UNIVERSITÄT GRAZ

NAWI Graz
Natural Sciences

Towards TMDs with contour deformations

Eduardo Ferreira

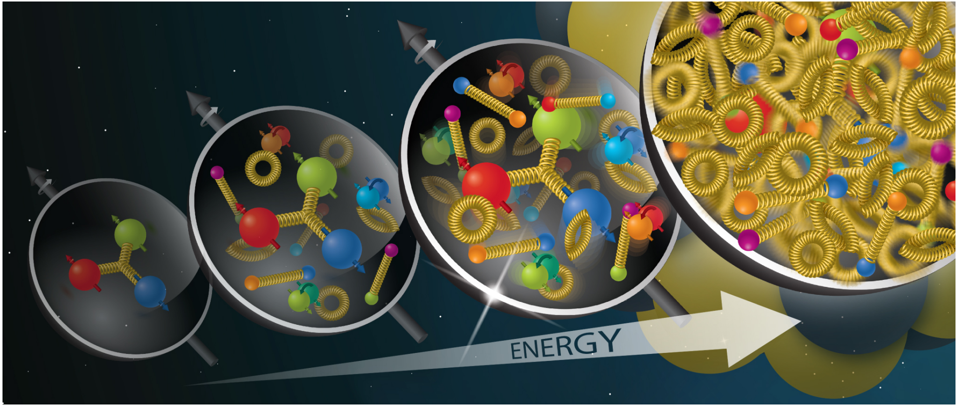
with Gernot Eichmann, Alfred Stadler

Quantfunc2024, Valencia – September 2-6, 2024

1 Partonic structure

Partonic Structure Functions

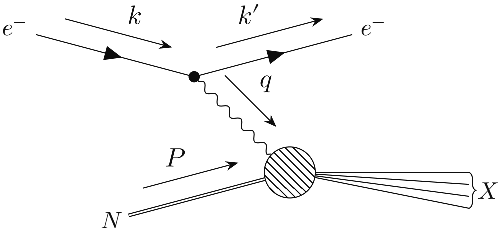
- PDFs, TMDs, etc... describe this picture:



(Picture source: EIC website)

- Different energy scales probe different aspects of the hadron.
- Contribution from **valence** and **sea** partons.

Parton model



(PDG Section 18 (Structure Functions))

- Common experiment:
  Deep inelastic scattering.
- Electron (or neutrino) probe interacts with nucleon via $\gamma, Z, W^{\pm}$.
- Main idea:
  - Hadrons are *bags* of **partons**.
  - Scattering occurs between the exchanged boson and a parton from the nucleon
- Cross-section separates:

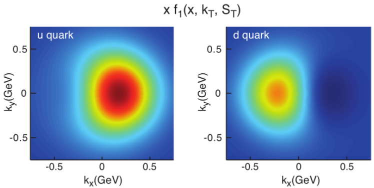
$$\frac{d^2\sigma}{dx dq^2} \propto L_{\mu\nu} W^{\mu\nu}$$

- $W^{\mu\nu}$ described via PDFs.

Beyond PDFs: TMDs and GPDs

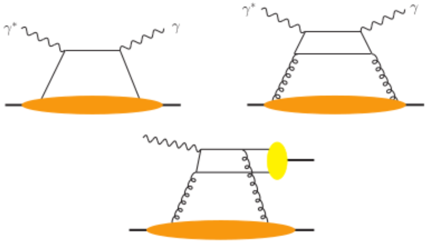
- **Transverse momentum distributions (TMDs):**
  - Info on confined motion of partons
  - Semi-inclusive DIS experiments
  - Transverse motion of the partons in the hadron
  - Generalization of the well-known PDFs:

$$\int d^2\mathbf{k}_T\, f(x, k_T, Q^2) = f(x, Q^2)$$



(Left: EIC White Paper, arXiv:1212.1701), (Right: Diehl, M; 2016)

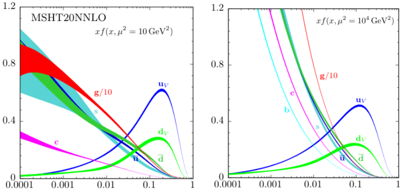
- **Generalized Parton Distributions (GPDs):**
  - Non-zero momentum transfer
  - Exclusive processes: Deeply virtual Compton scattering, meson production, ...
  - Transverse and spin structure
  - Path to form factors



Hadrons on the Light Front

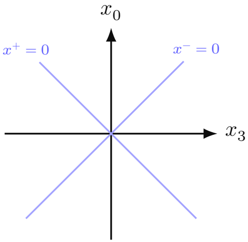
Goal: Use DSE/BSE to study hadrons on the light front, $x^+ = 0$.

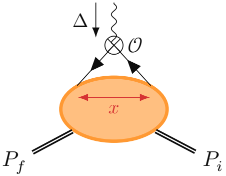
- Natural frame for defining parton distribution functions: PDFs, TMDs, ...



- Future: COMPASS/AMBER @ CERN
  EIC @ Brookhaven National Laboratory.

(AMBER: arXiv:1808.00848)
(EIC: arXiv:2305.14572)



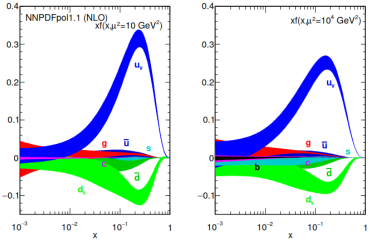


$$\langle P_f | \, \mathrm{T}\Phi(x)\mathcal{O}\Phi(0) \, | P_i \rangle$$

Hadrons on the Light Front

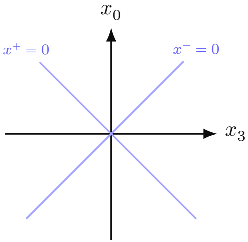
Goal: Use DSE/BSE to study hadrons on the light front, $x^+ = 0$.

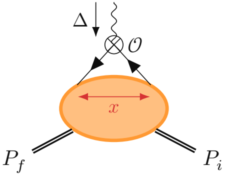
- Natural frame for defining parton distribution functions: PDFs, TMDs, ...



- Future: COMPASS/AMBER @ CERN
  EIC @ Brookhaven National Laboratory.

