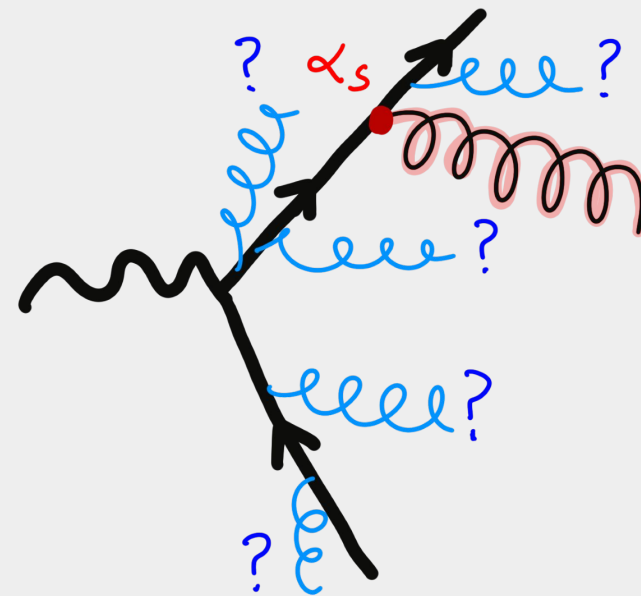


Resummed Distribution Functions: Unitary Completions of Fixed-Order Perturbation Theory



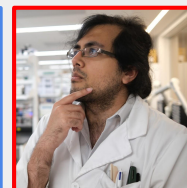
QCD@LHC

Rikab Gambhir

gambhirb@ucmail.uc.edu

In Collaboration with: Radha Mastandrea

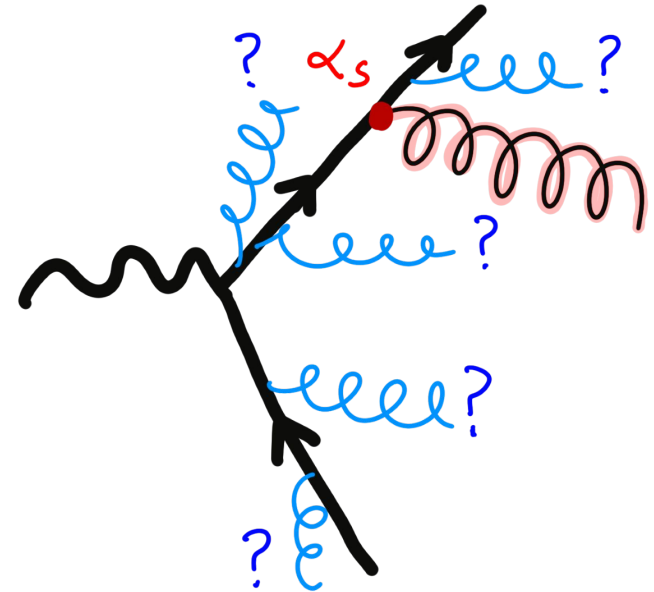
Try our repo!



Basic Premise

FACT 1: Rikab is not very good at hardcore QCD calculations. Resummation is *hard*!

Large logs? IR Divergences?
Resummation? SCET? Plus
functions? Sudakov Shoulders?
Multiple scales? Too much for me!
Can I just guess?



Goal:

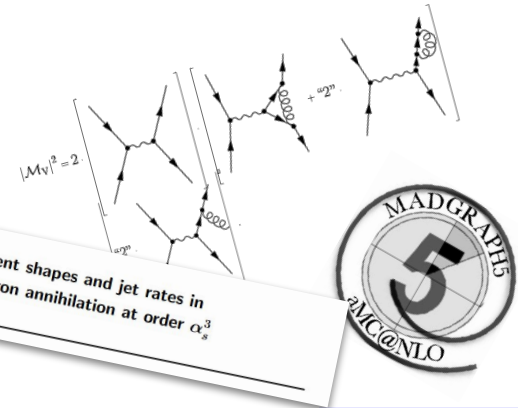
I want to see how far I can get predicting distributions for QCD observables anyways! *Including uncertainty, and as systematically and rigorously as possible!*

Basic Premise

FACT 1: Rikab is not very good at hardcore QCD calculations. Resummation is *hard*!

Large logs? IR Divergences?
Resummation? SCET? Plus
functions? Sudakov Shoulders?
Multiple scales? Too much for me!
Can I just guess?

Suppose I can calculate
the *fixed-order*
distribution for
observable(s) x :



EERAD3: Event shapes and jet rates in
electron-positron annihilation at order α_s^3

$$p(x|\alpha) = p_0(x) + \alpha p_1(x|\alpha) + \alpha^2 p_2(x|\alpha) + \dots + \alpha^M p_M(x|\alpha) + \mathcal{O}(\alpha^{M+1})$$

What I know: Fixed-order calculation

Unknown



FACT 2: He at least knows what a probability distribution *should* look like.

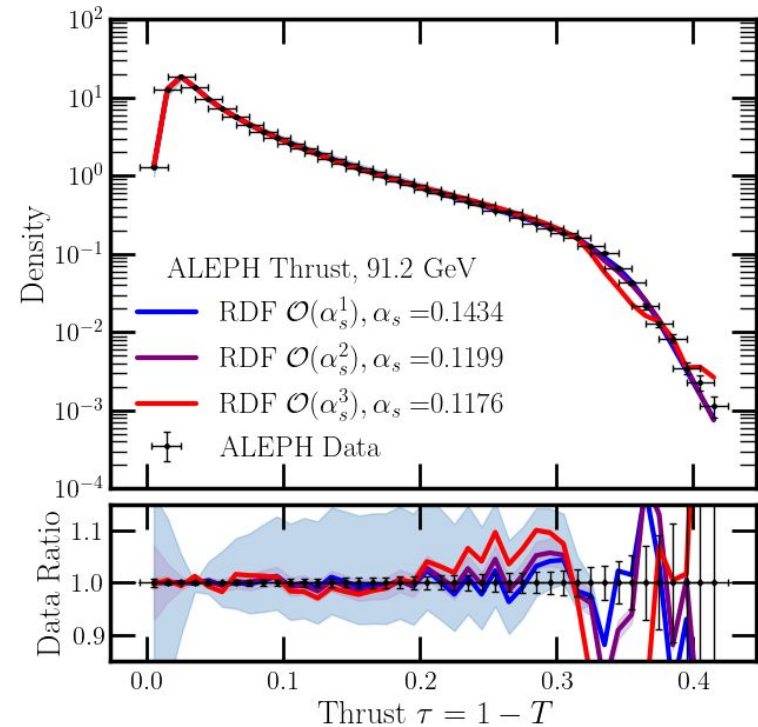
How far can we get with just *guessing* and enforcing consistency conditions?
What is the full distribution even *allowed* to be?

Let's parameterize it: The **Resummed Distribution Function**

What we are going to do

1. We will make *all-orders* statements about QCD, using only fixed-order structure and consistency conditions, *and no other knowledge!*
2. We will build a framework – the **Resummed Distribution Function (RDF)** – for encoding all possible higher-order terms, which is particularly useful for *defining theory uncertainties*.
3. We will use the RDF and above theory uncertainties to perform fits to ALEPH data – where we will get *reasonable values for α* using *only fixed order calculations* up to $\mathcal{O}(\alpha^3)$.

We don't have to know *anything* about resummation, logarithms, SCET etc, for this to work.



Order	RDF Prior σ		
	$\sigma = 0.5$	$\sigma = 1.0$	$\sigma = 2.0$
Thrust $\mathcal{O}(\alpha_s^1)$	$0.1555^{+0.0189}_{-0.0139}$	$0.1434^{+0.0409}_{-0.0212}$	$*0.19^{+0.0000}_{-0.0595}$
Thrust $\mathcal{O}(\alpha_s^2)$	$0.1202^{+0.0010}_{-0.0009}$	$0.1199^{+0.0019}_{-0.0020}$	$0.1249^{+0.0067}_{-0.0053}$
Thrust $\mathcal{O}(\alpha_s^3)$	$0.1202^{+0.0002}_{-0.0001}$	$0.1176^{+0.0002}_{-0.0002}$	$0.1164^{+0.0003}_{-0.0002}$

Basic Premise – Continued

$$p(x|\alpha) = p_0(x) + \alpha p_1(x|\alpha) + \alpha^2 p_2(x|\alpha) + \dots + \alpha^M p_M(x|\alpha) + \mathcal{O}(\alpha^{M+1})$$

Known Fixed-order calculation

Unknown



Goal: Use consistency conditions, *and nothing else*, to say something about this *with uncertainty!*

Consistency Conditions

The full, all-orders distribution (nature) should be:

1. Positive
2. Finite e.g. no “large logs”
3. Normalized
4. Smooth*

A fixed-order calculation is typically **none** of these things. Let's enforce it!

