Some fitting and extraction experiences and status of the GeParD code

Krešimir Kumerički

University of Zagreb, Croatia

Prospects for extraction of GPDs from global fits of current and future data

22-25 January, 2019, Heavy Ion Laboratory (Cyklotron), Warsaw

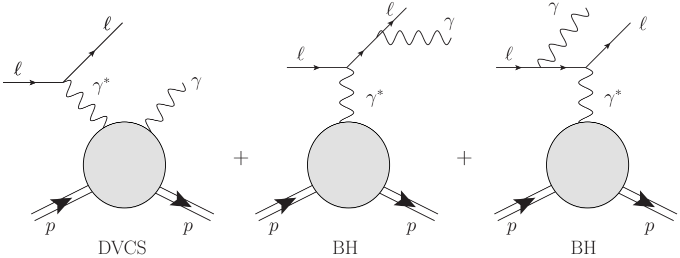
EUROPSKI STRUKTURNI I INVESTICIJSKI FONDOVI

Outline

1. Introduction and status of fits to DVCS data
2. GeParD
3. Neural net fits
4. Summary

Access to GPDs via DVCS

- Deeply virtual Compton scattering (DVCS) — "gold plated" process of exclusive physics
- DVCS is measured via leptoproduction of a photon



- Interference with Bethe-Heitler process gives unique access to both real and imaginary part of DVCS amplitude.

DVCS cross section

[Belitsky & Müller]

$$d\sigma \propto |\mathcal{T}|^2 = |\mathcal{T}_{\rm BH}|^2 + |\mathcal{T}_{\rm DVCS}|^2 + \mathcal{I} \,.$$

- where *e. g.* interference term is

$$\mathcal{I} \quad \propto \quad \frac{-e_\ell}{\mathcal{P}_1(\phi)\mathcal{P}_2(\phi)} \left\{ c_0^{\mathcal{I}} + \sum_{n=1}^{3} \left[c_n^{\mathcal{I}} \cos(n\phi) + s_n^{\mathcal{I}} \sin(n\phi) \right] \right\},$$

- where *e. g.* $c_1^{\mathcal{I}}$ harmonic for unpolarized target is

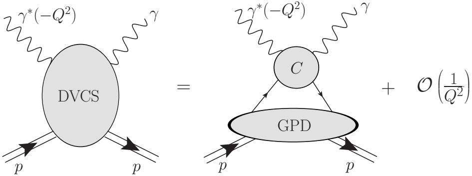
$$c_{1,\rm unpol.}^{\mathcal{I}} \propto \left[F_1 \,\mathfrak{Re}\,\mathcal{H} - \frac{t}{4M_p^2} F_2 \,\mathfrak{Re}\,\mathcal{E} + \frac{x_{\rm B}}{2-x_{\rm B}} (F_1 + F_2) \,\mathfrak{Re}\,\widetilde{\mathcal{H}} \right]$$

- and at leading order everything depends on four complex

Compton form factors (CFFs)

$$\mathcal{H}(\xi, t, Q^2), \quad \mathcal{E}(\xi, t, Q^2), \quad \widetilde{\mathcal{H}}(\xi, t, Q^2), \quad \widetilde{\mathcal{E}}(\xi, t, Q^2)$$

Factorization of DVCS $\longrightarrow$ GPDs

- [Collins et al. '98] 

- CFFs are convolution:

$$^{a}\mathcal{H}(\xi, t, Q^2) = \int \mathrm{d}x \; C^a(x, \xi, \frac{Q^2}{Q_0^2}) \; H^a(x, \xi, t, Q_0^2)$$

$a{=}q,G$

- $H^a(x, \eta, t, Q_0^2)$ — Generalized parton distribution (GPD)

Modelling GPDs in moment space

- Instead of considering momentum fraction dependence $H(x, \ldots)$
- ... it is convenient to make a transform into complementary space of conformal moments j:

$$H_j^q(\eta, \ldots) \equiv \frac{\Gamma(3/2)\Gamma(j+1)}{2^{j+1}\Gamma(j+3/2)} \int_{-1}^{1} \mathrm{d}x \, \eta^j \, C_j^{3/2}(x/\eta) \, H^q(x, \eta, \ldots)$$

- They are analogous to Mellin moments in DIS: $x^j \to C_j^{3/2}(x)$
- $C_j^{3/2}(x)$ — Gegenbauer polynomials

CFFs as Mellin-Barnes integral

$$\mathcal{H}(\xi, t, Q^2) = \frac{1}{2i} \int_{c-i\infty}^{c+i\infty} dj\, \xi^{-j-1} \left[i + \tan\left(\frac{\pi j}{2} \right) \right]$$

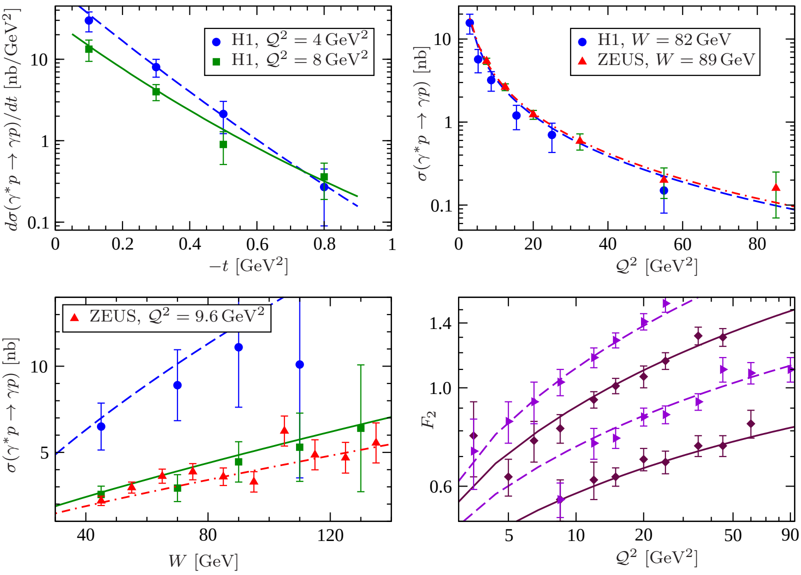
$$\times \; C_j(Q^2/\mu^2, \alpha_s(\mu))\; H_j(\xi, t, \mu^2)\,. \qquad (1)$$