(AMBER: arXiv:1808.00848)
(EIC: arXiv:2305.14572)





$$\langle P_f | \, \mathrm{T}\Phi(x)\mathcal{O}\Phi(0) \, | P_i \rangle$$

Hadronic quantities

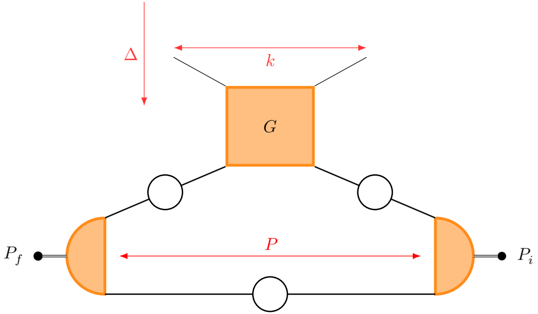
$$
\begin{tikzcd}
 & f(k,P) \arrow[d, "\int dk^-"'] & & H(k,P,\Delta) \arrow[ll, "\Delta=0"'] \arrow[d, "\int dk^-"] & \text{Correlation function} \\
\text{TMD} & f(x,\vec{k}) \arrow[d, "\int d^2\vec{k}_\perp"'] & W(x,\vec{k},\vec{b}) \arrow[l, "\int d^2\vec{b}"'] \arrow[d, "\int d^2\vec{k}_\perp"] & H(x,\vec{k},\xi,\vec{\Delta}) \arrow[l, "\text{FT, } \xi=0"'] \arrow[d, "\int d^2\vec{k}_\perp"] & \text{GTMD} \\
\text{PDF} & f(x) & f(x,\vec{b}) \arrow[l, "\int d^2\vec{b}"'] \arrow[d, "\int dx x^{n-1}"] & H(x,\xi,\Delta^2) \arrow[l, "\text{FT, } \xi=0"'] \arrow[d, "\int dx x^{n-1}"] & \text{GPD} \\
 & \text{FFs} & F_n(\Delta^2) & \sum_{k=0}^{n} A_{nk}\left(\Delta^2\right)(2\xi)^k \arrow[l, "\xi=0"'] & \text{GFFs}
\end{tikzcd}
$$

(Lorce, Pasquini, Vanderhaeghen; 2011), (Picture adapted from: Diehl, 2016), (Diehl, 2003), (Meißner, Goeke, Metz, Schlegel; 2008), (Meißner, Metz, Schlegel; 2009)

2 Triangle Diagram

Our Goal

- **Main Goal:** Get partonic distribution functions from hadron-hadron correlations via **FUN**ctional Methods



- G is the four-point quark correlation function, calculated with scattering equation.
- The quark propagator is calculated via quark DSE.
- The BSWF is calculated via the meson BSE.

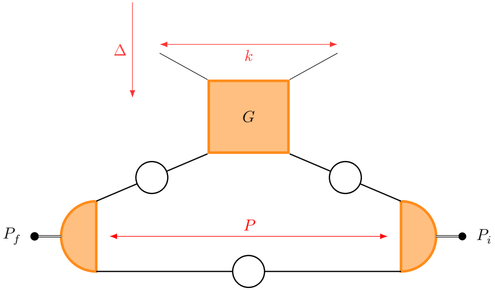
(Mezrag; 2015), (Diehl, Gousset; 1998), (Tiburzi, Miller; 2003), (Mezrag, Chang, Moutarde, Roberts, Rodríguez-Quintero, Sabatié, Schmidt; 2015), many others, ...

$$\mathcal{G}^{[\Gamma]}(P,k,\Delta) = \frac{1}{2}\mathrm{Tr}\left[\int dk^- \int \frac{d^4z}{2\pi^4} e^{ik\cdot z}\langle P_f|\,\overline{\psi}(z)\mathcal{W}\Gamma\psi(0)\,|P_i\rangle\right]$$

- Partonic distributions are calculated by integrating the correlator in k^- and taking appropriate traces.

Triangle Diagram

- We start by solving a simple model, and gradually build up the complexity of the calculation.
  - Hadrons are on-shell: $P^2 = -M^2$

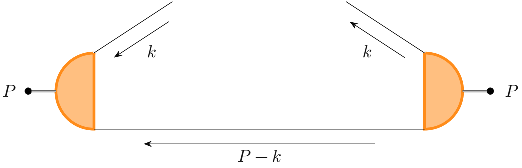


$$\Delta = \sqrt{\Delta^2}\begin{pmatrix}0\\0\\1\\0\end{pmatrix} \qquad P = 2m\sqrt{t}\begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

$$k = \left(\frac{1+\alpha}{2}\right)P + M\sqrt{R}\begin{pmatrix}0\\0\\\sqrt{1-Z^2}\\Z\end{pmatrix}$$

Triangle Diagram

- We start by solving a simple model, and gradually build up the complexity of the calculation.
  - Hadrons are on-shell: $P^2 = -M^2$
  - **Forward limit:** $\Delta \to 0$ – We get the PDFs and TMDs



$$P = 2m\sqrt{t}\begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

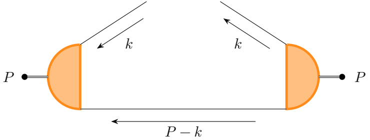
$$k = \left(\frac{1+\alpha}{2}\right)P + m\sqrt{R}\begin{pmatrix}0\\0\\\sqrt{1-Z^2}\\Z\end{pmatrix}$$

- Steps:
  1. Calculate BSWF;
  2. Calculate the correlator;
  3. Project to the light-front;

Light-front Projection

- The TMD is defined as:

$$\int dk^-$$



- In our kinematic variables:

$$k^- \propto -2i\sqrt{R}Z$$

- Need the BSWF in $Z \in (-\infty, \infty)$.
- Use Padé approximants:

$$f(Z) = \frac{a_0 + a_1 Z + a_2 Z^2 + \cdots + a_N Z^N}{1 + b_1 Z + b_1 Z^2 + \cdots + b_M Z^M}$$

(L. Schlessinger, 1968) (Tripolt et al., 2019) (D. Binosi, R-A. Tripolt; 2019)

- Keep analytic structure in mind – **contour deformations**

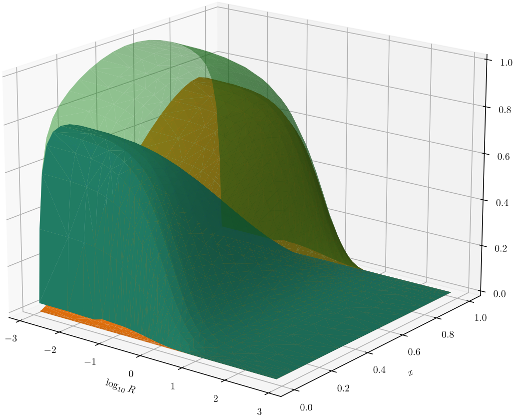
Definition of the LFWF

$$\mathrm{TMD}(R, \alpha) \propto -2i\sqrt{R}\int_{-\infty}^{\infty} dZ\, \mathcal{G}(R, Z, \alpha)$$