Basic Premise – Continued

$$p(x|\alpha) = p_0(x) + \alpha p_1(x|\alpha) + \alpha^2 p_2(x|\alpha) + \dots + \alpha^M p_M(x|\alpha) + \mathcal{O}(\alpha^{M+1})$$

Known Fixed-order calculation

Unknown



Consistency Conditions

The full, all-orders distribution (nature) should be:

1. Positive
2. Finite e.g. no “large logs”
3. Normalized
4. Smooth*

A fixed-order calculation is typically **none** of these things. Let's enforce it!

Goal: Use consistency conditions, *and nothing else*, to say *something* about this *with uncertainty!*

We introduce the **Resummed Distribution Function (RDF) q** :

$$q(t|\alpha) = f(t, \alpha) \cdot \exp \left[- \int_0^t dt' f(t', \alpha) \right]$$

Explained Shortly!

Which parameterizes everything the **unknown** terms *could ever be*, subject to the consistency conditions!

This also provides a framework for fitting and perturbative theory uncertainties.

Basic Premise – Spelled Out

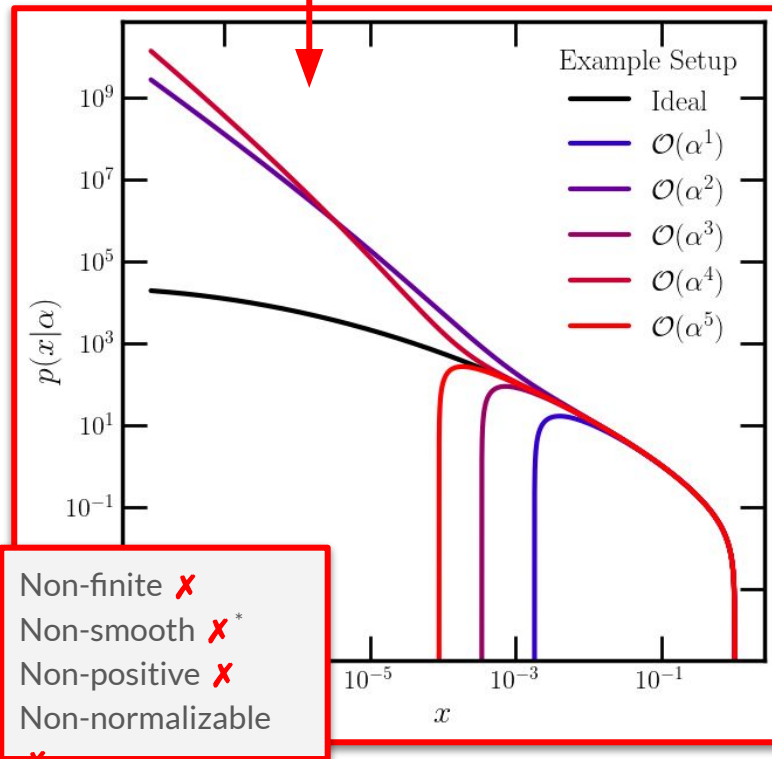
$$p(x|\alpha) = p_0(x) + \alpha p_1(x|\alpha) + \alpha^2 p_2(x|\alpha) + \dots + \alpha^M p_M(x|\alpha) + \mathcal{O}(\alpha^{M+1})$$

Known Fixed-order calculation

Unknown

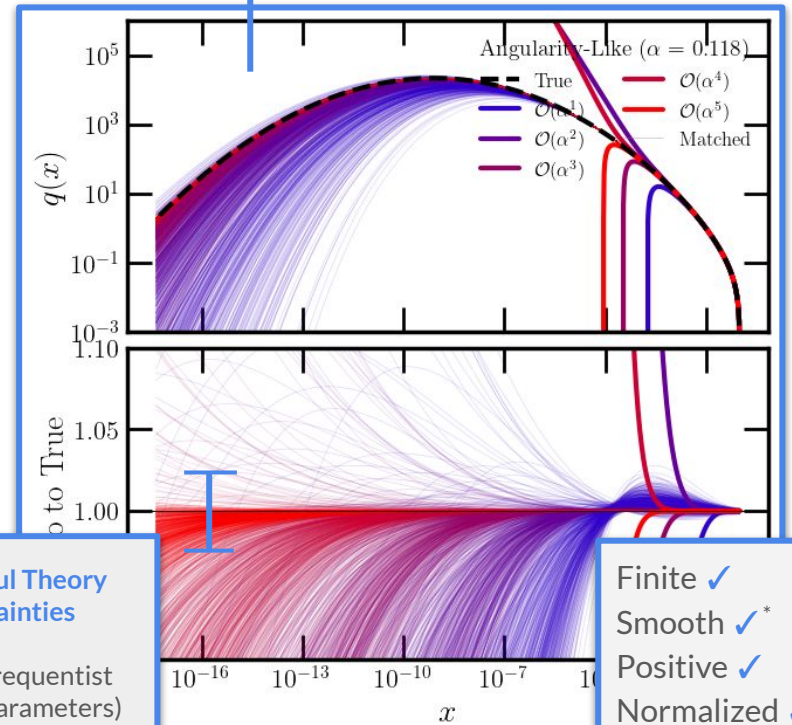
You give us this

We give you back this



Fixed-order distribution p in α for an observable x

Matching
To RDF



All non-pathological all-orders distributions q that match p to the given order

Constructing the **RDF**

Our goal is to parameterize the probability of measuring an observable x conditioned on a perturbative parameter α

Consistency Conditions

1. Positive
2. Finite e.g. no “large logs”
3. Normalized
4. Smooth

Perturbativity Condition

5. Analytic in α^*

Parameterize *all* such distributions!

The RDF (univariate)

Claim: Any distribution of the form

$$q(t|\alpha) = f(t, \alpha) \cdot \exp \left[- \int_0^t dt' f(t', \alpha) \right]$$

where:

- $f(t, \alpha) = \exp[-g(t, \alpha)]$
- The function g is bounded from above in t .
- The function g is analytic in α up to single logs.

$$t \equiv \log\left(\frac{1}{x}\right)$$

satisfies conditions **(1)-(5)** for any function g .

By parameterizing g , we parameterize all possible all-orders distributions!

Parametrizing the RDF

Our goal is to parameterize the probability of measuring an observable x conditioned on a perturbative parameter α

Choosing the g function

We may generically write:

$$g(t, \alpha) = -\log(g^*(t, \alpha)) + g_{\text{Analytic}}(t, \alpha)$$

positive and analytic

analytic

Each g function is just a power series in α .

We are free to choose *any*^{*} t dependence we like:
polynomials, functional bases, neural networks...

Claim: Any distribution of the form

$$q(t|\alpha) = f(t, \alpha) \cdot \exp \left[- \int_0^t dt' f(t', \alpha) \right]$$

where:

- $f(t, \alpha) = \exp[-g(t, \alpha)]$
- The function g is bounded from above in t .
- The function g is analytic in α up to single logs.

$$t \equiv \log\left(\frac{1}{x}\right)$$

For an observable x from 0 to 1, we transform to t from 0 to ∞

$$t \equiv \log\left(\frac{1}{x}\right)$$

This is where our QCD logs come from!

Parametrizing the RDF

Our goal is to parameterize the probability of measuring an observable x conditioned on a perturbative parameter α

Choosing the g function

We may generically write:

$$g(t, \alpha) = -\log(g^*(t, \alpha)) + g_{\text{Analytic}}(t, \alpha)$$

positive and analytic

analytic

Each g function is just a power series in α .

We are free to choose *any*^{*} t dependence we like:
polynomials, functional bases, neural networks...