- Evolution of GPDs:

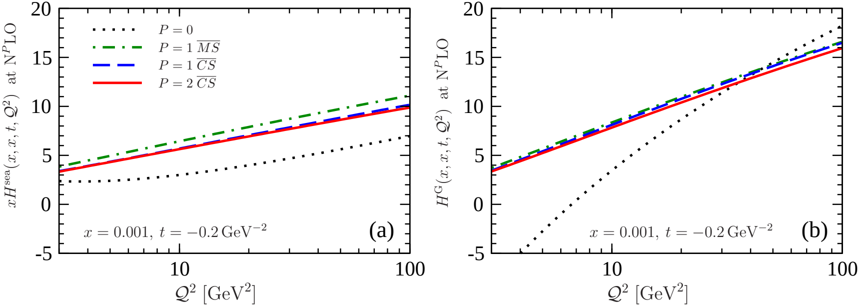
$$H_j(\eta, t, \mu) = \sum_k E_{jk}(\mu, \mu_0; \eta)\; H_k(\eta, t, \mu_0)\,, \qquad (2)$$

- C_j and E_{jk} known to (N)NLO
- TODO: It would be nice to have inversion formula giving $H(x, \eta, t)$ from $H_j(\eta, t)$. Special cases $H(x, 0, t)$ and $H(x, x, t)$ are known.

Example of fit result — HERA collider data



Resulting small-x $H(x, x, t)$



- P=0: LO; P=1: NLO; P=2: NNLO
- The whole procedure is extended to meson production [Müller, Lautenschlager, Passek-Kumerički, Schäfer '13] (vectors), [Duplančić, Müller, Passek-Kumerički '18] (pseudoscalars)

Global fit χ^2 values: KM and PARTONS

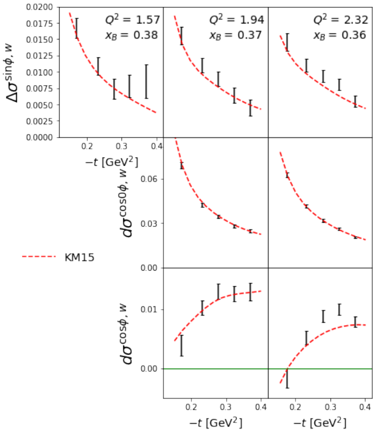
[K.K., Müller '09-'15]

Collaboration	Observable	Ref.	n_{pts}	KMM12		KM15	
				χ^2/n_{pts}	pull	χ^2/n_{pts}	pull
ZEUS	σ_{DVCS}	[19, 20]	11	0.49	-1.76	0.51	-1.74
ZEUS,H1	$d\sigma_{\text{DVCS}}/dt$	[19, 21, 22]	24	0.97	0.85	1.04	1.37
HERMES	$A_{\text{C}}^{\cos 0\phi}$	[23]	6	1.31	0.49	1.24	0.29
HERMES	$A_{\text{C}}^{\cos\phi}$	[23]	6	0.24	-0.56	0.07	-0.20
HERMES	$A_{\text{LU,I}}^{\sin\phi}$	[23]	6	2.08	-2.52	1.34	-1.28
CLAS	$A_{\text{LU}}^{\sin\phi}$	[24]	4	1.28	2.09		
CLAS	$A_{\text{LU}}^{\sin\phi}$	[4, 25]	13			1.24	0.63
CLAS	$\Delta\sigma^{\sin\phi,w}$	[7]	48			0.41	-1.66
CLAS	$d\sigma^{\cos 0\phi,w}$	[7]	48			0.16	-0.21
CLAS	$d\sigma^{\cos\phi,w}$	[7]	48			1.16	6.36
Hall A	$\Delta\sigma^{\sin\phi,w}$	[5]	12	1.06	-2.55		
Hall A	$d\sigma^{\cos 0\phi,w}$	[5]	4	1.21	2.14		
Hall A	$d\sigma^{\cos\phi,w}$	[5]	4	3.49	-0.26		
Hall A	$\Delta\sigma^{\sin\phi,w}$	[6]	15			0.81	-2.84
Hall A	$d\sigma^{\cos 0\phi,w}$	[6]	10			0.40	0.92
Hall A	$d\sigma^{\cos\phi,w}$	[6]	10			2.52	-2.42
HERMES,CLAS	$A_{\text{UL}}^{\sin\phi}$	[18, 26]	10	1.90	-1.89	1.10	-1.94
HERMES	$A_{\text{LL}}^{\cos 0\phi}$	[26]	4	3.44	2.17	3.19	1.99
HERMES	$A_{\text{UT,I}}^{\sin(\phi-\phi_S)\cos\phi}$	[27]	4	0.90	0.61	0.90	0.71
CLAS	$A_{\text{UL}}^{\sin\phi}$	[4]	10			0.76	0.38
CLAS	$A_{\text{LL}}^{\cos 0\phi}$	[4]	10			0.50	-0.22
CLAS	$A_{\text{LL}}^{\cos\phi}$	[4]	10			1.54	2.40

[Moutarde, Sznajder, Wagner, '18]