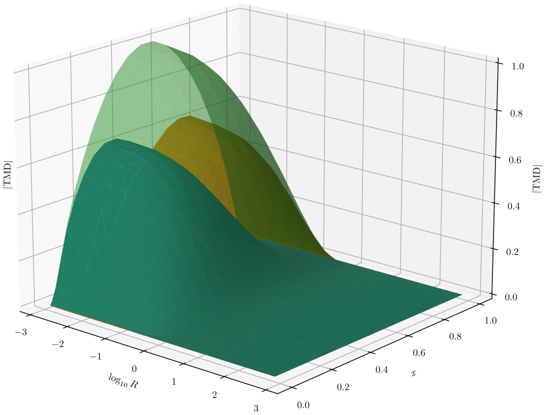
(Eichmann, EF, Stadler; 2022)

TMD: Some results

- $\sqrt{t} = 1/3i, \beta = 4, c = 1$



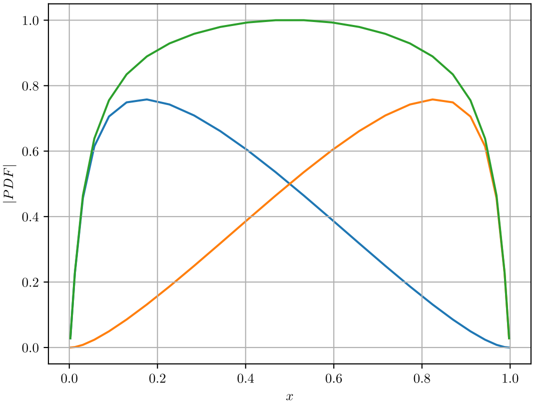
- $\sqrt{t} = 0.5i, \beta = 1, c = 1$



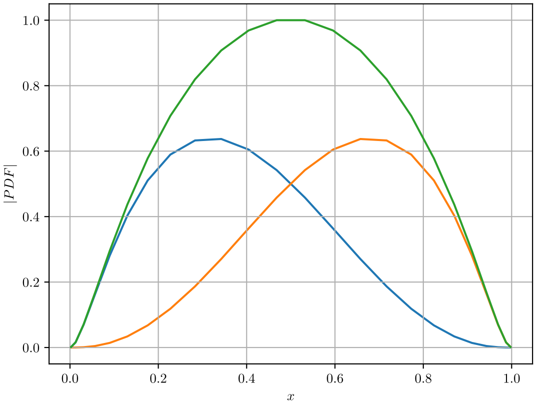
- Upper line open
- Lower line open
- Sum of both

PDFs

- $\sqrt{t} = 1/3i, \beta = 4, c = 1$



- $\sqrt{t} = 0.5i, \beta = 1, c = 1$



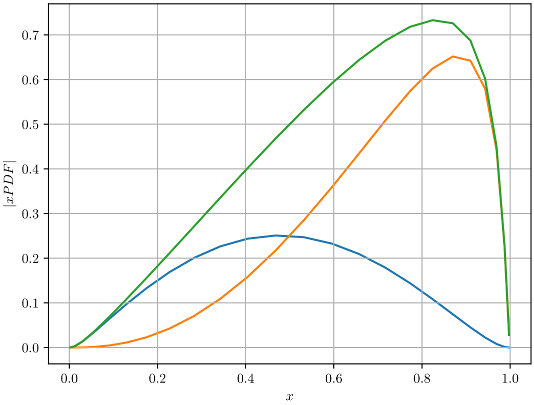
Upper line open

Lower line open

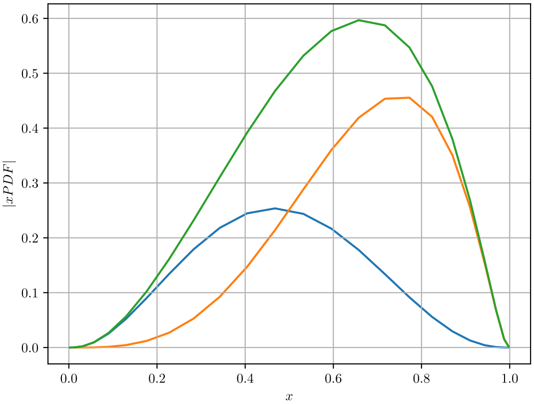
Sum of both

PDFs

- $\sqrt{t} = 1/3i, \beta = 4, c = 1$



- $\sqrt{t} = 0.5i, \beta = 1, c = 1$



Upper line open

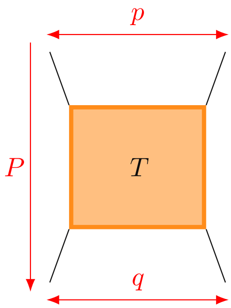
Lower line open

Sum of both

3 Four-point function

4-point function

- 4- point function determined from scattering equation:
  $\mathbf{G} = \mathbf{G_0} + \mathbf{G_0 T G_0} \implies \mathbf{T} = \mathbf{K} + \mathbf{K G_0 T} \implies \mathbf{T} = (\mathbb{1} - \mathbf{K G_0})^{-1}\,\mathbf{K}.$



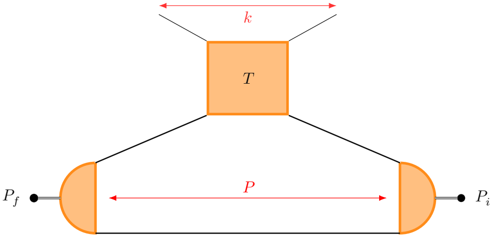
$$T(t, X, R, Z, Y, Q) = K(X, R, p\cdot q)$$
$$+\frac{1}{2}\frac{m^4}{2\pi^4}\int_0^\infty dx\, x \int_{-1}^{1} dz\,\sqrt{1-z^2}G_0\,(x, z, t)$$
$$\times \int_{-1}^{1} dy \int_0^{2\pi} d\Psi K(X, x, k\cdot q)T(t, x, y, z, R, Q)$$

- Fully off-shell: 6 Lorentz invariants
  - 3 radial: X, t, R;
  - 3 angular: Y, Z, Q;

- Same G_0 and K as in the BSE
- Solved numerically as linear system
- Contains *all* dynamics of 2 ϕ particles
  - **Must produce bound state poles dynamically!**

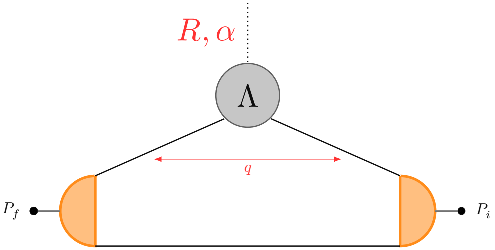
(Eichmann, Duarte, Peña, Stadler; 2019)http://arxiv.org/abs/1907.05402

Including the four-point function

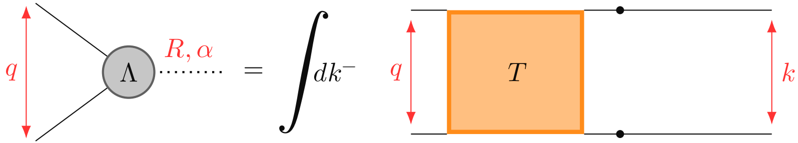


- *Close* the triangle.
- LF projection is now done in the open legs of T.