Our parameterization: **Polynomials**

We choose the simplest thing:

Kinematic Thresholds

$$g^*(t, \alpha), g_{\text{Analytic}}(t, \alpha) = \sum_{m=0, n=0}^{M, N} g_{mn} \alpha^m t^n \Theta(t - \theta_{mn})$$

Where each g_{mn} is a parameter matrix:

$$g_{mn} = \begin{matrix} & \xrightarrow{\text{Powers of logs } t} & \\ \begin{pmatrix} g_{00} & g_{01} & g_{02} & g_{03} & g_{04} & g_{05} & g_{06} & g_{07} & g_{08} & g_{09} \\ g_{10} & g_{11} & g_{12} & g_{13} & g_{14} & g_{15} & g_{16} & g_{17} & g_{18} & g_{19} \\ g_{20} & g_{21} & g_{22} & g_{23} & g_{24} & g_{25} & g_{26} & g_{27} & g_{28} & g_{29} \\ g_{30} & g_{31} & g_{32} & g_{33} & g_{34} & g_{35} & g_{36} & g_{37} & g_{38} & g_{39} \\ g_{40} & g_{41} & g_{42} & g_{43} & g_{44} & g_{45} & g_{46} & g_{47} & g_{48} & g_{49} \end{pmatrix} & \downarrow \text{Orders of } \alpha \end{matrix}$$

Known

Unknown

*Need $g_{0N} < 0$ to be bounded from above.

In many analytic cases, we have access to an exact $g_m(t)$ and therefore don't need it to be a polynomial in t

Matching and Perturbation Theory

Given a fixed order calculation, which *should* be a distribution:

$$p(x|\alpha) = p_0(x) + \alpha p_1(x|\alpha) + \alpha^2 p_2(x|\alpha) + \dots + \alpha^M p_M(x|\alpha) + \mathcal{O}(\alpha^{M+1})$$

What I know: Fixed-order calculation

Unknown

We want to inject our **theory knowledge** exactly into the RDF g parameters, so that the **free unknown coefficients** of the RDF will parameterize our ignorance about higher-order physics.

Matching

Powers of logs t									
g_{00}	g_{01}	g_{02}	g_{03}	g_{04}	g_{05}	g_{06}	g_{07}	g_{08}	g_{09}
g_{10}	g_{11}	g_{12}	g_{13}	g_{14}	g_{15}	g_{16}	g_{17}	g_{18}	g_{19}
g_{20}	g_{21}	g_{22}	g_{23}	g_{24}	g_{25}	g_{26}	g_{27}	g_{28}	g_{29}
g_{30}	g_{31}	g_{32}	g_{33}	g_{34}	g_{35}	g_{36}	g_{37}	g_{38}	g_{39}
g_{40}	g_{41}	g_{42}	g_{43}	g_{44}	g_{45}	g_{46}	g_{47}	g_{48}	g_{49}

$g_{mn} =$

Known

Unknown

Orders of α

Matching and Perturbation Theory [Details]

Given a fixed order calculation, which *should* be a distribution:

$$p(x|\alpha) = p_0(x) + \alpha p_1(x|\alpha) + \alpha^2 p_2(x|\alpha) + \dots + \alpha^M p_M(x|\alpha) + \mathcal{O}(\alpha^{M+1})$$

What I know: Fixed-order calculation

Unknown

Analytic Matching Algorithm

$$g(t, \alpha) = -\log(p_{m^*}(t)\alpha^{m^*}) + \left[\sum_{k \geq 1} \frac{1}{k} \left(\left[-\sum_{m > m^*}^M \frac{p_m(t)}{p_{m^*}(t)} \alpha^{m-m^*} \right]^k - \left[\sum_{m=m^*}^{M-m^*} \int_0^t dt' p_m(t') \alpha^m \right]^k \right) + \mathcal{O}(\alpha^{M+1-m^*}) \right]$$

Completely fixes the $\mathcal{O}(\alpha^M)$ part of g !

Matching

Implementation

Looks scary, but autodiff magic makes this programmatically easy for a python function $p(t, \alpha)$

Try our Repo!



Powers of logs t									
g_{00}	g_{01}	g_{02}	g_{03}	g_{04}	g_{05}	g_{06}	g_{07}	g_{08}	g_{09}
g_{10}	g_{11}	g_{12}	g_{13}	g_{14}	g_{15}	g_{16}	g_{17}	g_{18}	g_{19}
g_{20}	g_{21}	g_{22}	g_{23}	g_{24}	g_{25}	g_{26}	g_{27}	g_{28}	g_{29}
g_{30}	g_{31}	g_{32}	g_{33}	g_{34}	g_{35}	g_{36}	g_{37}	g_{38}	g_{39}
g_{40}	g_{41}	g_{42}	g_{43}	g_{44}	g_{45}	g_{46}	g_{47}	g_{48}	g_{49}

Known

Unknown

Orders of α

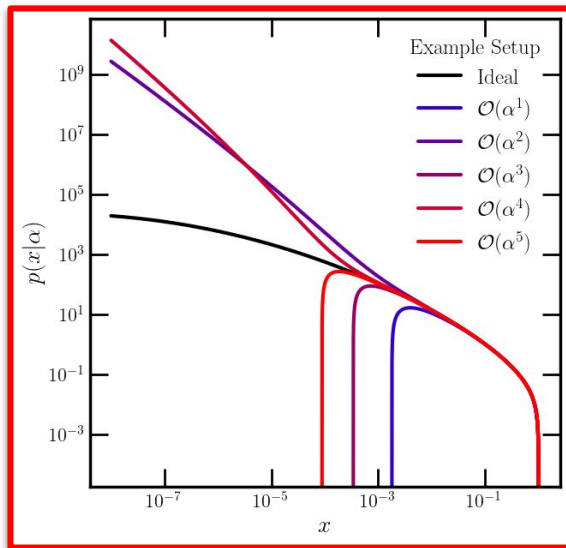
Upshot: Fixed-Order to All-Orders

Given a fixed order calculation, which *should* be a distribution:

$$p(x|\alpha) = p_0(x) + \alpha p_1(x|\alpha) + \alpha^2 p_2(x|\alpha) + \dots + \alpha^M p_M(x|\alpha) + \mathcal{O}(\alpha^{M+1})$$

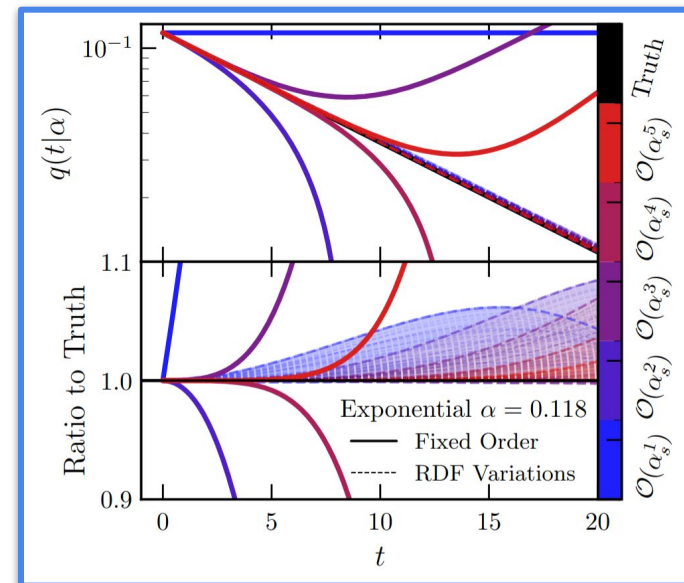
What I know: Fixed-order calculation

Unknown



The RDF lets us encode fixed-order physics we do know, and parameterize high-order physics we don't know

All while being a proper probability distribution and preserving perturbative structure exactly!



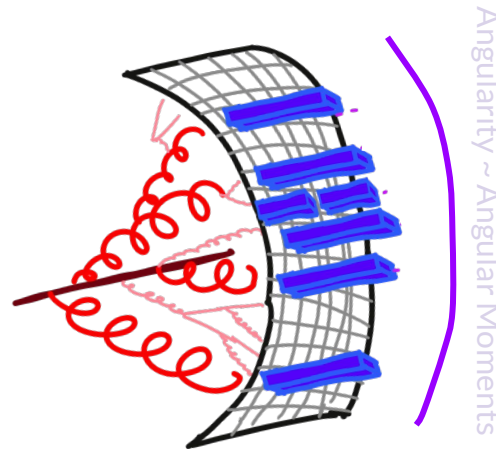
$$q^{(\text{Exp}) (M)}(t|\alpha) = g^*(t, \alpha) \exp \left[-g_{\text{Analytic}}(t, \alpha) - \int_0^t dt' g^*(t', \alpha) \exp(-g_{\text{Analytic}}(t', \alpha)) \right]$$

$$= \alpha \exp(-\alpha t) + \mathcal{O}(\alpha^{M+1}).$$

QCD-Like Toy: Pseudo-Angularity

Indulge me a game: Let's pretend the world is described by *Leading Log QCD*, but we didn't know that.