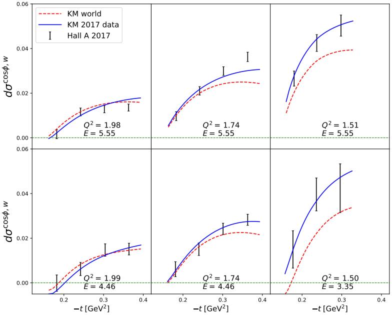
No.	Collab.	Year	Ref.	χ^2	n	χ^2/n
1	HERMES	2001	13	9.8	10	0.98
2		2006	114	2.9	4	0.72
3		2008	115	24.2	18	1.35
4		2009	116	40.1	35	1.15
5		2010	117	40.3	18	2.24
6		2011	118	14.5	24	0.60
7		2012	119	25.4	35	0.73
8	CLAS	2001	14	—	0	—
9		2006	120	0.9	2	0.47
10		2008	121	371.1	283	1.31
11		2009	122	36.4	22	1.66
12		2015	123	351.4	311	1.13
13		2015	124	937.9	1333	0.70
14	Hall A	2015	112	220.2	228	0.97
15		2017	113	258.8	276	0.94
16	COMPASS	2018	55	10.7	1	10.67

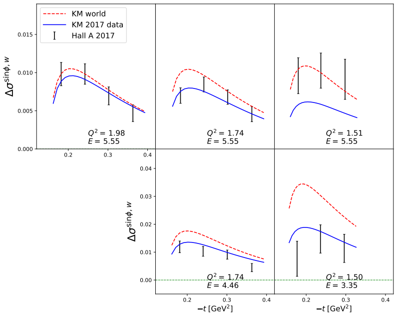
Example fixed target: Hall A (2015)



- KM15 global fit is fine. $\chi^2/n_{\rm d.o.f.} = 240./275$ ✓

Including Hall A 2017 data in global world fit: fail ✗



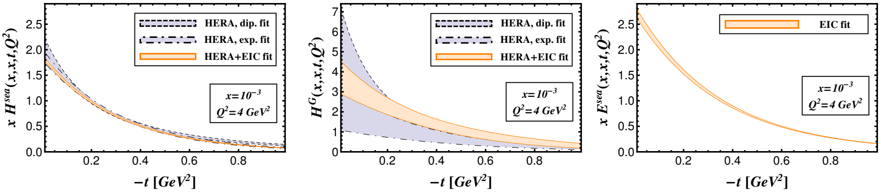


Global world DVCS data fit

before 2017	including 2017 Hall A
$\chi^2/n_{\mathrm{d.o.f}} = 240./275$ ✓	$\chi^2/n_{\mathrm{d.o.f}} = 545./337$ ✗

DVCS at EIC

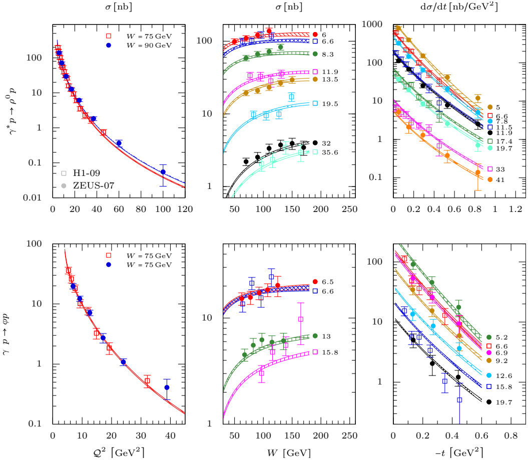
- This framework was used to estimate impact of EIC [Aschenauer, Fazio, K.K., Müller '13], [EIC white paper]
- Fit to simulated DVCS data ($d\sigma^{\rm DVCS}/dt$ and A_{UT}) at 20 GeV $\times$ 250 GeV taking $E_{\rm sea}(x,\eta,t) = \kappa_{\rm sea} H_{\rm sea}(x,\eta,t)$



- Improved knowledge of low-t quark and gluon GPDs H ($\Longrightarrow$ 3D parton imaging)
- Improved knowledge of sea quark GPD E
- TODO: Do the same for DVMP

NLO DIS+DVCS+DVMP small-x global fit

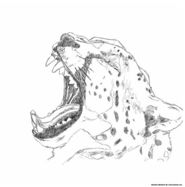
First global fits to DIS+DVCS+DVMP HERA collider data [Lautenschlager, Müller, Schäfer '13] show promise and prefer NLO:



(Here one needs relaxed model with different Regge trajectories for each SO(3) partial wave of conformal space GPD.)

GeParD software

GeParD* software — history, # of lines of code



Language	files	blank	comment	code
Python	46	2289	2865	13061
Fortran 77	120	7822	30240	12190
IPython Notebook	2	0	0	1956
C	5	161	263	454
Mathematica	1	124	15	415
make	4	101	133	296
C/C++ Header	2	40	74	91
DOS Batch	1	0	0	44
Markdown	1	3	0	5
SUM:	182	10540	33590	28512

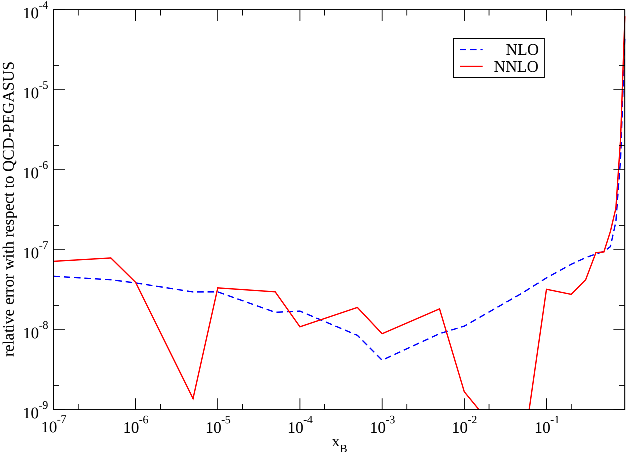
* Not to be confused with biology software (`GEnome PAir Rapid Dotter`)