Including the four-point function



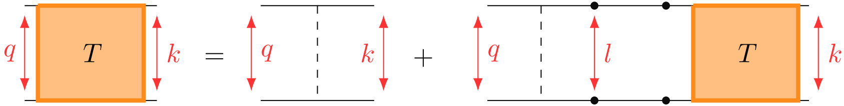
- *Close* the triangle.
- LF projection is now done in the open legs of T.
- Need a **light-front vertex.**
- Triangle is now a loop diagram
- Contains all dynamics between the two open legs:
  - Infinite ladder of χ exchanges.



- Can be calculated from the scattering equation.

Equation for the vertex

- Do the light-front projection in the scattering equation



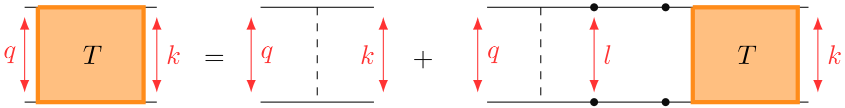
Scattering equation

$$\mathbf{T}(q,k) = \mathbf{K}(q,k) + \int d^4l\, \mathbf{K}(q,l)\mathbf{G_0}(l)\mathbf{T}(l,k)$$

(Bednar, Cloët, Tandy; 2020)

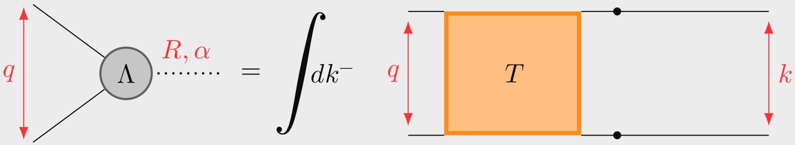
Equation for the vertex

- Do the light-front projection in the scattering equation



- Introduce the definition for Λ in the equation.

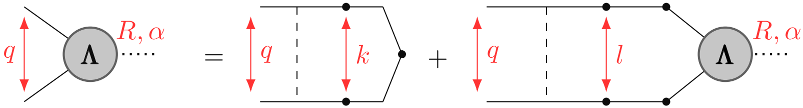
Definition of the vertex



(Bednar, Cloët, Tandy; 2020)

Equation for the vertex

- Do the light-front projection in the scattering equation



- LF projection happens in inhomogeneous term

Vertex equation

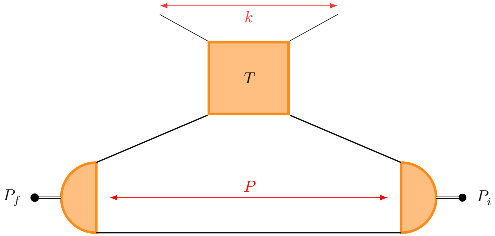
$$\mathbf{\Lambda}(q,R,\alpha) = \int dk^- \; \mathbf{K}(q,k)\mathbf{G_0}(k) + \int d^4l \; \mathbf{K}(q,l)\mathbf{G_0}(l)\mathbf{\Lambda}(l,R,\alpha)$$

- Kernel and G_0 fully known – no extrapolation needed!

(Bednar, Cloët, Tandy; 2020)

A leap-of-faith *Put everything together*

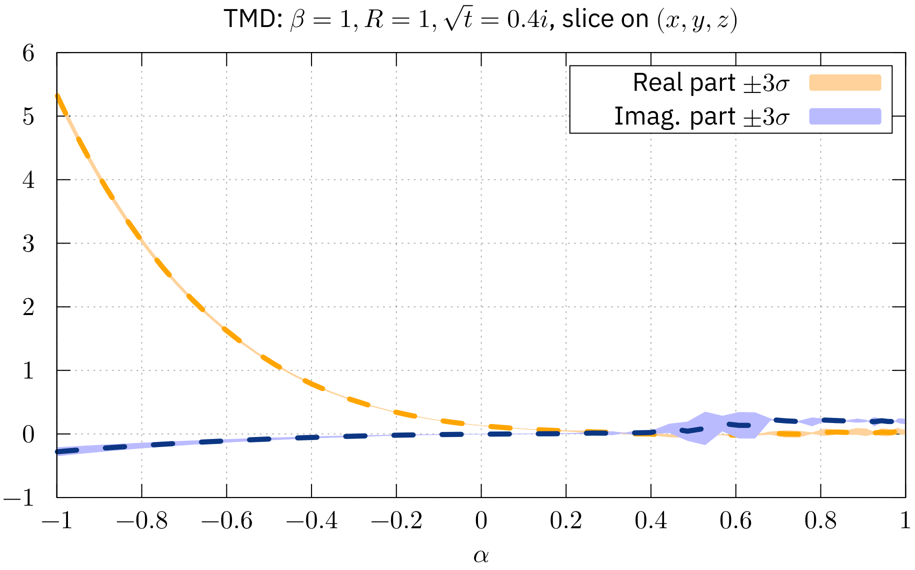
- Can we just add the full 4-point function? – **YES!**



1. Calculate 4-point function
2. Calculate BSE
3. Do the loop integration
4. Project to LF
5. Do estimate on extrapolation error

- We solve one triangle for each point in the R, α grid – **HPC Needed!**
  - $N_\alpha \times N_R \times N_Z$ 4-point functions!
- **Preliminary** result.

Results


TMD: $\beta = 1, R = 1, \sqrt{t} = 0.4i$, slice on (x, y, z)
Real part $\pm 3\sigma$
Imag. part $\pm 3\sigma$
6
5
4
3
2
1
0
−1
−1
−0.8
−0.6
−0.4
−0.2
0
0.2
0.4
0.6
0.8
1
α

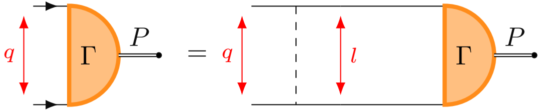
Conclusions

- We have a method for TMDs/PDFs
- We have incorporated BSE solutions
- We have implemented contour deformations – resonances!
- We have made progress in the inclusion of the 4-point function
- Still, analytic details need to be ironed out

Backup

Scalar toy model *Bethe-Salpeter Equation*

- Scalar model:
  - ϕ of mass m
  - χ of mass μ
- BS amplitude for bound-state of two ϕ:



$$q = \frac{\alpha}{2} P + m\sqrt{\rho} \begin{pmatrix} 0 \\ \times \\ \times \\ \times \end{pmatrix} \qquad l = \frac{\alpha}{2} P + m\sqrt{l} \begin{pmatrix} \times \\ \times \\ \times \\ \times \end{pmatrix}$$

