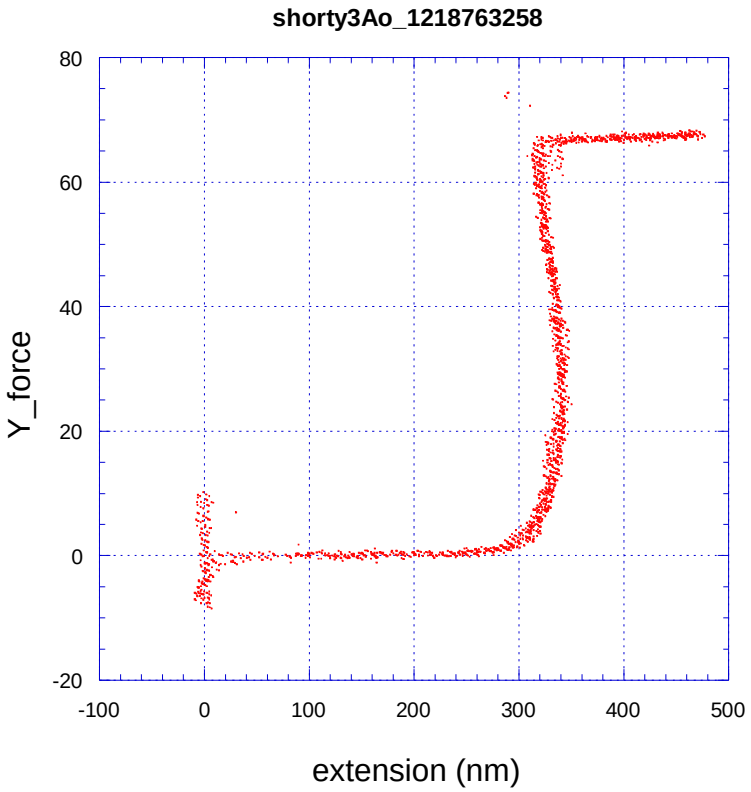


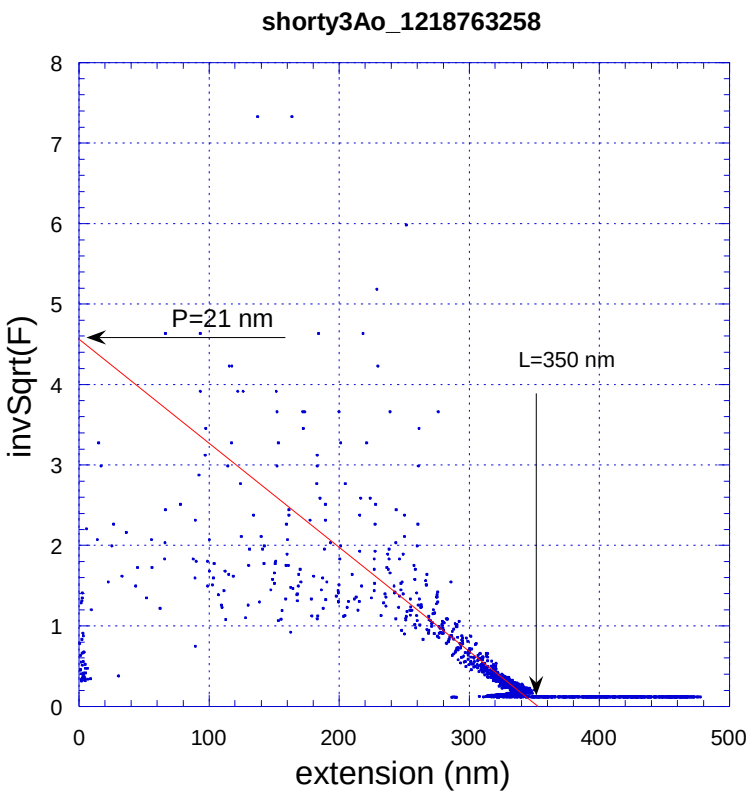


1 kbp pulled to overstretch

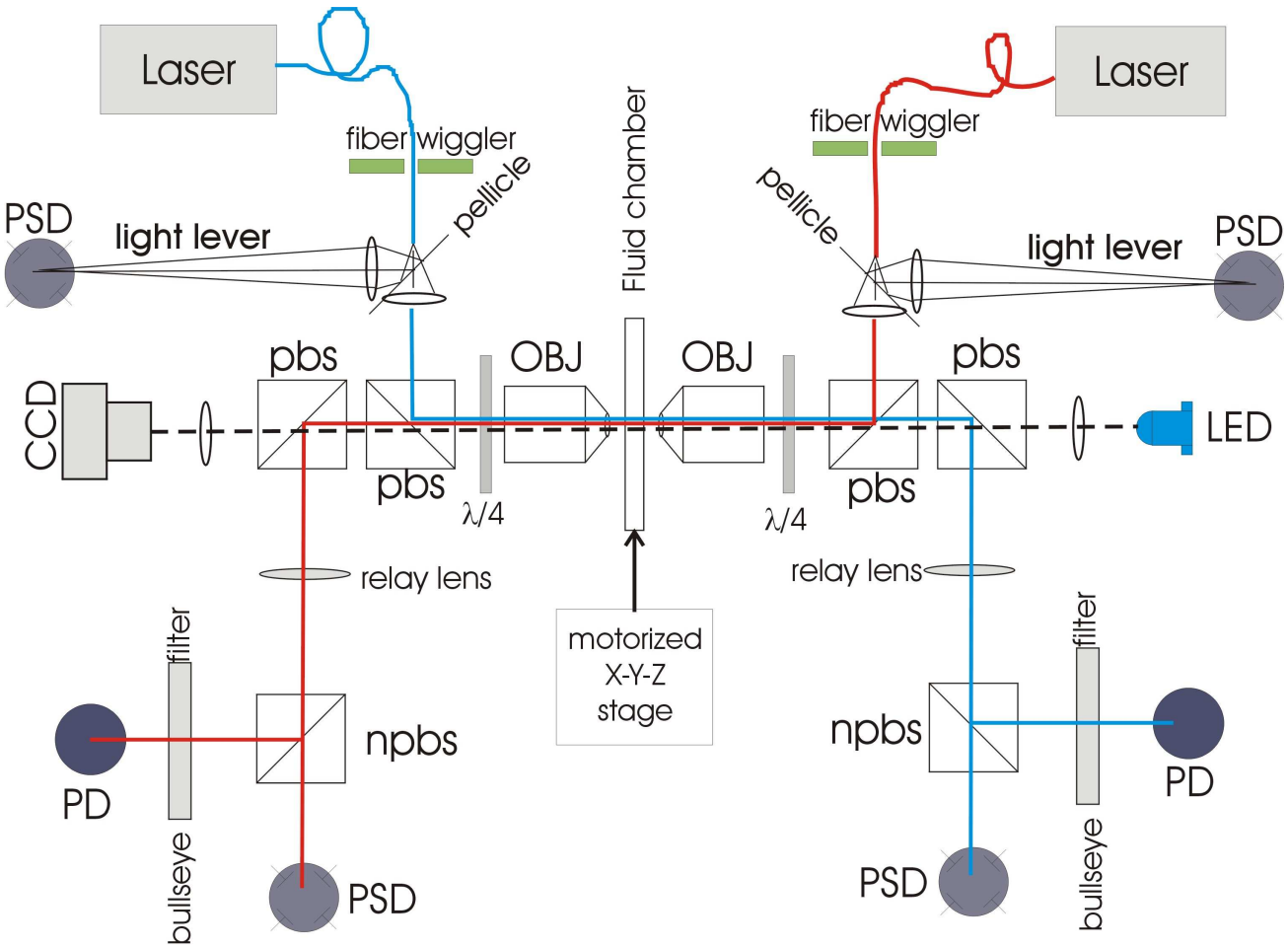


