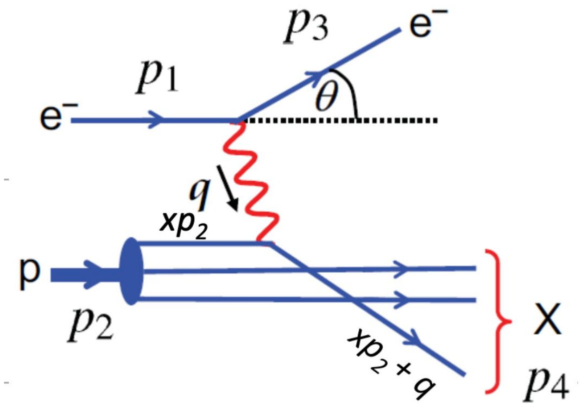
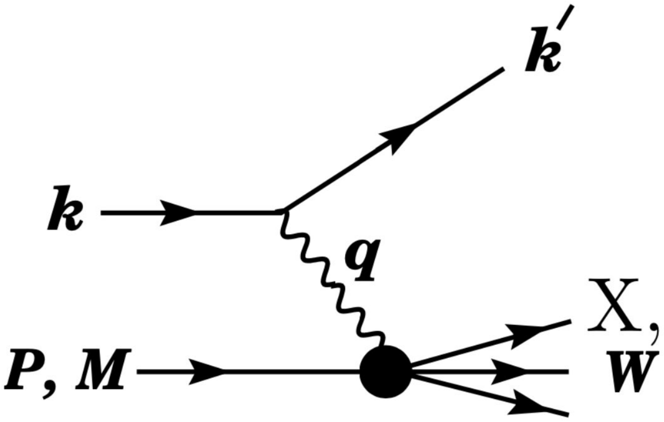
Parity-Violating Asymmetry