(Wick; 1954) (Cutkosky; 1954)

- The BSWF is a function of the kinematic invariants:

$$-M^2 = P^2 \qquad \frac{q^2}{m^2} \propto \rho \qquad z \propto \hat{k} \cdot \hat{P}$$

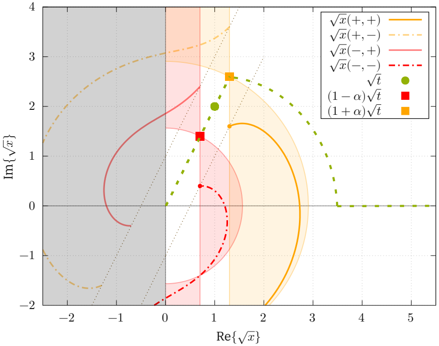
$$\psi(\rho, z, x) = \frac{m^4}{(2\pi)^3} \frac{1}{2} \int_0^\infty dl\, l$$
$$\times \int_{-1}^{1} dz_l \sqrt{1 - {z_l}^2}\, \mathbf{G_0}(l, z_l, \alpha)$$
$$\times \int_{-1}^{1} dy_l\, \mathbf{K}(\rho, z, l, z_l, y_l)\, \psi(l, z_l, \alpha)$$

Analytic Structure of BSE $\mathrm{Im}\sqrt{t} > \min\left(\frac{1}{1\pm\alpha}\right)$

$$\mathbf{G_0}^{-1} = (l_+^2 + m^2)(l_-^2 + m^2)$$

- Branch cuts in complex $\sqrt{l}$ plane

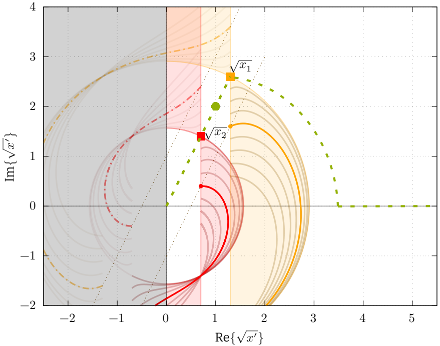
$$\sqrt{l}_\pm^\lambda = \mp(1\pm\alpha)\sqrt{t}\left[z_l + i\lambda\sqrt{1 - z_l^2 + \frac{1}{(1\pm\alpha)^2 t}}\right]$$



$$\mathbf{K}^{-1} = (q-l)^2 + \mu^2$$

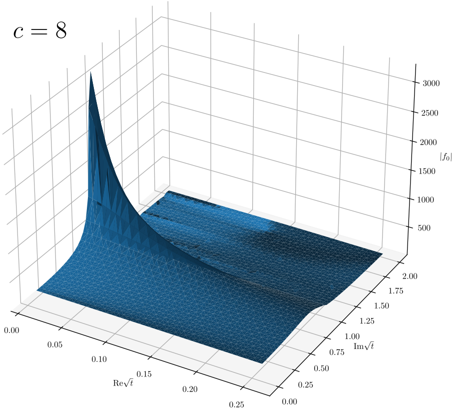
- Branch cuts in complex $\sqrt{l}$ plane depend on path taken:

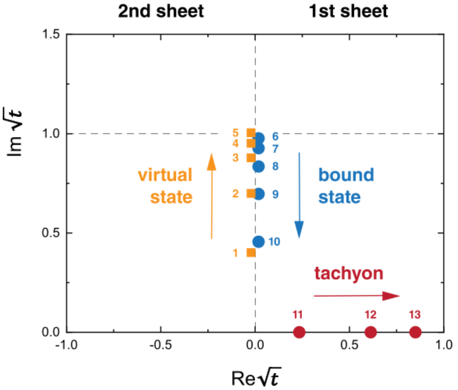
$$\sqrt{l} = \sqrt{\rho}\left(\Omega \pm i\sqrt{1 - \Omega^2 + \frac{\beta^2}{\rho}}\right)$$



4-point function results

- T is made fully on-shell
- Partial-wave expansion shows bound-state pole in the first Riemann sheet





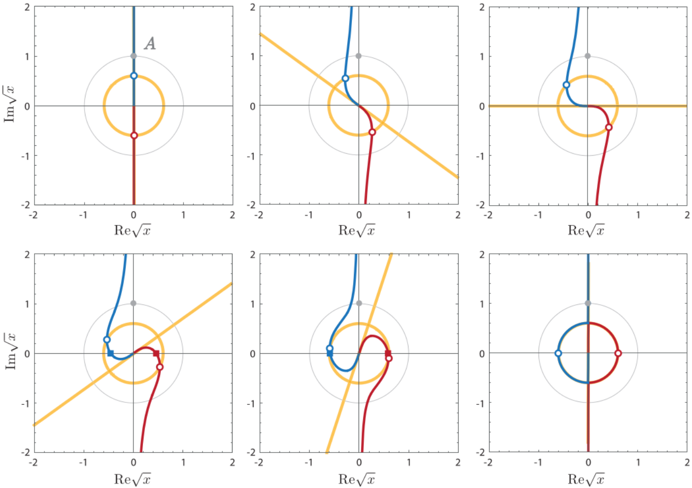
(Eichmann, Duarte, Peña, Stadler; 2019)

- Describes both long-range and short-range qq dynamics.

AC cuts

- Form of cuts is very common:

$$\sqrt{x}_{\pm} = A\left[z \pm \sqrt{z^2 + C^2}\right]$$



AC cuts

- Form of cuts is very common:

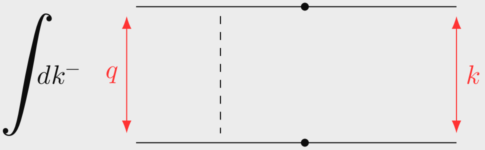
$$\sqrt{x}_{\pm} = A\left[z \pm \sqrt{z^2 + C^2}\right]$$

  - A scales (and rotates if imaginary).
  - Curves begin at $\pm\infty$; end at origin.
  - Near origin, curves follow AC^2 direction.
  - For a certain $\arg C$, the curves cross the real axis.
  - Purely imaginary $C = id$ gives rise to the circle around the origin.

Projecting the inhomogeneous term

- Easier said than done...
- Very challenging to find a good contour deformation that works.

Inhomogeneous term



- Residue theorem is a perfect candidate

Projecting the inhomogeneous term

- Easier said than done...
- Very challenging to find a good contour deformation that works.