More retrograde slope at higher forces





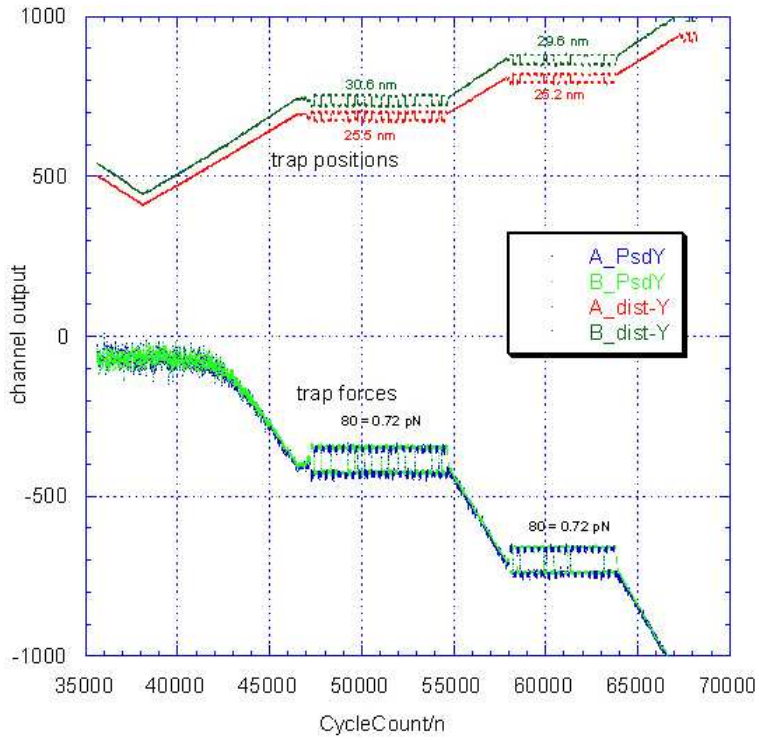
A new method to measure optical trap stiffness with dual counter-propagating beams