Saajid Chowdhury

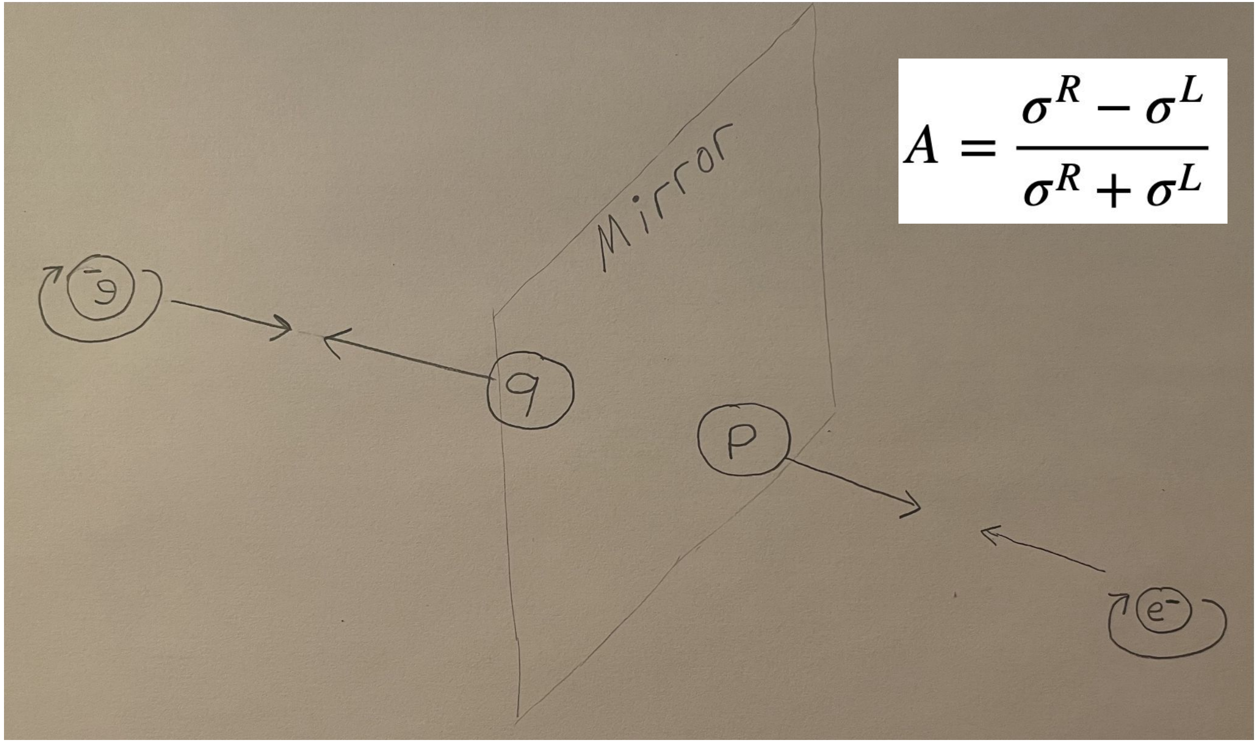
Definitions

- x = fraction of proton's total momentum carried by quark/gluon struck by electron
- Q^2 = squared momentum transferred to electron
- Differential cross section = $\frac{d^2\sigma}{dx\,dQ^2}$ as a function of certain x, Q^2; probability that event has kinematics within dx, dQ^2 of these
- Spin polarization = fraction (component) of electron spin directed towards its momentum, ranges from +1 to -1
- Structure functions = e.g. $F_2(x, Q^2)$, factors in cross section formulas, sensitive to quark distributions (over x and Q^2) in proton





Physics



Goals

- Simulate events with polarized electrons (helicity +1, -1) to see if EIC can measure asymmetries
    - To use asymmetries more precisely calculate structure functions
- Compare asymmetries for each x, Q^2 from simulation to theory

Work so far

- Ran 10 million events each for electron spin polarizations +1, -1
- Plotted asymmetry vs. x for Q^2 = 2.0, 5.1, 8.2, 12.9, 20.5, 51.5, 129.2, 514.5 GeV^2
- Compared to theory

Math

Givens:

$$\frac{d^2\sigma}{dx\,dy} = \frac{4\pi\alpha^2}{xyQ^2}\left(\left(1-y+\frac{y^2}{2}\right)F_2^{NC} + \left(y-\frac{y^2}{2}\right)xF_3^{NC}\right)$$

$$F_2^{NC} = F_2^{\gamma} - (g_V^e - \lambda g_A^e)\eta_{\gamma Z}F_2^{\gamma Z}$$

$$xF_3^{NC} = -(g_A^e - \lambda g_V^e)\eta_{\gamma Z}xF_3^{\gamma Z}$$

(Tanabashi et al.)

Some approximations:

$$Q^2 \gg M \rightarrow x^2y^2M^2/Q^2 \approx 0$$

$$\text{neglect } (g_V^e)^2, (g_A^e)^2$$

$$2F_1^{\gamma}x = F_2^{\gamma} \text{ (Callan-Gross)}$$

Goal: express in terms of structure functions:

$$A = \frac{\sigma^R - \sigma^L}{\sigma^R + \sigma^L}$$

More math

$$\sigma_r = \frac{1}{1 - y + y^2/2} \frac{x y Q^2}{4\pi\alpha^2} \frac{d^2\sigma}{dx\,dy}$$

$$\sigma_r = F_2^{NC} + \frac{y - y^2/2}{1 - y + y^2/2} x F_3^{NC}$$

$$Y_\pm = 1 \pm (1 - y)^2$$

$$\sigma_r = F_2^{NC} + \frac{Y_-}{Y_+} x F_3^{NC}$$

$$A = \frac{\sigma_r^R - \sigma_r^L}{\sigma_r^R + \sigma_r^L}$$

$$A = \frac{g_A^e \eta_{\gamma Z} x F_1^{\gamma Z} + \frac{Y_-}{2Y_+}\left(g_V^e \eta_{\gamma Z} x F_3^{\gamma Z}\right)}{x F_1^{\gamma} - g_V^e \eta_{\gamma Z} x F_1^{\gamma Z} - \frac{Y_-}{2Y_+}\left(g_A^e \eta_{\gamma Z} x F_3^{\gamma Z}\right)}$$

neglect $-g_V^e \eta_{\gamma Z} x F_1^{\gamma Z}$ and $-\frac{Y_-}{2Y_+}\left(g_A^e \eta_{\gamma Z} x F_3^{\gamma Z}\right)$

$$A = \eta_{\gamma Z}\left(g_A^e \frac{F_1^{\gamma Z}}{F_1^{\gamma}} + \frac{Y_-}{2Y_+} g_V^e \frac{F_3^{\gamma Z}}{F_1^{\gamma}}\right)$$

$$\eta_{\gamma Z} = \frac{G_F Q^2}{2\sqrt{2}\pi\alpha}$$

Asymmetry Formula

$$A = \frac{G_F Q^2}{2\sqrt{2}\pi\alpha}\left(g_A^e \frac{F_1^{\gamma Z}}{F_1^\gamma} + g_V^e \frac{Y_-}{2Y_+} \frac{F_3^{\gamma Z}}{F_1^\gamma} \right)$$