Fortran 77 part

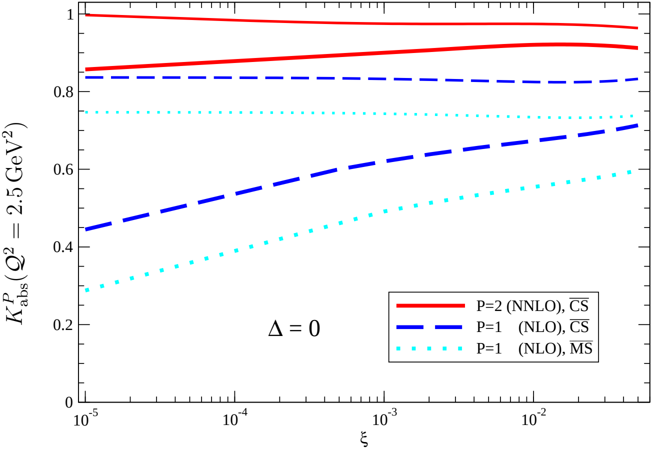
- What is implemented:
  - conformal space evolution of GPDs: up to NLO ($\overline{\rm MS}$ scheme) and partialy to NNLO (conformal scheme)
  - unpol. DIS hard-scattering coefficient functions to NNLO
  - unpol. DVCS hard-scattering coefficient functions to NNLO
  - unpol. DVMP hard-scattering coefficient functions to NLO (not thoroughly tested)
  - Compton Form Factors $\mathcal{H}$ and $\mathcal{E}$
  - DVMP transition form factor ($\mathcal{H}^{\rho^0 p}$)
  - small-x $\sigma^{\rm DVCS}$ and $\sigma^{DV\rho P}$
- Features:
  - fast and accurate
  - code thoroughly documented
  - Mathematica and Python interfaces

F77 part, study 1: evolution check

- Comparison to QCD-Pegasus PDF evolution software [A. Vogt '04]



F77 part, study 2: (N)NLO corrections



Thick lines:
"hard" gluon
$N_G = 0.4$
$\alpha_G(0)$=$\alpha_\Sigma(0)$
$+ 0.05$

Thin lines:
"soft" gluon
$N_G = 0.3$
$\alpha_G(0)$=$\alpha_\Sigma(0)$
$- 0.02$

$$K^P_{\rm abs} \equiv \left| \frac{\mathcal{H}^{(P)}}{\mathcal{H}^{(P-1)}} \right|$$

GeParD — interactive example 1/12

- Python part of GeParD implements complete [Belitsky, Müller et al.] DVCS formulas, all measured DVCS observables, and enables interactive work. An illustration follows:

```
In [12]: m = Model.ComptonGepard(p=1, q02=2.)      # initialize NLO KM model for CFFs
         th = Approach.BMK(m)                      # use BMK formulas for DVCS
         th.name = 'NLO preliminary'               # name to be used on plots

In [13]: th.m.parameters['AL0S'] = 1.2             # change some model parameter

In [14]: utils.describe_data(DISpoints)            # DISpoints = data set

         npt x obs    collab  FTn    id  ref.
         --------------------------------------------
          8 x F2     H1      N/A    201 Nucl.Phys.B470(96)3
          8 x F2     H1      N/A    202 Nucl.Phys.B470(96)3
          9 x F2     H1      N/A    203 Nucl.Phys.B470(96)3
          9 x F2     H1      N/A    204 Nucl.Phys.B470(96)3
```

GeParD — interactive example 2/12

```
 5 x F2      H1       N/A    209 Nucl.Phys.B470(96)3
 4 x F2      H1       N/A    210 Nucl.Phys.B470(96)3
 4 x F2      H1       N/A    211 Nucl.Phys.B470(96)3
 2 x F2      H1       N/A    212 Nucl.Phys.B470(96)3
----------------------------------------------
TOTAL = 85
```

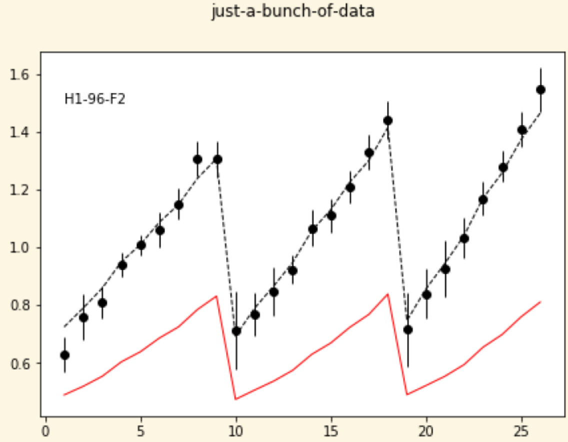
```
Out[14]: 85
```

```
In [15]: pt = DISpoints[0]  # first point from the set
         pt.xB, pt.Q2, pt.y1name, pt.y1unit, pt.val, pt.err
```

```
Out[15]: (0.013, 12.0, 'F2', 1, 0.531, 0.05186520991955976 4)
```

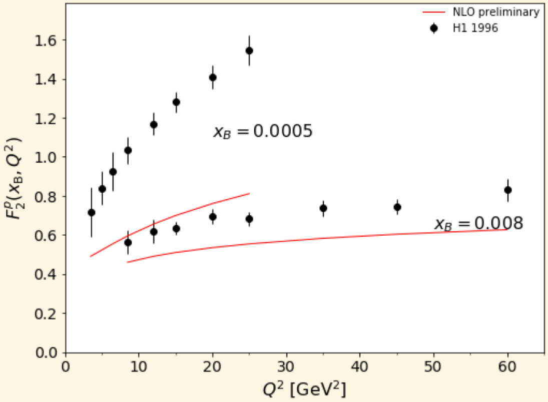
GeParD — interactive example 3/12

```
In [42]: fig = plots.jbod(points=DISpoints[44:70], lines=[th, KM15])
```



GeParD — interactive example 4/12

```
In [41]: fig = plots.HERAF2Q2(lines=[th])   # specificaly designed plots
```




```
In [18]: th.print_chisq(DISpoints)

P(chi-square, d.o.f) = P(3787.21, 85) = 0.0000
```