We'll predict *jet angularities* x , which (roughly) describe how wide a QCD jet is in terms of angular moments.



Annoying notation: $x = \lambda$ for angularities

$$\lambda^{(\beta)} = \sum_i z_i \left(\frac{\theta_i}{R} \right)^\beta$$

R = Jet Radius

z_i = Particle energy fraction

θ_i = Distance to center

In this world, the true (unknown) distribution of angularities is:

$$p_{\text{Rayleigh}}(x|\alpha) = \alpha \frac{\log\left(\frac{1}{x}\right)}{x} e^{-\alpha \frac{\log^2\left(\frac{1}{x}\right)}{2}}$$

x space: Sudakov Double Log

$$p_{\text{Rayleigh}}(t|\alpha) = \alpha t e^{-\alpha \frac{t^2}{2}}$$

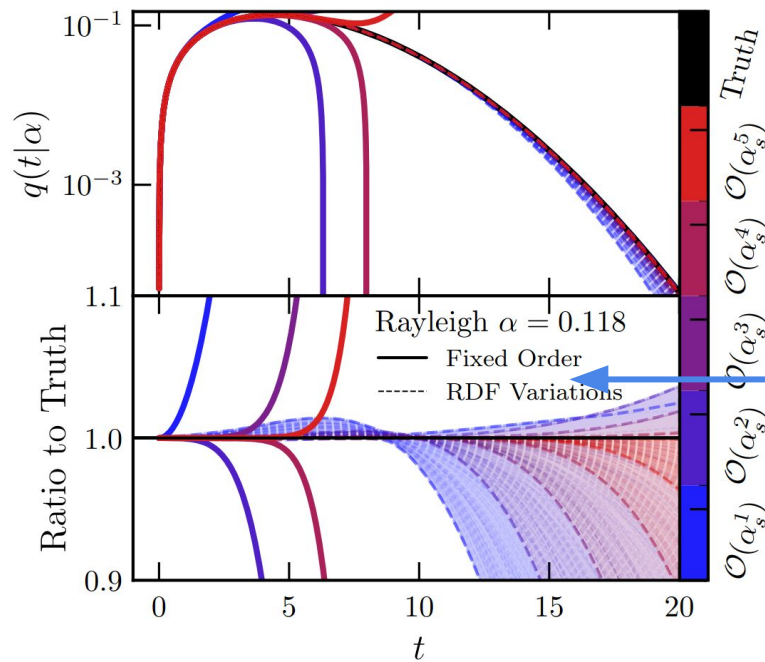
t space: Rayleigh Distribution

In this world, a fixed order QCD calculation would give us:

$$p_{\text{FO}}^{(\text{Ray})}(t|\alpha) = \alpha t \sum_{m=0}^{M-1} \frac{1}{m!} \left(-\alpha \frac{t^2}{2} \right)^m + \mathcal{O}(\alpha^{M+1})$$

QCD-Like Toy: Pseudo-Angularity

Indulge me a game: Let's pretend the world is described by *Leading Log QCD*, but we didn't know that.



The exact RDF we get is:

$$q^{(\text{Rayleigh})}(M)(t|\alpha) = \alpha t \exp\left(-\frac{\alpha t^2}{2}\right) + \mathcal{O}(\alpha^{M+1})$$

Unitarity is enough to fix the Sudakov factor at LL, exactly!

We are free to vary/guess higher-order g coefficients, corresponding to varying/guessing N^{nLL} structure.

The RDF gives well-behaved distributions from fixed order!

In this world, a fixed order QCD calculation would give us:

$$p_{\text{FO}}^{(\text{Ray})}(t|\alpha) = \alpha t \sum_{m=0}^{M-1} \frac{1}{m!} \left(-\alpha \frac{t^2}{2}\right)^m + \mathcal{O}(\alpha^{M+1})$$

Applying the RDF

As promised!

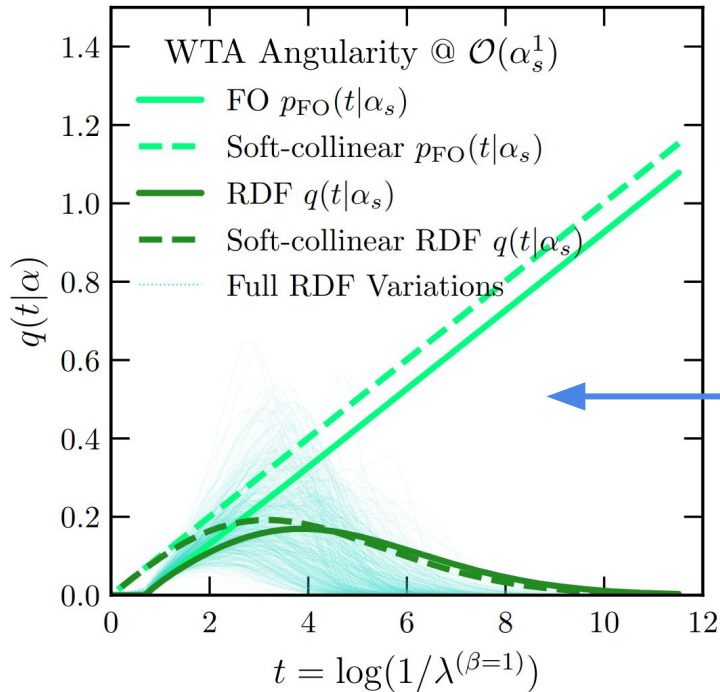
QCD Example: Real Angularities

The same game, but now with **real angularities** in real QCD, to $O(\alpha^1)$!

Technical details: “Winner-Take All” Angularities with angular exponent $\square$ in conical jets

The exact RDF we get is:

$$q(t|\alpha) = \left[\left(\frac{\alpha_s C_F}{\pi \beta} \right) \left(2t + 2 \log(1 - e^{-t}) + 3e^{-t} - \frac{3}{2} \right) \Theta(t - \log 2) \right] \\ \times \exp \left[- \left(\frac{\alpha_s C_F}{\pi \beta} \right) \left[t^2 - (\log 2)^2 + 2 (\text{Li}_2(e^{-t}) - \text{Li}_2(\frac{1}{2})) - 3(e^{-t} - \frac{1}{2}) - \frac{3}{2}(t - \log 2) \right] \right] \\ \text{(plus higher order } g \text{ not shown)}$$



Applying the RDF

$$p_{\text{FO}}(t) = \left(\frac{\alpha_s C_F}{\pi \beta} \right) \left(2t + 2 \log(1 - e^{-t}) + 3e^{-t} - \frac{3}{2} \right) \Theta(t - \log 2)$$

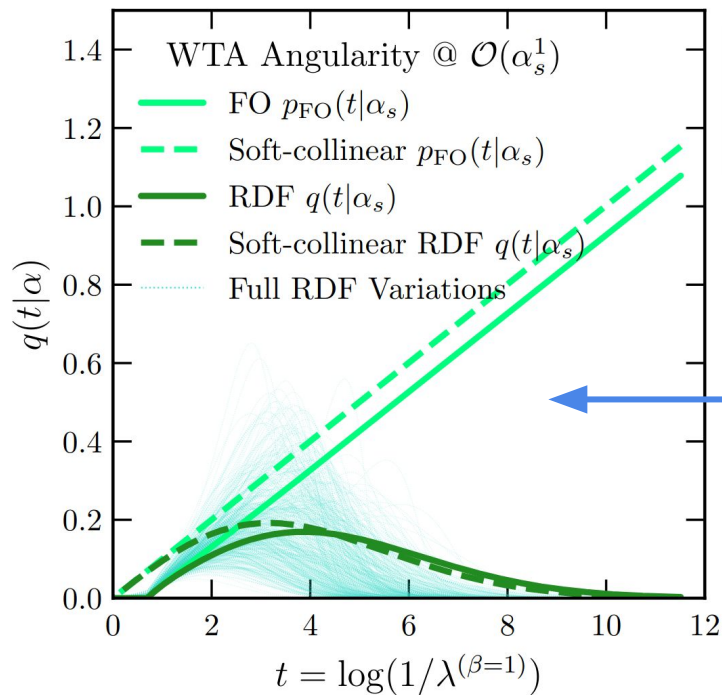
In our world, a fixed order QCD calculation gives us:

As promised!

QCD Example: Real Angularities

The same game, but now with **real angularities** in real QCD, to $O(\alpha^1)$!