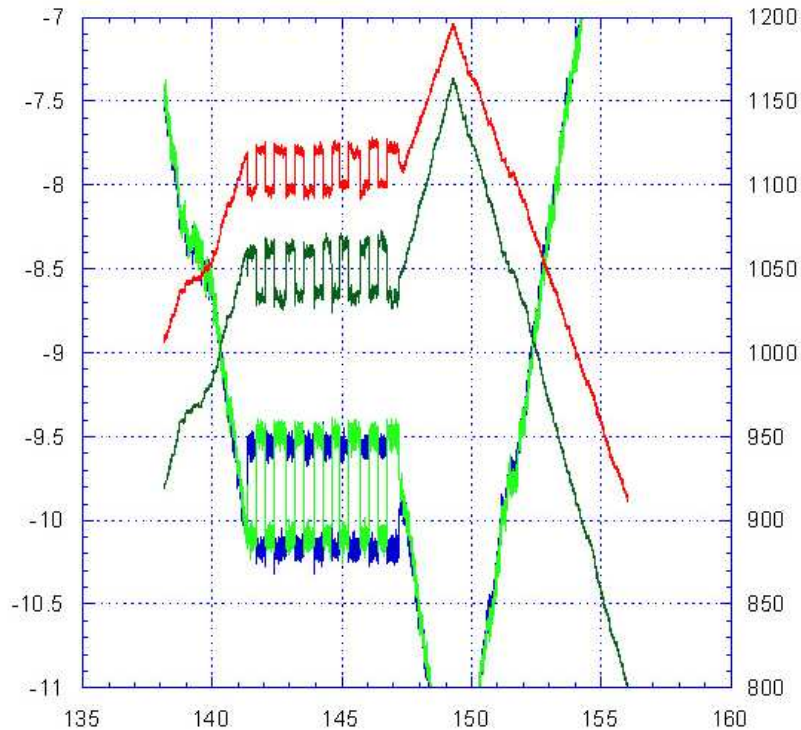


const-force mode with force hops

```
void testStiffness (void* dataRef)          // causes alternating constant-force offsets
that alternates between traps
{  pData data = (pData) dataRef;
     static int count = 0, flip = 0;
     const int forceJumpA = round (1/ data->instrument.trapAForceSensitivity.y);     // picoNewstons
     const int forceJumpB = round (1/ data->instrument.trapBForceSensitivity.y);     // picoNewstons

     if (count++ > 9)
     {
          data->instrument.stiffTestHighA = flip;  // signal to Status columnn data file
          if (flip++)
               {
               CommandTrapAB (0, 0,
                                        psdTargetX,  data->instrument.trapATargetX,
                                        psdTargetY,  data->instrument.trapATargetY + forceJumpA,
                                 0, 0,
                                        psdTargetX,  data->instrument.trapBTargetX,
                                        psdTargetY,  data->instrument.trapBTargetY - forceJumpB,
                                    data);
               flip = 0;
               }
          else
               CommandTrapAB (0, 0,
                                        psdTargetX,  data->instrument.trapATargetX,
                                        psdTargetY,  data->instrument.trapATargetY - forceJumpA,
                                    0, 0,
                                        psdTargetX,  data->instrument.trapBTargetX,
                                        psdTargetY,  data->instrument.trapBTargetY + forceJumpB,
                                    data);
          count = 0;
     }
}
```

Alternate force jumps (+/- 1 pN) sum to the same tension. How far do the individual traps move?