GeParD — interactive example 5/12

In [19]:
```
%%time
th.model.release_parameters('NS', 'AL0S', 'AL0G')
f = Fitter.FitterMinuit(DISpoints, th)
f.fit()
```

FCN = 72.85271021975446	TOTAL NCALL = 115	NCALLS = 115
EDM = 1.082463486221191e-06	GOAL EDM = 1e-05	UP = 1.0

Valid	Valid Param	Accurate Covar	PosDef	Made PosDef
True	True	True	True	False
Hesse Fail	HasCov	Above EDM		Reach calllim
False	True	False		False

±	Name	Value	Hesse Error	Minos Error-	Minos Error+	Limit-	Limit+	Fixed?
0	NS	0.149507	0.00706756					No
1	AL0S	1.06274	0.0197111					No
2	ALPS	0.15	1					Yes

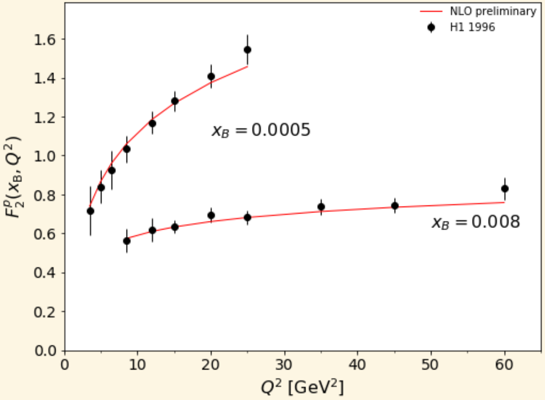
- Two "fitters" are implemented: Minuit and Neural network

GeParD — interactive example 6/12

In [20]:
```
th.print_chisq(DISpoints)
```

```
P(chi-square, d.o.f) = P(72.85, 82) = 0.7549
```

In [21]:
```
fig = plots.HERAF2Q2(lines=[th])
```



GeParD — interactive example 7/12

```
In [22]: utils.describe_data(DVMPpoints)

         npt x obs    collab  FTn    id  ref.
         ---------------------------------------------
          5 x X      H1      N/A    76  arXiv:0910.5831
         20 x X      H1      N/A    79  arXiv:0910.5831
         ---------------------------------------------
         TOTAL = 25

Out[22]: 25

In [23]: th.print_chisq(DVMPpoints)

         P(chi-square, d.o.f) = P(1752.85, 22) = 0.0000

In [24]: th.model.fix_parameters('NS', 'AL0S', 'AL0G')
         th.model.release_parameters('M02S', 'M02G', 'SECS', 'SECG', 'THIS', 'THIG')

In [25]: %%time
         f = Fitter.FitterMinuit(DVMPpoints, th)
         f.fit()

         CPU times: user 9min 55s, sys: 582 ms, total: 9min 55s
         Wall time: 25.7 s

In [30]: th.print_chisq(DISpoints+DVMPpoints)

         P(chi-square, d.o.f) = P(85.51, 104) = 0.9066
```

GeParD — interactive example 8/12

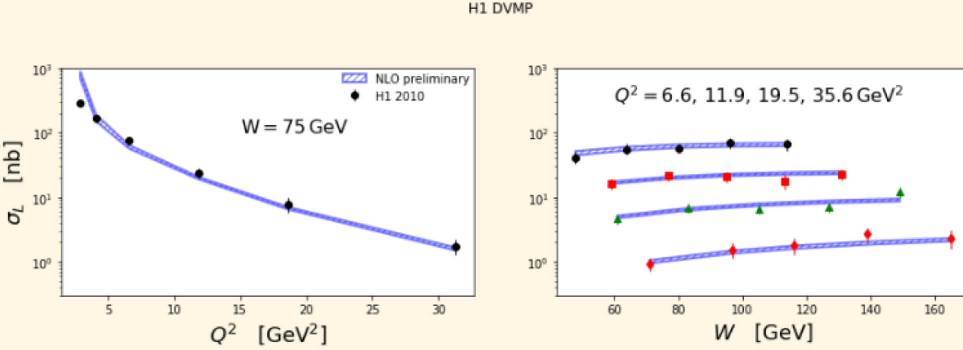
```
In [22]: utils.describe_data(DVMPpoints)

npt x obs    collab  FTn    id  ref.
----------------------------------------
 5 x X       H1      N/A    76  arXiv:0910.5831
20 x X       H1      N/A    79  arXiv:0910.5831
----------------------------------------
TOTAL = 25

Out[22]: 25

In [23]: th.print_chisq(DVMPpoints)

P(chi-square, d.o.f) = P(1752.85, 22) = 0.0000
```

GeParD — interactive example 9/12

```
[24]: th.model.fix_parameters('NS', 'AL0S', 'AL0G')
      th.model.release_parameters('M02S', 'M02G', 'SECS', 'S

[25]: %%time
      f = Fitter.FitterMinuit(DVMPpoints, th)
      f.fit()

      CPU times: user 9min 55s, sys: 582 ms, total: 9min 55s
      Wall time: 25.7 s

30]: th.print_chisq(DISpoints+DVMPpoints)

      P(chi-square, d.o.f) = P(85.51, 104) = 0.9066
```