Technical details: “Winner-Take All” Angularities with angular exponent $\square$ in conical jets



The exact RDF we get is:

$$q(t|\alpha) = \left[\left(\frac{\alpha_s C_F}{\pi \beta} \right) \left(2t + 2 \log(1 - e^{-t}) + 3e^{-t} - \frac{3}{2} \right) \Theta(t - \log 2) \right] \times \exp \left[- \left(\frac{\alpha_s C_F}{\pi \beta} \right) \left[t^2 - (\log 2)^2 + 2 \left(\text{Li}_2(e^{-t}) - \text{Li}_2\left(\frac{1}{2}\right) \right) - 3 \left(e^{-t} - \frac{1}{2} \right) - \frac{3}{2} (t - \log 2) \right] \right]$$

(plus higher order g not shown because it is long)

The story barely changes! We capture the LL limit for free as t goes to zero, and can capture higher-order uncertainty.

One new complication: **Kinematic thresholds**. Angularities don't saturate $x \rightarrow 1$ at fixed order, so the RDF is cut at $t > t_0$.

Technical details: Thresholds can be encoded as a $N^1\text{LL}$ shift of the LL logs.

Applying the RDF

$$p_{\text{FO}}(t) = \left(\frac{\alpha_s C_F}{\pi \beta} \right) \left(2t + 2 \log(1 - e^{-t}) + 3e^{-t} - \frac{3}{2} \right) \Theta(t - \log 2)$$

In our world, a fixed order QCD calculation gives us:

Numeric Matching: Beyond Leading Order

Beyond leading order in QCD, fixed-order analytic calculations are unavailable.

Instead, programs like **EERAD3** or Madgraph5 produce numerical predictions.

We'll use this to generate QCD thrust to $O(\alpha^3)$

Not a problem for the RDF! Let's just alter g to make it numerically robust, then we can match it to Monte-Carlo histograms at fixed order.

We fit for coefficients g_{mn}^* and $g_{\text{Analytic},mn}$, thresholds θ_m , and temperatures T_m

$$g(t, \alpha) = -\log \left(\sum_{m=m^*}^M \frac{\alpha^m}{m!} \left| \sum_{n=0}^N g_{mn}^* \frac{t^n}{n!} \Theta_{T_m}(t - \theta_m^*) \right|_{T_m} \right)$$

Smoothed-out absolute value to keep the log argument positive.

$$+ \sum_{m=0, n=0}^{M-m^*, N} g_{\text{Analytic},mn} \frac{t^n}{n!} \frac{\alpha^m}{m!} \Theta_{T_m}(t - \theta_{\text{Analytic},m})$$

Factorials for numeric scaling

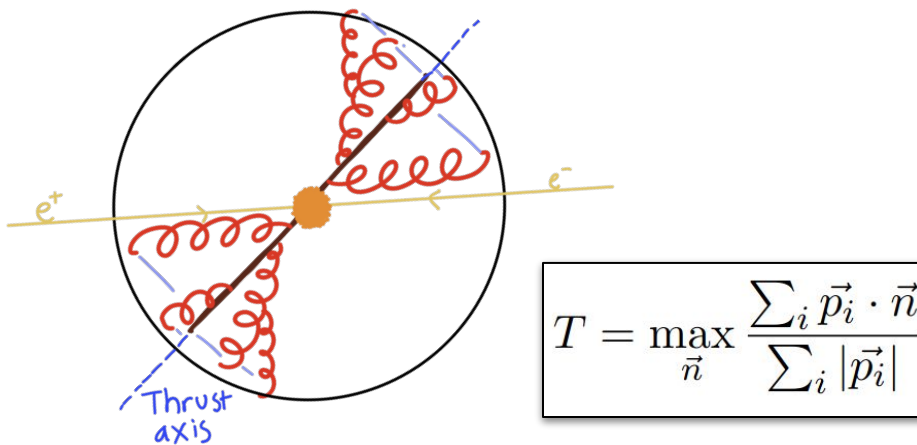
"Softmax" with a learned temperature to smooth thresholds

Fit to minimize the mean-squared error between the **Taylor expansion of q** and generated histograms for many α

Thrust and EERAD3

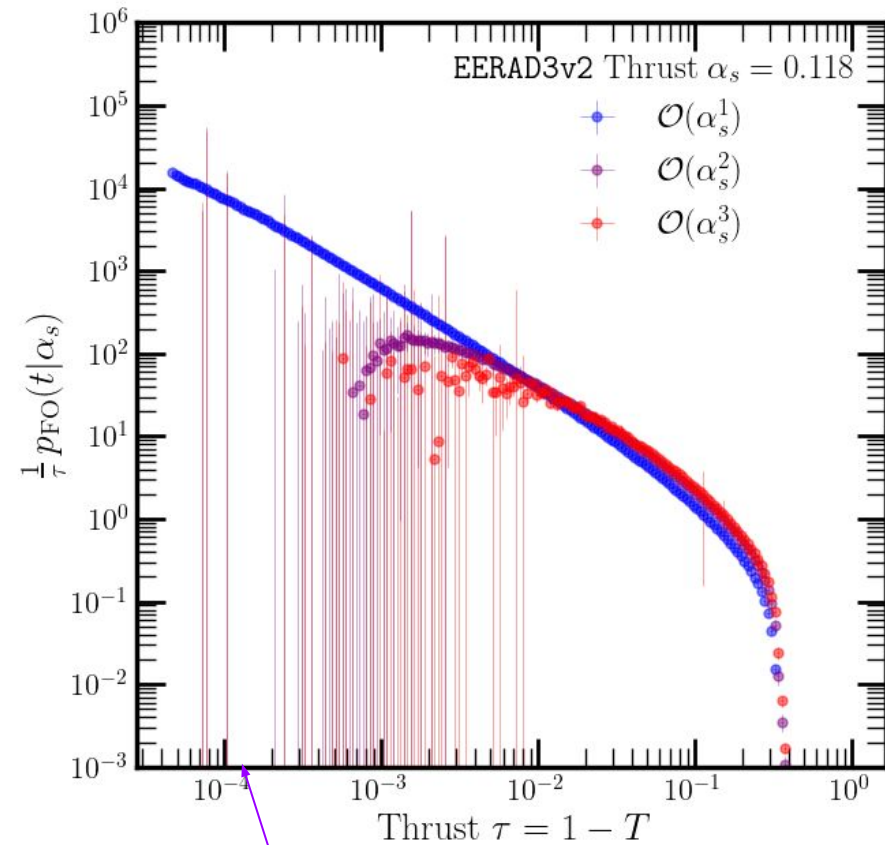
Thrust is a staple observable for precision QCD, studied to high order.

Roughly, how pencil-like is my event?



We use **EERAD3v2** to numerically calculate the fixed order distributions to $\mathcal{O}(\alpha^3)$. We get histograms of thrust for many different α values at each order.