(Y. X. Zhao et al.)

Executables for running events

```
[saajidchowdhury@eic0101 djangoh]$ cat ep.Rad=0.NC.Apve.in
OUTFILENAM
outfiles/djangoh.NC.Apve.noRad.20x250
TITLE
DJANGOH 4.6.10 for eRHIC for PR, LO at 20x250, Wmin=1.4
EL-BEAM
            20D0    -1.0D0    -1
PR-BEAM
            250D0   0.0D0

RNDM-SEEDS
            1    1

LHAPATH
/gpfs/mnt/gpfs01/eic/data/LHAPDF59SHARE/lhapdf/PDFsets

START
            10000000
```

```
[saajidchowdhury@eic0101 djangoh]$ cat run_ep_Apve_norad.sh
#!/usr/bin/bash

echo "-----------------------------------"
echo "Running DJANGOH Simulation for ep Collider!!!"
echo "..."
echo ""


OUTFILE1=outfiles/djangoh.NC.Apve.noRad.20x250_evt.dat
if test -f "$OUTFILE1"; then
        rm -f "$OUTFILE1"
fi
OUTFILE2=outfiles/djangoh.NC.Apve.noRad.20x250_out.dat
if test -f "$OUTFILE2"; then
        rm -f "$OUTFILE2"
fi

#Create file for random number generation
./make_random.py

djangoh < ep.Rad=0.NC.Apve.in > logfiles/ep.Rad=0.NC.Apve.log

echo "Completed Simulation!!!"
echo ""

echo "Making Output ROOT File..."
root -l -b -q 'make_tree.C("djangoh.NC.Apve.noRad.20x250_evt.dat")'
echo "Done!!!"
```

Condor: computing cluster

```
[[saajidchowdhury@eic0101 djangoh]$ cat cond.sub
Universe = vanilla
Notification = Never
Executable = run_ep_Apve_norad.sh
GetEnv = True
Input = /dev/null
Output = jobout/cond.out
Error = jobout/cond.err
Log = jobout/cond.log
Queue
```

Python: Calculate theory

- Hardcode constants
- For each PDF set:
    - For selected Q^2 values, for several x values:
        - Get structure function values from PDF set
        - Calculate asymmetry
        - Output Q^2, x, asymmetry to file

```
A = GF*q2/(2*math.sqrt(2)*pi*a) *
(gAe * f2gz/f2g +
gVe*Ym*2*x/(2*Yp) * f3gz/f2g)
```

Reaction	Structure Functions	Index
$e^{\pm} + T \to e^{\pm} + X$	$F_2^{\gamma},\ F_L^{\gamma}$	900, 901
	$F_2^{\gamma Z},\ F_L^{\gamma Z},\ F_3^{\gamma Z}$	902, 903, 904