GeParD — interactive example 10/12

```
fig = plots.DVMP(wdep, bands=[th])
```



GeParD — interactive example 11/12

```
th.m.print_covariance()
```

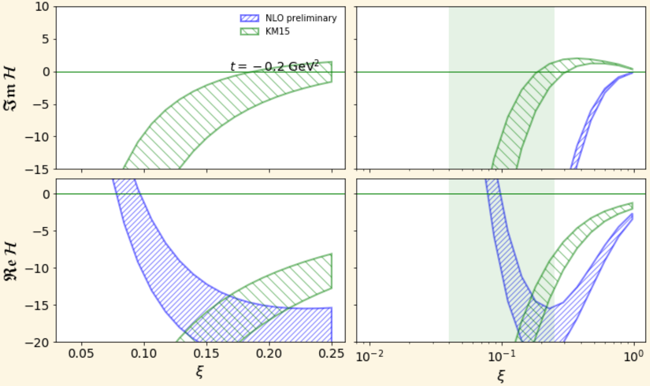
```
      |   M02S    SECS    THIS    M02G    SECG    THIG
------+------------------------------------------------
 M02S |  0.006  -0.010  -0.000   0.001  -0.007   0.002
 SECS | -0.010   0.085  -0.008  -0.002  -0.016  -0.003
 THIS | -0.000  -0.008   0.007   0.001  -0.003  -0.002
 M02G |  0.001  -0.002   0.001   0.011   0.003   0.004
 SECG | -0.007  -0.016  -0.003   0.003   0.052  -0.005
 THIG |  0.002  -0.003  -0.002   0.004  -0.005   0.007
```

```
th.m.print_parameters(compare_with=[KM15.m])
```

```
   NS -> 0.15     0.152
 AL0S -> 1.06     1.16
 ALPS -> 0.15     0.15
 M02S -> 0.605    0.482
```

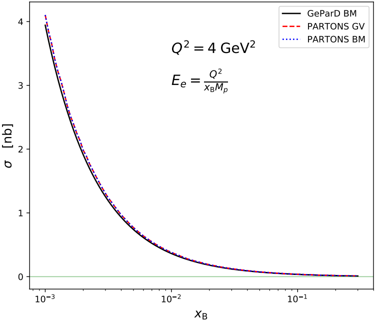
GeParD — interactive example 12/12

In [33]:
```
fig = plots.CFF(bands=[th, KM15])
```



GeParD vs PARTONS, cross-section

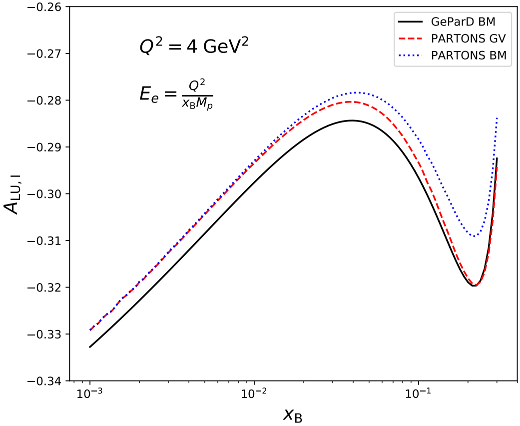
- BM = [Belitsky & Müller], GV = [Guichon & Vanderhaeghen]



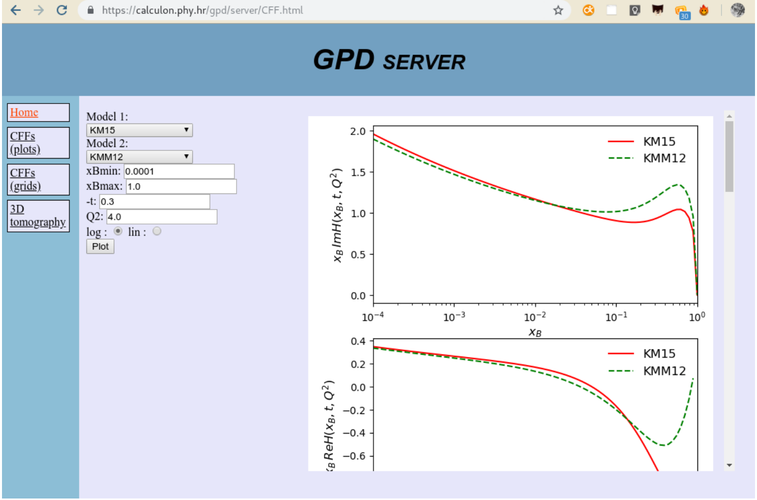
- 100 kinematical points:
  - GeParD: 24 seconds
  - PARTONS (XML driven): 9 seconds

GeParD vs PARTONS, beam spin asymmetry

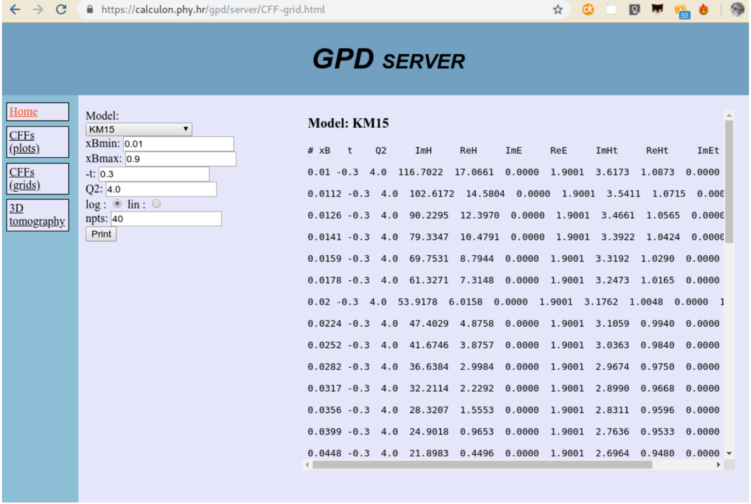
- BM = [Belitsky & Müller], GV = [Guichon & Vanderhaeghen]