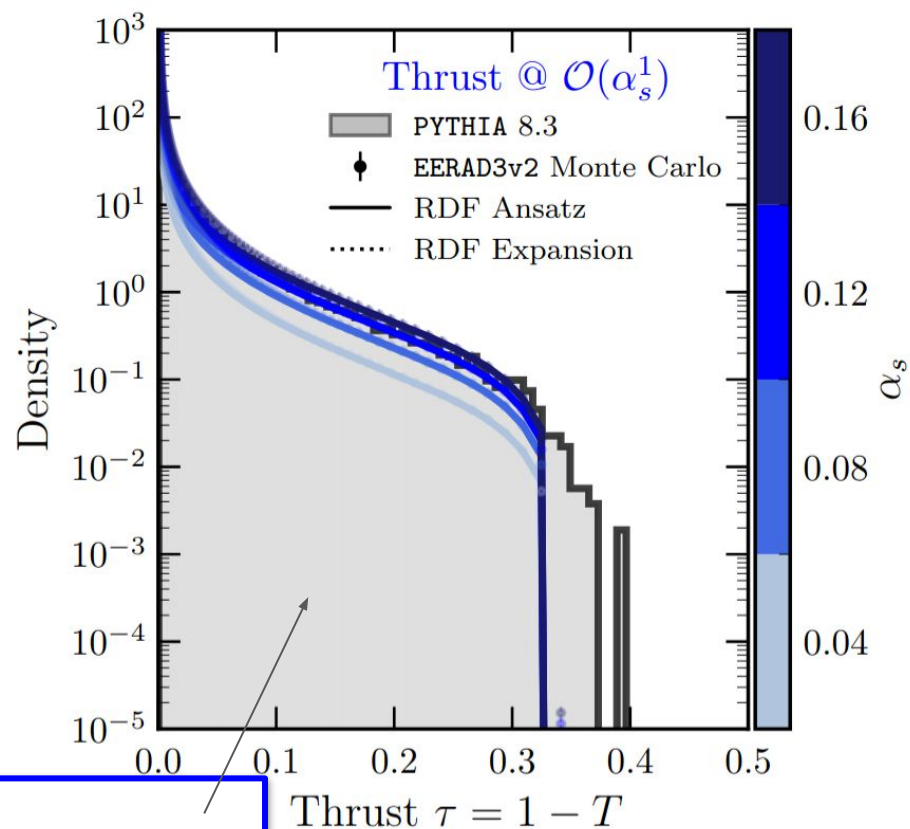
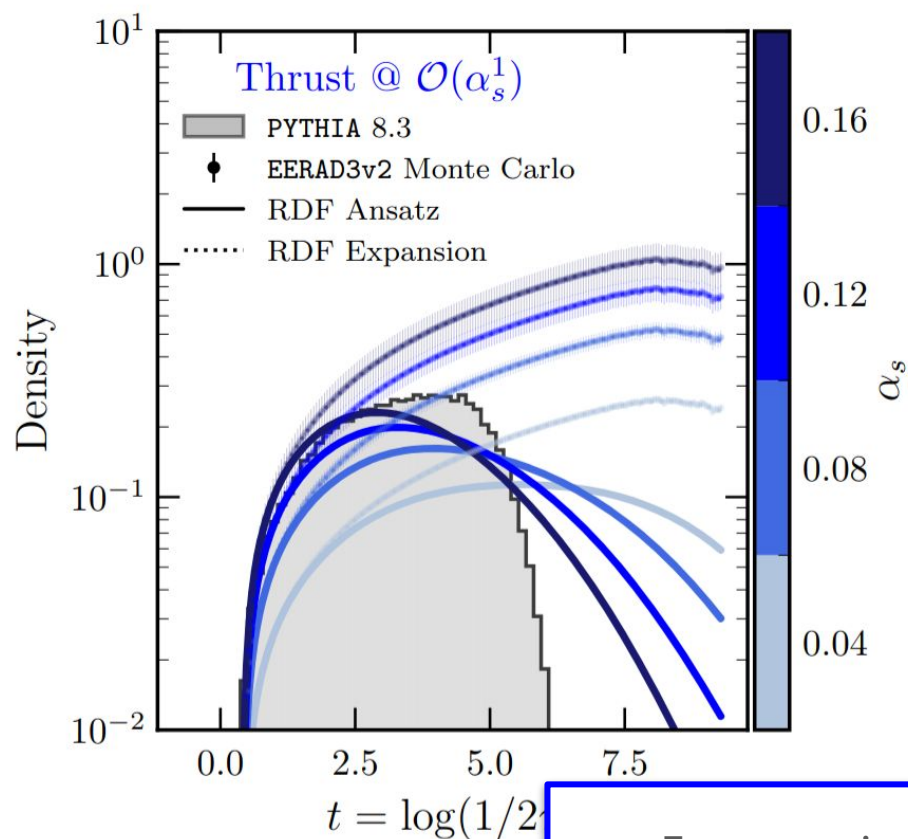
e^+e^- collisions at the Z pole



At $\mathcal{O}(\alpha^2)$, goes negative!

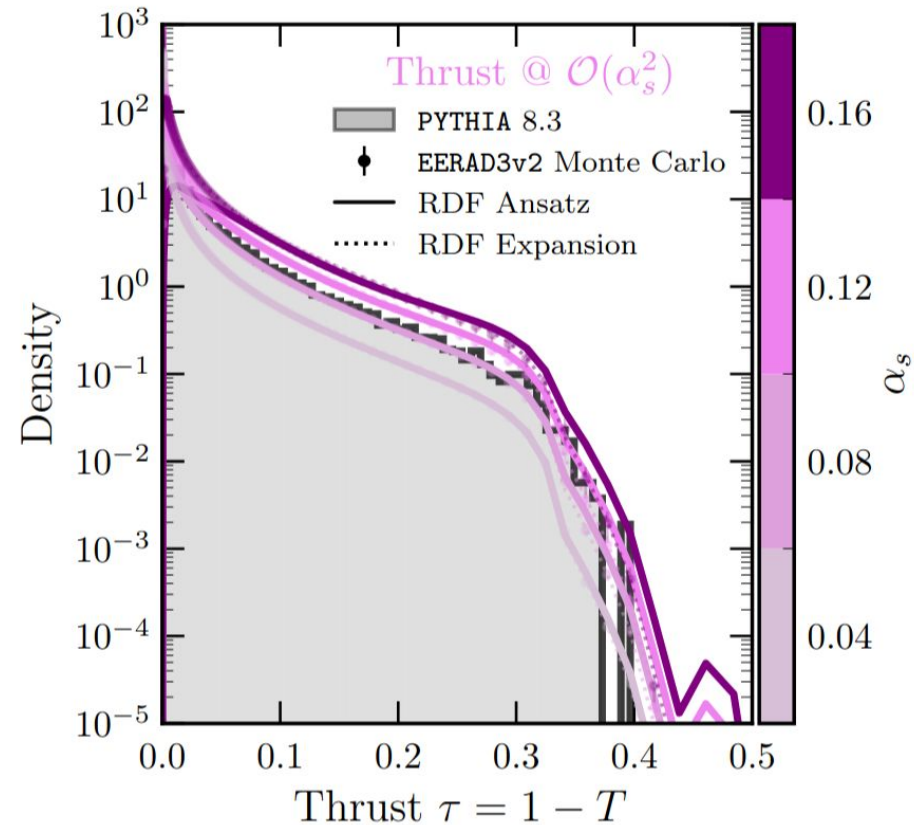
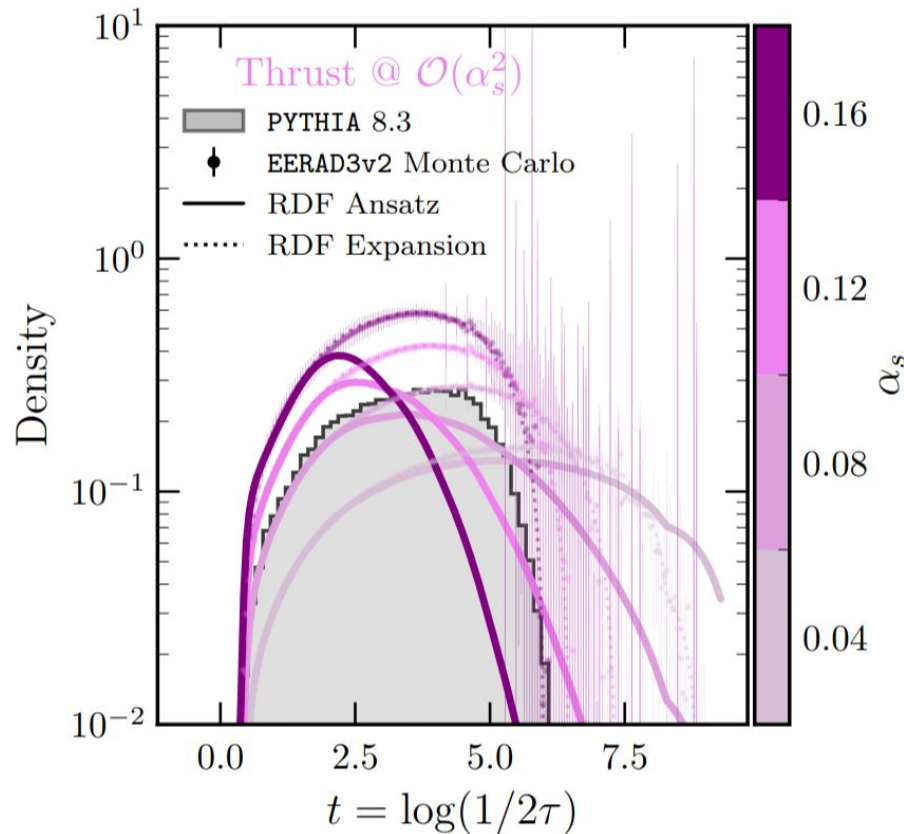
Matching Thrust to $\mathcal{O}(\alpha_s^1)$