Hyperspherical coordinates

$$\int dZ \, \frac{1}{\left[R + (1+\alpha)^2 t + 2(1+\alpha)\sqrt{t}\sqrt{R}Z + 1\right]^2} \frac{g^2}{R + \rho - 2\sqrt{R}\sqrt{\rho}\Omega + \beta^2}$$

$$\Omega = zZ + \sqrt{1-z^2}\sqrt{1-Z^2}\,y$$

- Residue theorem is a perfect candidate
- Hyperspherical coordinates have issues with $\sqrt{1-Z^2}$

Projecting the inhomogeneous term

- Easier said than done...
- Very challenging to find a good contour deformation that works.

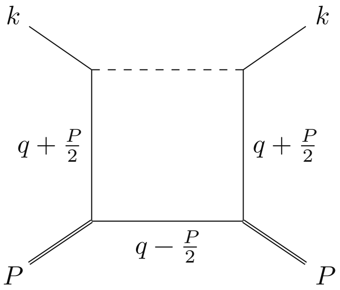
LF coordinates (before Wick rotation)

$$\int dk^- \frac{1}{\left[R^2-(1+\alpha)^2 t+1-\sqrt{t}(1+\alpha)k^- - i\epsilon\right]^2} \frac{g^2}{R^2-\rho^2+\beta^2-2\sqrt{R}\sqrt{\rho}\Omega+q^+k^- - i\epsilon}$$

- Residue theorem is a perfect candidate
- Hyperspherical coordinates have issues with $\sqrt{1-Z^2}$
- LF coordinates work but poles move around too much

Box diagram

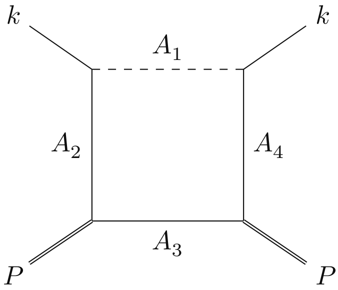
- Take a step back...
- We can get a simpler result: **Box diagram**



- Reduce to bare minimum
- Single χ exchange
- No vertices – point interactions
- “First order” term of the ladder diagram
- **Feynman parameters**

- FP result will give “baseline” result we can fallback on.

Feynman Parameters

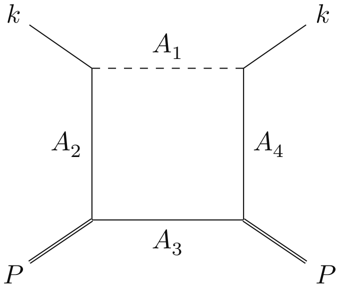


$$\propto \int dx_1 \int dx_2 \int dx_3 \int dx_4 \int d^4 q \times \frac{\delta\left(1 - x_1 - x_2 - x_3 - x_4\right)}{\left[x_1 A_1 + x_2 A_2 + x_3 A_3 + x_4 A_4\right]^4}$$

- Change of variables to simplify loop momentum:

$$l = q + \sum_i x_i p_i \qquad \Delta = \sum_i x_i \left(p_i^2 + m_i^2\right) - \left(\sum_i x_i p_i\right)^2$$

Feynman Parameters



$$\propto \int_0^1 dx_1 \int_0^{1-x_1} dx_2 \int_0^{1-x_1-x_2} dx_3 \int d^4l \, \frac{1}{[l^2+\Delta^2]^4}$$

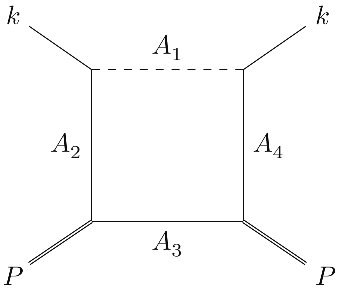
$$\int d^4l \frac{1}{[l^2+\Delta^2]^4} = I_{04} = \frac{1}{12\Delta^2}$$

- Simplify feynman parameters:

$$a = x_1 + x_2 \qquad b = x_1 - x_2 \qquad c = x_3$$

- Integrand is independent of b

Feynman Parameters



$$\propto \int_0^1 da \int_0^{1-a} dc \, \frac{2a}{\Delta^2}$$
$$\Leftrightarrow \int_0^1 de \int_0^e dc \, \frac{2(e-c)}{\Delta^2}$$

- Left with an integral over 2 parameters of a double pole in Z
- Add the two k^2 propagators on top:

$$\mathrm{Box} = \frac{1}{\left[k^2+m^2\right]^2} \int_0^1 de \int_0^e dc \, \frac{1}{\Delta^2}$$

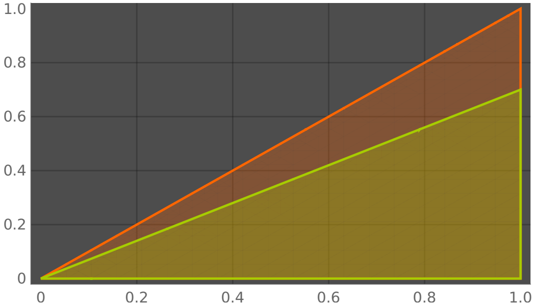
$$\Delta = (1-e)\, ek^2 + (1-c)\, cP^2 + (1-e)\left(\mu^2 - m^2\right) + m^2 - 2\,(1-e)\, ck\cdot P$$

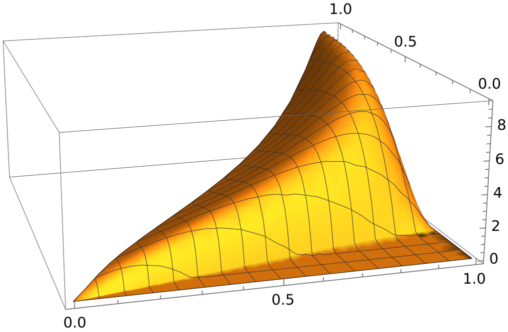
Residues

- We have two double poles: **Propagator** & **Integrand**

$$Z_{\text{prop}} = -\frac{1 + R + t\,(1+\alpha)^2}{2\sqrt{R}\sqrt{t}\,(1+\alpha)} \qquad\qquad Z_\triangle = f(\text{kinematics}, e, c)$$

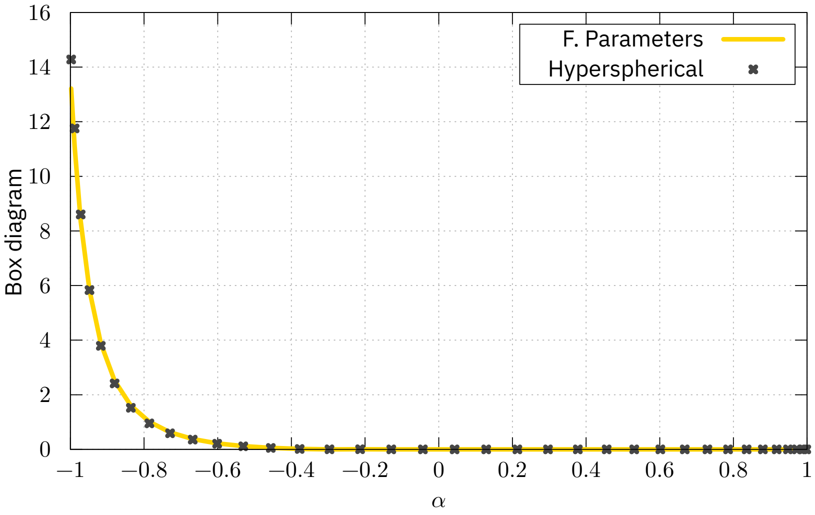
- We can trace the position of $Z_\triangle$ on a e, c plane:





Hyperspherical coordinates?