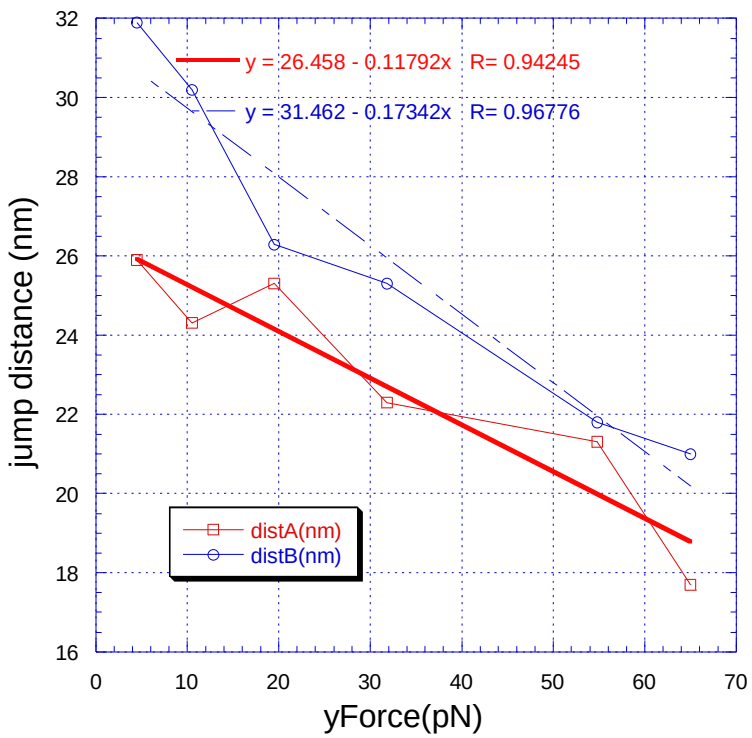
Stiffness Analyzer program

at http://tweezerslab.unipr.it

```
program StiffnessTest;

{   Reads stiffness test file and averages distance jumps . To use this program, the data columns must include CycleCount/n, Y_force,
A_dist-Y, B_dist-Y, and Status.  Further, status reporting must be modified in the Host ltposix.c to read:
data->quicktime.recent[recentIndex][kChannelTweezStatus] =
                                (float) (data->instrument.runMode     * 1000
                                + data->quicktime.byteFromTrapA[mode] * 100
                                + data->quicktime.byteFromTrapB[mode] * 10
                                + data->instrument.stiffTestHighA     * 2
                                + data->instrument.movingUpward       * 1  );
Also the new variable data->instrument.stiffTestHighA  must be defined in ltdata.h and set/reset in the TestStiffness function of ltcontrol.c
by stating    data->instrument.stiffTestHighA = flip  }

uses crt;
const riseTime = 5; {number of lines before Hi/Lo pulse stabilizes}
var  i,j,k,row,L,column, Status,ii, jj,
    CycleCntCol,YforceCol,AdistYCol,BdistYCol,StatusCol:integer;
    fileName,dataLine,dataWord:string;
    inFile, outFile:text;
    APsdY, APsdSum,BPsdY,BPsdSum,Yforce,AdistY,BdistY,CycleCnt,CycleCntSum,
    YforceSum,AdistYSum,BdistYSum: real;
    pastStatus:array[1..riseTime] of integer;
    pulseAvg:array[0..100,0..5] of real;
            {CycleCnt,YforceAv, AdistAv, BdistAv, status, k}

procedure openInputFile;
begin
clrscr;
```

Stiffness Tester output

Cycle	Force	distA	distB	HiPulses	LoPulses
1258708	64.2	13.9	28.5	19	20
1285434	4.5	25.9	31.9	4	4
1292750	10.5	24.3	30.2	18	20
1305064	19.5	25.3	26.3	35	34
1319831	31.8	22.3	25.3	21	23
1337793	54.8	21.3	21.8	25	23
1351266	65.0	17.7	21.0	14	16



Jump distance for 2 pN change as function of average force

Treating the test data

- (1) Normalize jump distances to ΔF=1 pN
- (2) Make linear fits to distance vs. force/2 such that

  $$distA = D_A - M_A * (yForce/2)\ ,$$

  $$distB = D_B - M_B * (yForce/2)$$
- (3) Integrate compliance to get offset with force.
- (4) Subtract from trap position to get molecular extension.

  $$extensionA = AdistY - D_A(yForce/2) + (M_A/2)\ (yForce/2)^2$$

  $$extensionB = BdistY - D_B(yForce/2) + (M_B/2)\ (yForce/2)^2$$

  (5) Should give 2 force-extension curves that are the same.

Work in progress

To try:

- Pull more short molecules
- Vary bead sizes
- Try beads stuck on pipette

The end

- Thanks to Rodrigo, Wei, and Carlos