Tune the RDF $m = 1$ parameters such that the RDF Expansion equals the Monte Carlo



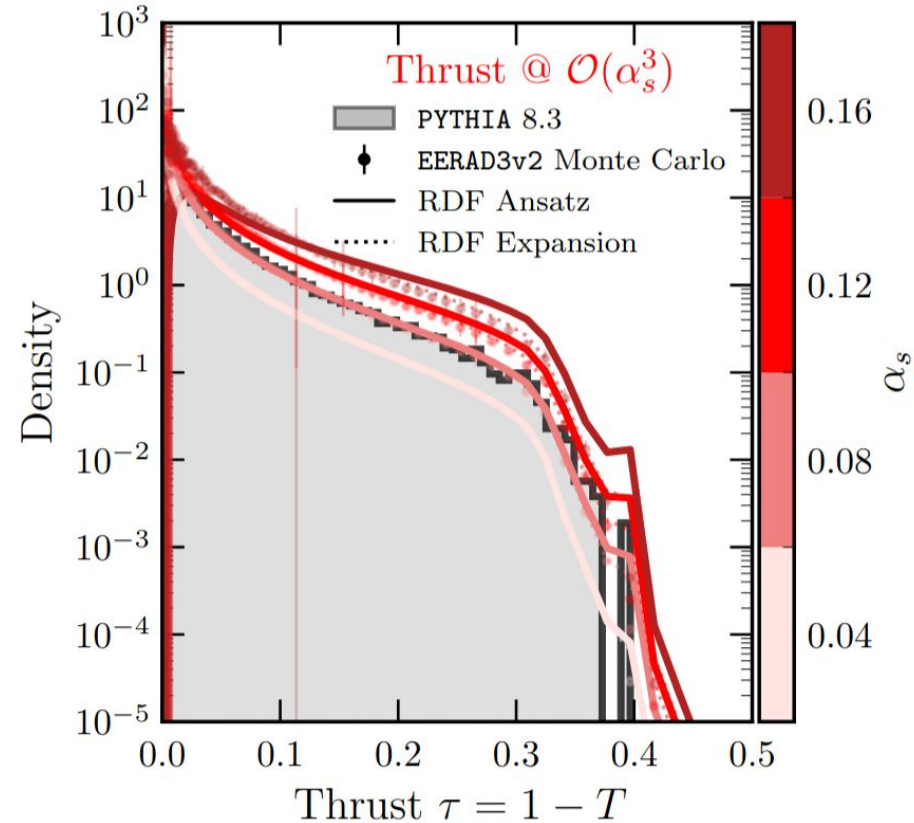
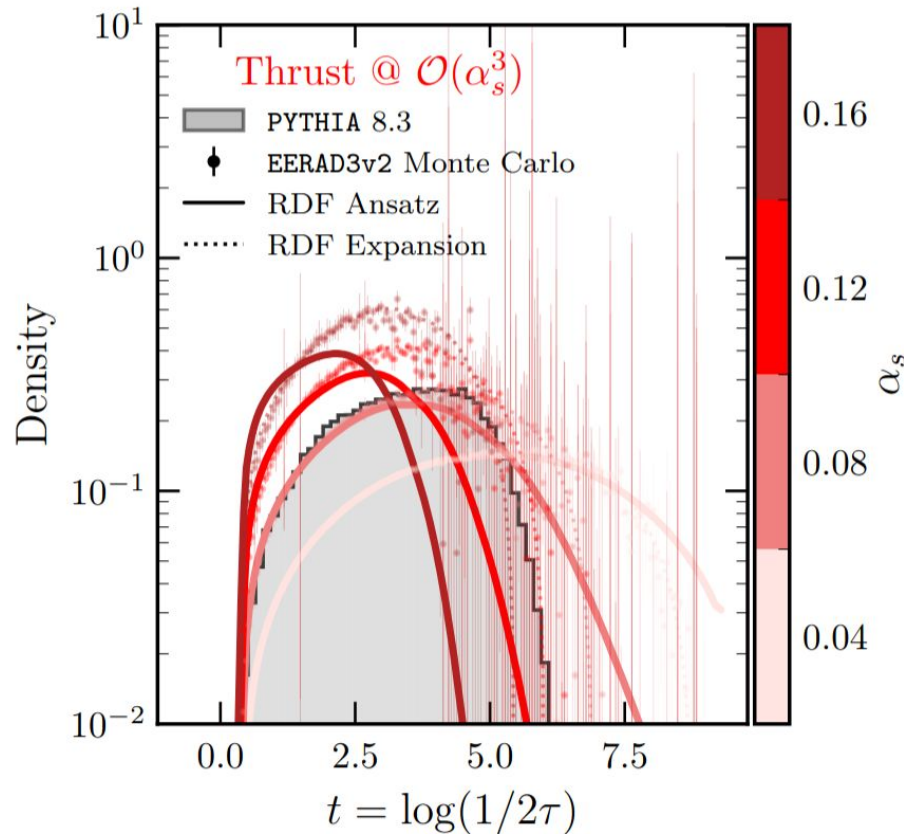
For comparison, PYTHIA
Leading Log Shower + some NLL
and nonperturbative effects

Matching Thrust to $\mathcal{O}(\alpha^2)$

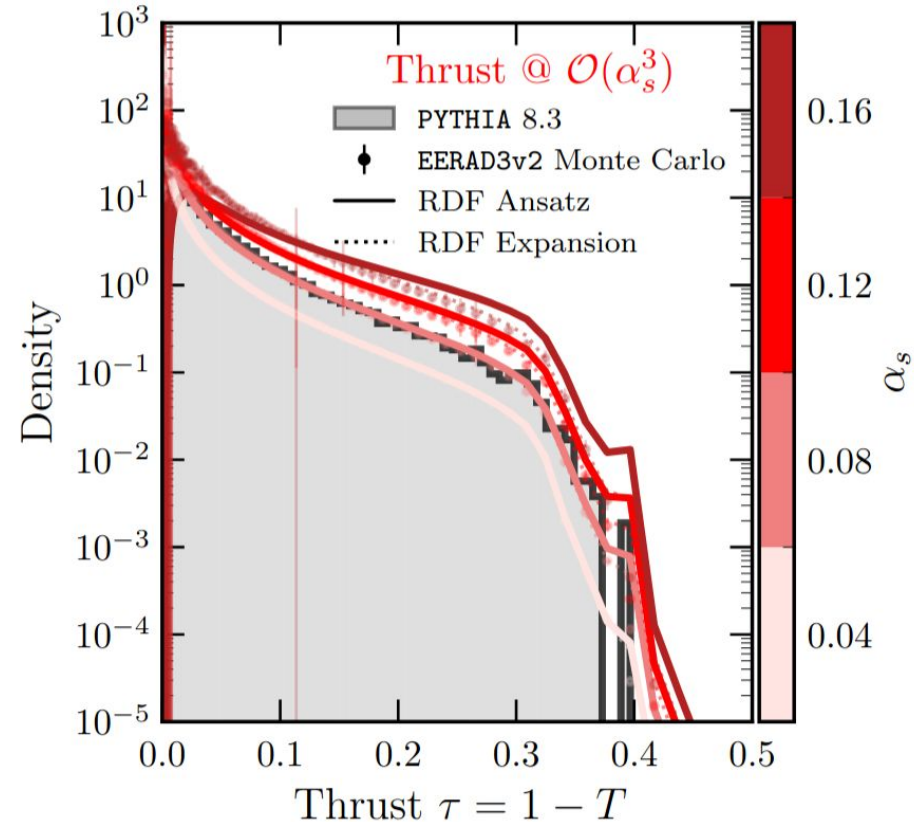
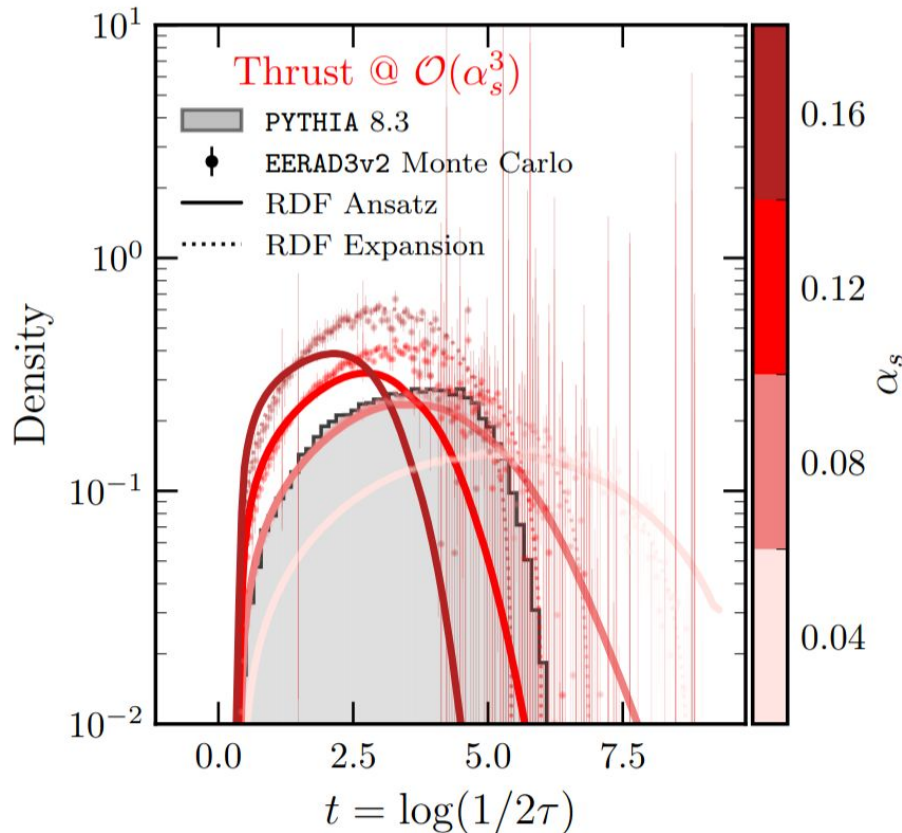