- We just did a "change of variables" – **Result should be the same!**
- We get the same result (thankfully!)

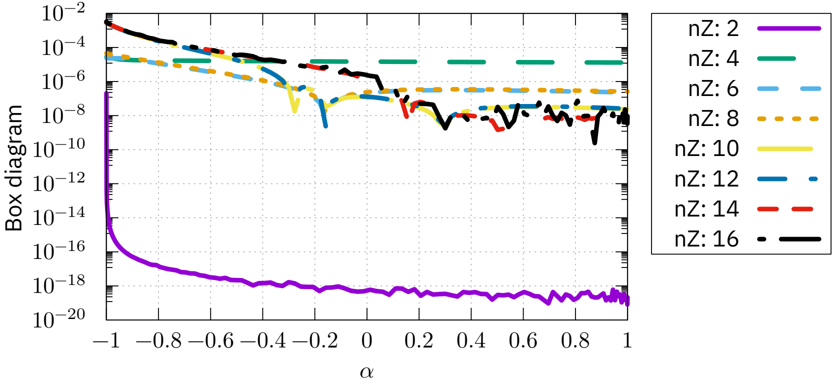


Extrapolation (can be) unreliable

- Schlessinger method is fast but limited – at most $\propto \frac{1}{Z}$
  - Coefficients must be very accurately calculated

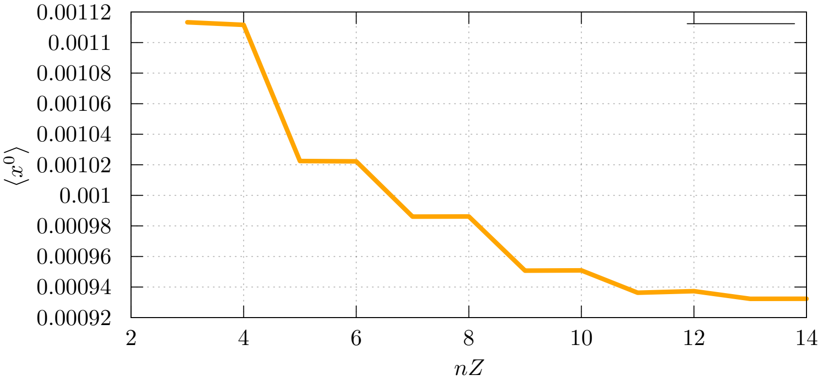
$$R(Z) = \cfrac{f_0}{1 + \cfrac{a_0(Z - Z_0)}{1 + \cfrac{a_1(Z - Z_1)}{1 + \dots}}}$$

- Different input points $\implies$ different output!



Use Padé Approximants

- In this case we know the analytic dependence – can improve extrapolation
- Uncertainty **massively** reduced – automate search for best parameters
  - Use statistics to "approximate" uncertainty



- Figure of merit: **First Mellin moment** $\langle x^0 \rangle = \int d\alpha f(\alpha)$.