https://jeffersonlab.github.io/txgrids/_build/html/grids.html

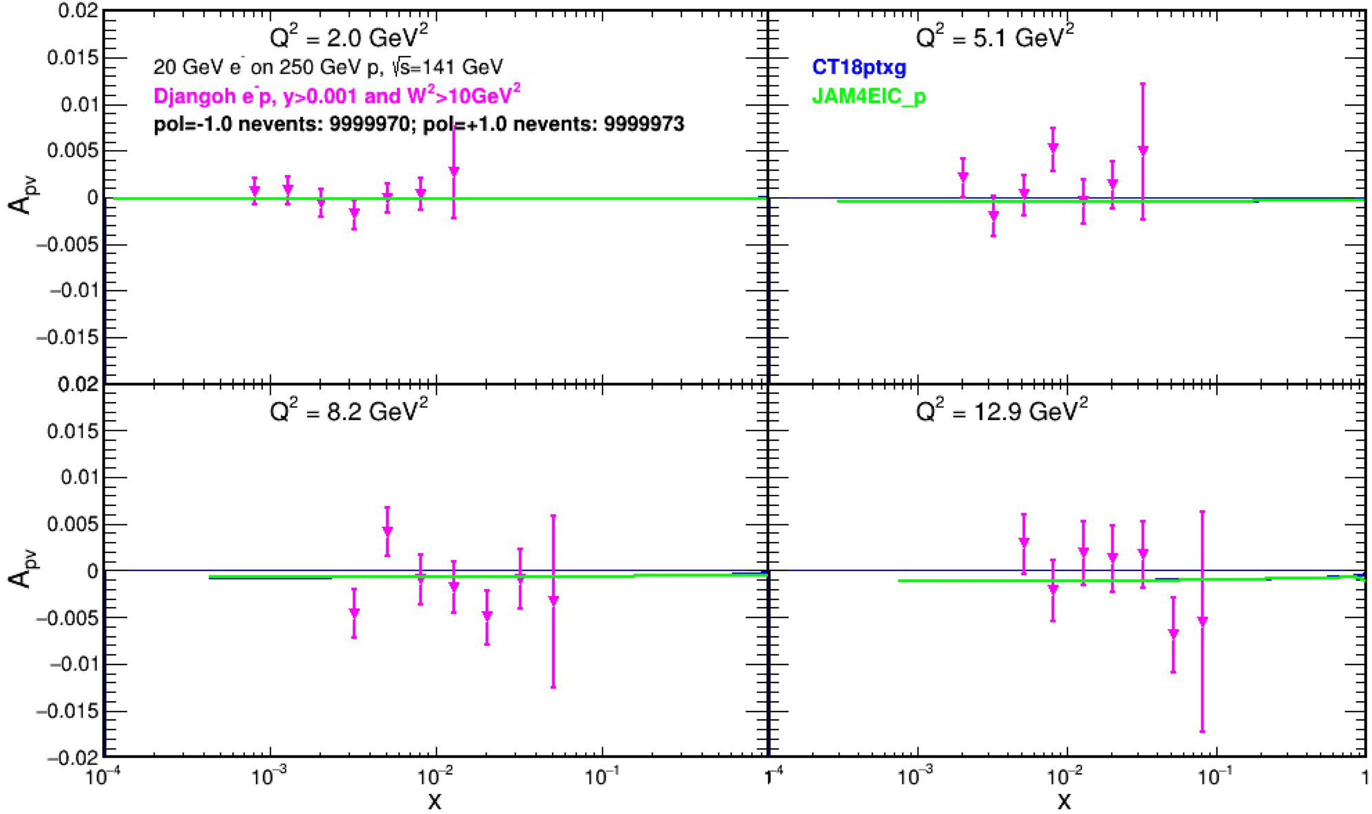
ROOT: Plot asymmetry

- Theory graph method:
  - given file containing theoretical asymmetries from given PDFset, make TGraph

- Main method:
  - Hardcode constants
  - Set up bins x, Q^2
  - Calculate bin yields for polarizations +1,-1
  - Calculate bin asymmetries
  - Plot A(x) for selected Q^2 values