GPD/CFF server



GPD/CFF server



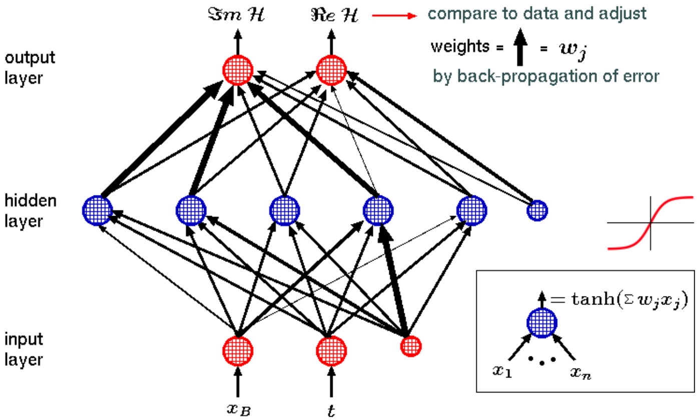
# xB	t	Q2	ImH	ReH	ImE	ReE	ImHt	ReHt	ImEt	
0.01	-0.3	4.0	116.7022	17.0661	0.0000	1.9001	3.6173	1.0873	0.0000	
0.0112	-0.3	4.0	102.6172	14.5804	0.0000	1.9001	3.5411	1.0715	0.000	
0.0126	-0.3	4.0	90.2295	12.3970	0.0000	1.9001	3.4661	1.0565	0.0000	
0.0141	-0.3	4.0	79.3347	10.4791	0.0000	1.9001	3.3922	1.0424	0.0000	
0.0159	-0.3	4.0	69.7531	8.7944	0.0000	1.9001	3.3192	1.0290	0.0000	
0.0178	-0.3	4.0	61.3271	7.3148	0.0000	1.9001	3.2473	1.0165	0.0000	
0.02	-0.3	4.0	53.9178	6.0158	0.0000	1.9001	3.1762	1.0048	0.0000	1
0.0224	-0.3	4.0	47.4029	4.8758	0.0000	1.9001	3.1059	0.9940	0.0000	
0.0252	-0.3	4.0	41.6746	3.8757	0.0000	1.9001	3.0363	0.9840	0.0000	
0.0282	-0.3	4.0	36.6384	2.9984	0.0000	1.9001	2.9674	0.9750	0.0000	
0.0317	-0.3	4.0	32.2114	2.2292	0.0000	1.9001	2.8990	0.9668	0.0000	
0.0356	-0.3	4.0	28.3207	1.5553	0.0000	1.9001	2.8311	0.9596	0.0000	
0.0399	-0.3	4.0	24.9018	0.9653	0.0000	1.9001	2.7636	0.9533	0.0000	
0.0448	-0.3	4.0	21.8983	0.4496	0.0000	1.9001	2.6964	0.9480	0.0000	

GeParD — test suite

```
% nosetests -v
Calculate anomalous dimensions up to NLO. ...  passed
Calculate CFF H. ... passed
Calculate basic cross section Xunp.  ... passed
Calculate long.  polarized cross section XLP in BM10 Approach.  ... passed
Calculate transv.  polarized cross section XTP in BMK Approach.  ... passed
Calculate longitudinal TSA in BM10 Approach.  ... passed
Calculate transversal TSA in BMK Approach and frame.  ... passed
Calculate NLO DVMP coef.  functions ... passed
Calculate LO DVMP TFFs for rho production at input scale.  ... passed
Calculate LO DVMP TFFs for rho production + evolution.  ... passed
Test model:  KM09a ... passed
Test GK model:  Re(CFFH) ... passed
Non-singlet NLO CFF H ... passed
Singlet NLO CFF H ... passed
Calculate GPDs on trajectory xi=x ... passed
Calculate GPDs on trajectory xi=0 ... passed
Non-singlet LO CFF H evolved ... passed
Singlet LO CFF H evolved ... passed
Calculate LO DVCS partial cross section using gepard (no evolution).  ... passed
Calculate NLO DVCS cross section using gepard (+evolution).  ... passed
GepardDR with switched-off DR should be same as Gepard.  ... passed
Testing fitting by FitterMinuit.  ... passed
Test fitting to HERA DVCS via gepard nl-SO3 at NLO ... passed
Singlet NLO MSBAR CFF H evolved ... passed
Testing basic Neural Net framework.  ... passed
Testing Neural Net fitting by FitterBrain.  ... passed
[...]

----------------------------------------------------------------------
64 tests run in 12.391 seconds (64 tests passed)
```

Neural net fits

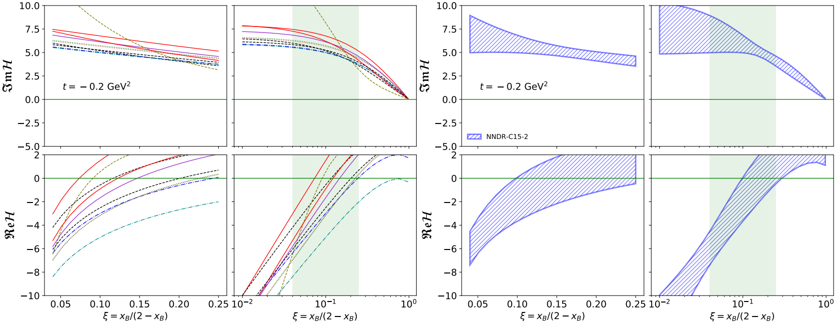
Fitting with neural networks



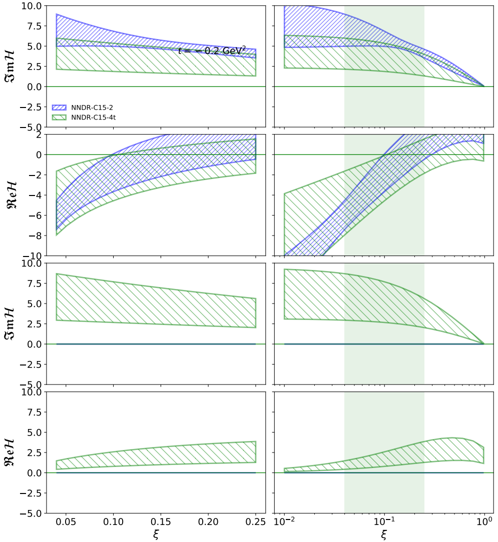
- Essentially a least-square fit of a complicated many-parameter function. $f(x) = \tanh(\sum w_i \tanh(\sum w_j \cdots))$ $\Rightarrow$ no theory bias