Use Padé Approximants

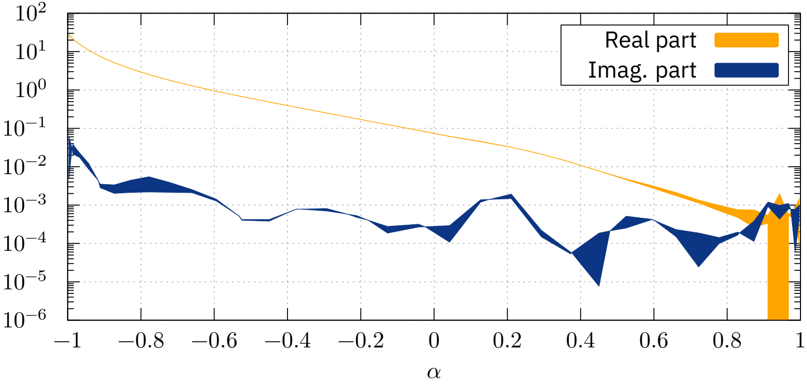
- In this case we know the analytic dependence – can improve extrapolation
- Uncertainty **massively** reduced – automate search for best parameters
  - Use statistics to "approximate" uncertainty

```
[main::main()]                                   Points selected: (-0.971882,0) (-0.964138,0) (-0.637737,0) (-0.559991,0) (-0.367686,0) (-0.108877,0) (-0.00673456,0) (0.333391,0) (0.670008,
0) (0.775018,0) (0.929139,0) (0.895408,0)
[main::main()]                                   Points selected: (-0.906222,0) (-0.991732,0) (-0.816613,0) (-0.536235,0) (-0.373951,0) (-0.0363416,0) (0.119572,0) (0.346285,0) (0.568612,0)
 (0.756554,0) (0.884481,0) (0.988616,0)
[main::main()]                                   Points selected: (-0.994135,0) (-0.788311,0) (-0.861096,0) (-0.537644,0) (-0.301774,0) (-0.168043,0) (0.092449,0) (0.345567,0) (0.642733,0)
(0.72315,0) (0.856472,0) (0.96919,0)
[main::main()]                                   Points selected: (-0.956588,0) (-0.919778,0) (-0.756584,0) (-0.503692,0) (-0.366042,0) (-0.105799,0) (0.110391,0) (0.349409,0) (0.601376,0)
(0.731116,0) (0.850736,0) (0.940282,0)
[main::main()]                                   Points selected: (-0.971467,0) (-0.919462,0) (-0.801734,0) (-0.548751,0) (-0.307002,0) (-0.135181,0) (0.167636,0) (0.316577,0) (0.560352,0)
(0.823707,0) (0.945959,0) (0.824111,0)
[main::main()]                                   Points selected: (-0.996096,0) (-0.947657,0) (-0.756661,0) (-0.517286,0) (-0.396726,0) (-0.202665,0) (0.17211,0) (0.420647,0) (0.61719,0) (0
.769154,0) (0.797286,0) (0.947448,0)
[main::main()]                                   Points selected: (-0.957069,0) (-0.923619,0) (-0.744116,0) (-0.604607,0) (-0.285585,0) (-0.247233,0) (0.159705,0) (0.428975,0) (0.581043,0)
(0.670109,0) (0.839225,0) (0.954249,0)
[main::main()]                                   Points selected: (-0.99626,0) (-0.830503,0) (-0.763215,0) (-0.582946,0) (-0.318577,0) (-0.126642,0) (0.104791,0) (0.361273,0) (0.627808,0) (
0.725125,0) (0.844745,0) (0.970048,0)
[main::main()]                                   Points selected: (-0.894138,0) (-0.916913,0) (-0.744792,0) (-0.498678,0) (-0.387726,0) (-0.085339,0) (0.039761,0) (0.328953,0) (0.54071,0) (
0.617861,0) (0.909084,0) (0.841881,0)
[main::main()]                                   Points selected: (-0.985678,0) (-0.859571,0) (-0.733059,0) (-0.611597,0) (-0.379311,0) (-0.18972,0) (0.184463,0) (0.279375,0) (0.64143,0) (0
.809775,0) (0.867791,0) (0.932604,0)
[main::main()]                                   Points selected: (-0.918332,0) (-0.933731,0) (-0.672537,0) (-0.616266,0) (-0.328011,0) (-0.132243,0) (0.132344,0) (0.427782,0) (0.692186,0)
(0.718246,0) (0.849771,0) (0.965346,0)
[main::main()]                                   Points selected: (-0.886897,0) (-0.924523,0) (-0.722975,0) (-0.424517,0) (-0.363058,0) (-0.124189,0) (0.119086,0) (0.419112,0) (0.556684,0)
(0.678858,0) (0.822687,0) (0.962312,0)
[main::main()]                                   Points selected: (-0.875028,0) (-0.890325,0) (-0.781825,0) (-0.562061,0) (-0.362633,0) (-0.18577,0) (0.206556,0) (0.298088,0) (0.638054,0) (
0.758299,0) (0.86561,0) (0.905794,0)
[main::main()]                                   Points selected: (-0.957465,0) (-0.796093,0) (-0.772831,0) (-0.590603,0) (-0.389053,0) (-0.111661,0) (0.119117,0) (0.400803,0) (0.550732,0)
(0.688639,0) (0.889395,0) (0.977207,0)
[main::main()]                                   Points selected: (-0.90401,0) (-0.865011,0) (-0.842518,0) (-0.537244,0) (-0.398679,0) (-0.113835,0) (0.0799175,0) (0.380676,0) (0.591231,0)
(0.91713,0) (0.779767,0) (0.912919,0)
[main::main()]                                   Points selected: (-0.947619,0) (-0.928872,0) (-0.65895,0) (-0.584036,0) (-0.380337,0) (-0.133252,0) (0.148025,0) (0.468123,0) (0.50868,0) (0
.724906,0) (0.935442,0) (0.971511,0)
[main::main()]                                   Points selected: (-0.981875,0) (-0.859104,0) (-0.780963,0) (-0.587309,0) (-0.37501,0) (-0.228738,0) (0.170687,0) (0.320618,0) (0.554563,0) (
0.855494,0) (0.952401,0) (0.899179,0)
[main::main()]                                   Points selected: (-0.994747,0) (-0.936452,0) (-0.799706,0) (-0.502598,0) (-0.293855,0) (-0.238063,0) (0.0710795,0) (0.418727,0) (0.528205,0)
 (0.791017,0) (0.841999,0) (0.922319,0)
[main::main()]                                   Points selected: (-0.981901,0) (-0.942334,0) (-0.817244,0) (-0.502693,0) (-0.362869,0) (0.0531426,0) (0.188461,0) (0.378999,0) (0.517289,0)
(0.76008,0) (0.958332,0) (0.829141,0)
[main::main()]                                   Points selected: (-0.830801,0) (-0.895413,0) (-0.753161,0) (-0.533404,0) (-0.292988,0) (-0.116918,0) (0.126602,0) (0.398628,0) (0.602596,0)
(0.730918,0) (0.883785,0) (0.984913,0)
[main::main()]                                   Points selected: (-0.920836,0) (-0.91827,0) (-0.686716,0) (-0.599125,0) (-0.331364,0) (-0.143183,0) (0.14999,0) (0.304703,0) (0.628909,0) (0
.78723,0) (0.821294,0) (0.927566,0)
[main::main()]                                   Points selected: (-0.939713,0) (-0.896893,0) (-0.773103,0) (-0.538324,0) (-0.32484,0) (-0.0845858,0) (0.0429227,0) (0.300719,0) (0.535192,0)
 (0.759677,0) (0.940532,0) (0.958998,0)
[main::main()]                                   Points selected: (-0.912195,0) (-0.928642,0) (-0.786102,0) (-0.560542,0) (-0.356796,0) (-0.123163,0) (0.118005,0) (0.335118,0) (0.571327,0)
```

- Generate large collection of Z points (≈ 50).

Use Padé Approximants

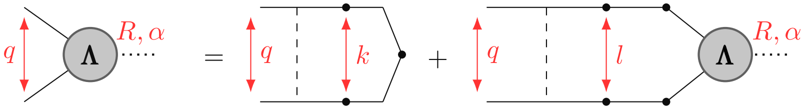
- In this case we know the analytic dependence – can improve extrapolation
- Uncertainty **massively** reduced – automate search for best parameters
  - Use statistics to “approximate” uncertainty



- Print the results as $\overline{f}(\alpha) \pm \sigma_f(\alpha)$.

Reminder:

- We actually want to generalize the LF projection – can then be attached to whatever diagram



- LF projection happens in inhomogeneous term – need to actually understand this.

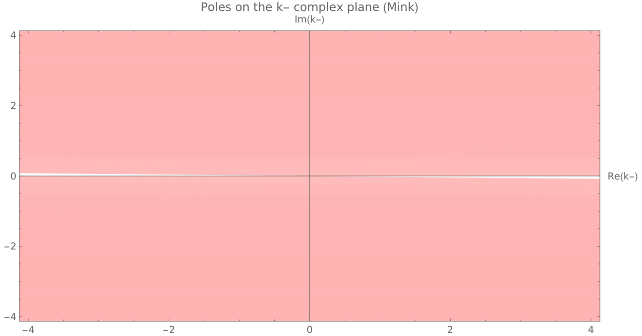
Vertex equation

$$\mathbf{\Lambda}(q,R,\alpha) = \int dk^{-}\; \mathbf{K}(q,k)\mathbf{G_0}(k) + \int d^4l\; \mathbf{K}(q,l)\mathbf{G_0}(l)\mathbf{\Lambda}(l,R,\alpha)$$

(Bednar, Cloët, Tandy; 2020)

Residues

- Can plot the poles in complex plane
- **They move too much**



- Problems for small β – also appear in previous results
- Need to “deform” the Wick rotation
  - What is a good deformation?
  - What does this look in practical caclulations like?