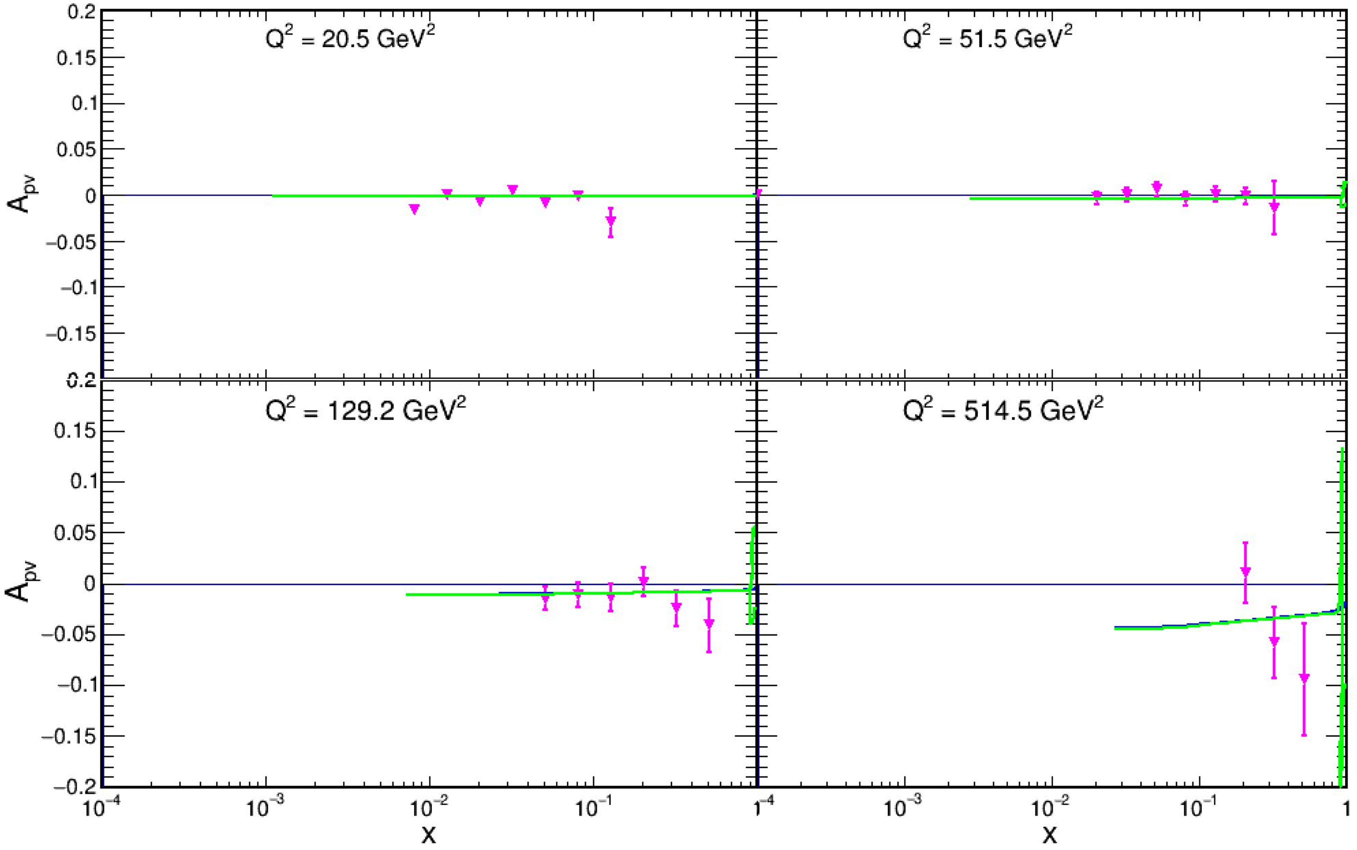
```
//For each x,Q^2 bin:

n1 = t1->GetEntries()
n2 = t2->GetEntries()
L1 = n1 / crosstot1
L2 = n2 / crosstot2
A[i][j] = (L2 * N1 - L1 * N2) /
(L2 * N1 + L1 * N2);
```

Plots: Q^2 = 2.0, 5.1, 8.2, 12.9



CT18ptxg = NNLO
JAM4EIC = NLO

Plots: Q^2 = 20.5, 51.5, 129.2, 514.5



CT18ptxg = NNLO
JAM4EIC = NLO

Plans for Future

- Use Condor Queue command to run 100 million events by running in parallel 10 instances of 10 million events each
  - Using Dan’s and Barak’s python script to generate random seeds for Djangoh to prevent duplicate random numbers
- Account for detector effects (e.g. radiation)
- Understand theory more
- Make code more efficient and concise
- AI
  - applied to particle identification detectors, Experimental Design and Simulations, Reconstruction/Analysis, Control of Experimental Systems, Detector Readout, and Computing Frontiers

Citations

M. Tanabashi *et al.* (Particle Data Group), Phys. Rev. D **98**, 030001 (2018) and 2019 update 6th December, 2019 11:50am, page 3

Y. X. Zhao *et al.*: Neutral Weak Interactions at an EIC, page 2

LHAPDF grids (txgrids) from https://jeffersonlab.github.io/txgrids/_build/html/grids.html

Barak