Study A: NN fit to CLAS 2015 data

- We start by fitting just to the CLAS 2015 $d\sigma$ and $\Delta\sigma$ measurements [Jo et al. '15], and just $\mathcal{H}$
- We utilize dispersion relations (one NNet represents $\mathfrak{Im}\,\mathcal{H}$, another represents $D(t)$)
- Uncertainty is estimated by averaging over ensemble of neural nets:

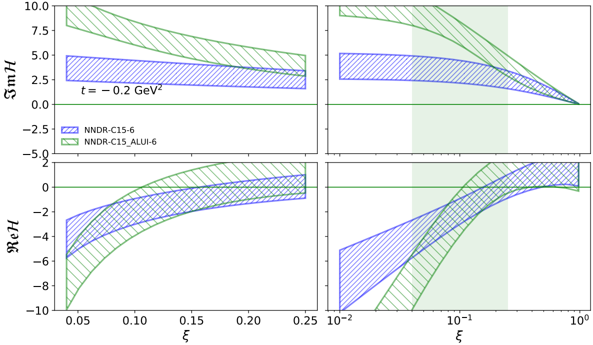


Model dependence: $\mathcal{H}$ vs. $\mathcal{H}+\widetilde{\mathcal{H}}$



Adding more data points

- Adding HERMES $A_{LU,I}$ data. (Model now includes $\mathcal{H}$ and $\mathcal{E}$)



Study B: NN fit to world fixed target data

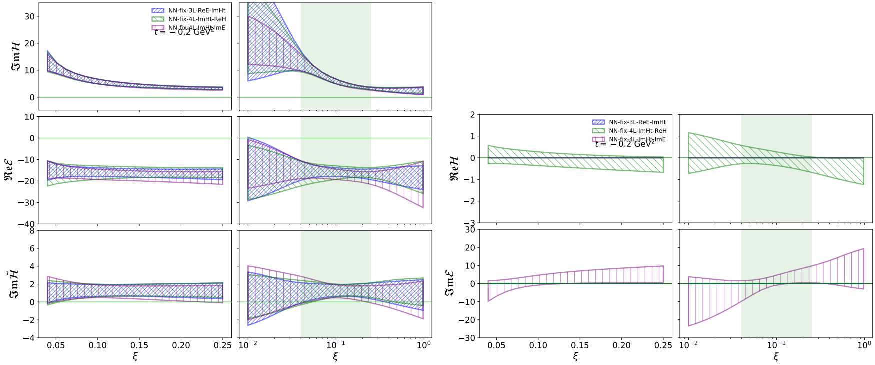
- Representative subset of world DVCS fixed target data:

```
npt x obs    collab  harm.  ref.
------------------------------------------
 6 x ALUI   HERMES  -1.0    arXiv:1203.6287
12 x AUTDVCS  HERMES  0     arXiv:0802.2499
12 x AUTI   HERMES  1.0     arXiv:0802.2499
 6 x BCA    HERMES  0.0     arXiv:1203.6287
 6 x BCA    HERMES  1.0     arXiv:1203.6287
12 x BSDw   CLAS    -1      arXiv:1504.02009
15 x BSDw   HALLA   -1      arXiv:1504.05453
12 x BSSw   CLAS    0.0     arXiv:1504.02009
12 x BSSw   CLAS    1.0     arXiv:1504.02009
10 x BSSw   HALLA   0.0     arXiv:1504.05453
10 x BSSw   HALLA   1.0     arXiv:1504.05453
 6 x BTSA   HERMES  0.0     arXiv:1004.0177v1
 3 x TSA    CLAS    -1      arXiv:hep-ex/0605012
 6 x TSA    HERMES  -1.0    arXiv:1004.0177v1
------------------------------------------
TOTAL = 128
```

- We now use completely unconstrained neural nets representing $\mathfrak{Im}\,\mathcal{H}$, $\mathfrak{Re}\,\mathcal{H}$, $\mathfrak{Im}\,\mathcal{E}$, $\mathfrak{Re}\,\mathcal{E}$, ... (do not assume dispersion relations)

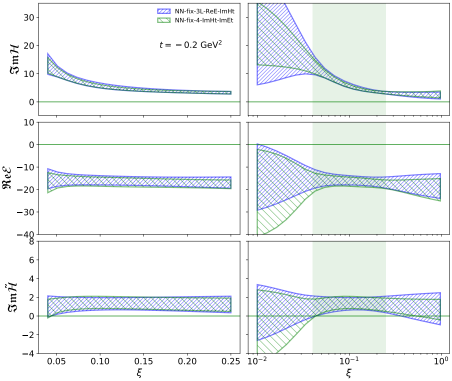
Results (1/2)

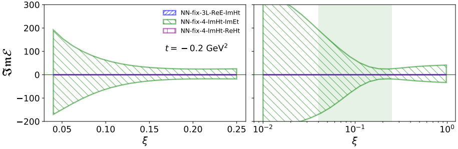
- Only $\mathfrak{Im}\,\mathcal{H}$, $\mathfrak{Im}\,\widetilde{\mathcal{H}}$ and $\mathfrak{Re}\,\mathcal{E}$ consistently extracted as different from zero, and, with somewhat smaller significance, $\mathfrak{Re}\,\mathcal{H}$ and $\mathfrak{Im}\,\mathcal{E}$:



Results (2/2)

- Other CFFs come out consistent with zero. Only bounds on their size are obtained. *E. g.* for $\mathfrak{Im}\,\widetilde{\mathcal{E}}$:





Summary

- Global fits to DVCS data reasonably healthy (some problems with newest Hall A data)
- Substantial effort will be needed to include DVMP in global NLO fits and estimate EIC impact, but all components for such analysis are in principle known
- GeParD software can maybe be useful to community. Should be released "real soon now".
- Neural network method has a unique capability of extraction of Compton form factors (and, later, GPDs) with reliable uncertainties