Iterative Matching: We freeze the $\mathcal{O}(\alpha^1)$

Matching Thrust to $\mathcal{O}(\alpha_s^3)$



Matching Thrust to $\mathcal{O}(\alpha_s^3)$



We'll keep these in our pocket for now and use this to fit to real data in a bit!

α_s Extraction from Data

Once we have our prediction for our observables t given α , let's compare it to data!

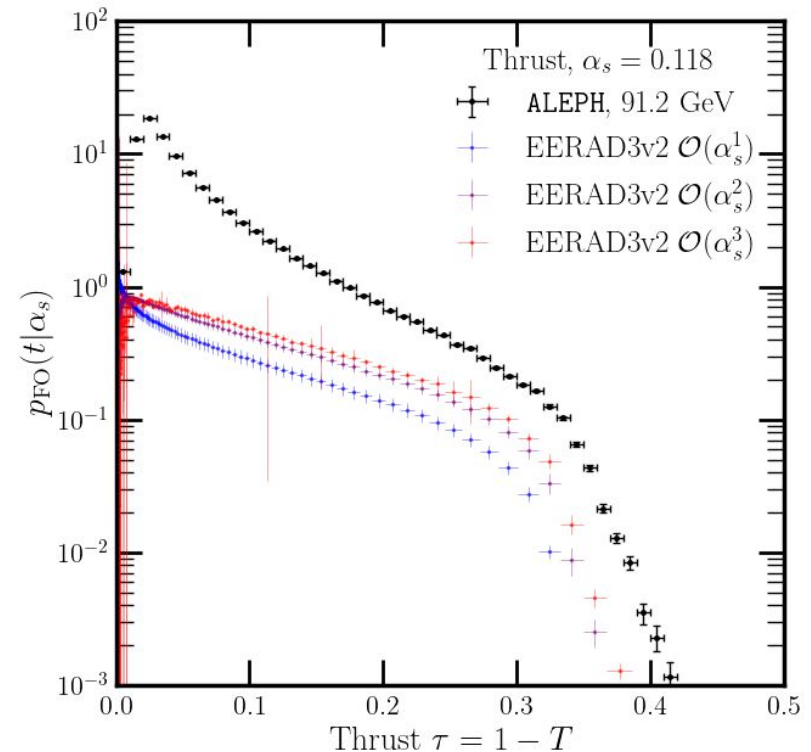
ALEPH (LEP) measurements for thrust in e^+e^- collisions at 91.2 GeV

Goal: given data t , find the α that maximizes $q(t | \alpha)$ — **Maximum Likelihood Estimation (MLE)**

The data is binned, so we will use a binned MLE. If we had access to event-level data, we could do an unbinned MLE, since the RDF is directly a valid likelihood!

Ill-defined MLE for fixed order calculations without windows or regulators.

ALEPH vs. Fixed Order Predictions



*I am a theorist and therefore I am going to completely ignore systematic uncertainties.

Theory Uncertainties and Nuisance Parameters

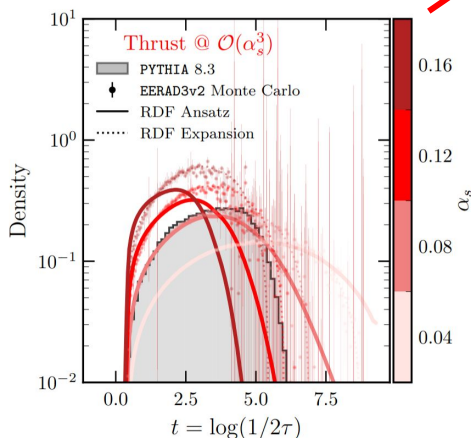
Theory uncertainties are about trying to **quantify what we don't know**.
But, we've already parameterized the physics we don't know with g coefficients!

Promote to **Nuisance Parameters**

The same split
we've been doing
this whole time

$$g(t, \alpha) = \underbrace{g_{\text{Matched}}(t, \alpha, \phi)}_{\mathcal{O}(\alpha_s^M)} + \underbrace{g_{\text{Nuisance}}(t, \alpha; \nu)}_{\mathcal{O}(\alpha_s^{M+1})}$$

Derived
from

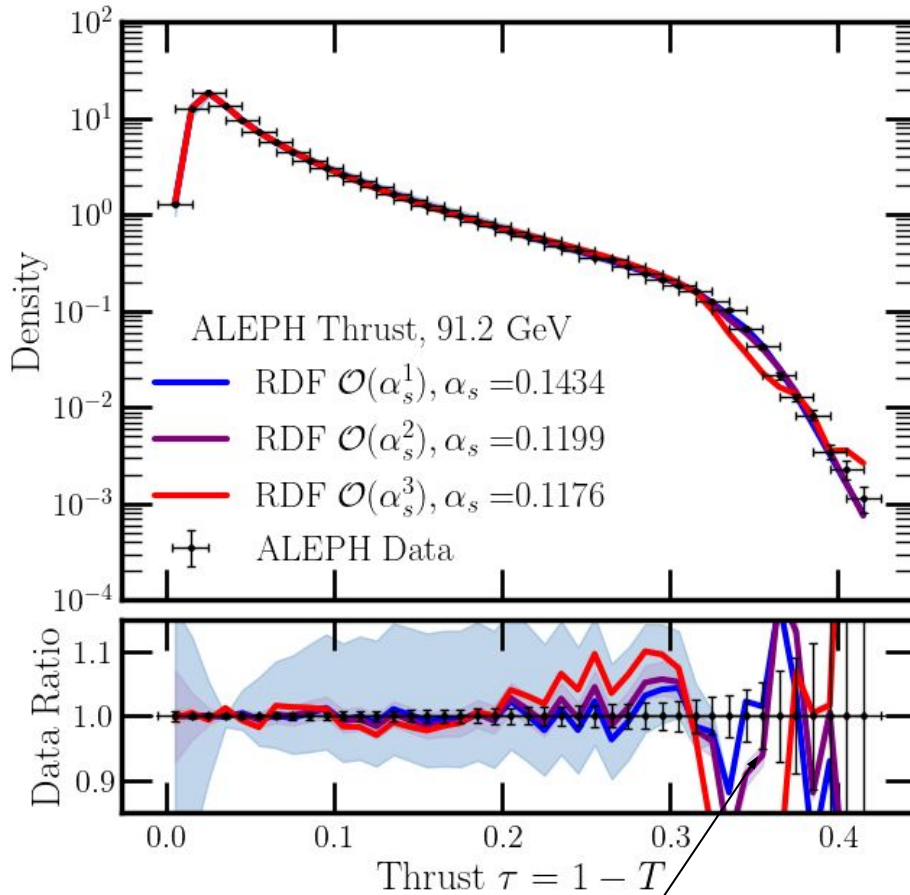


Then, just fit the profile likelihood to data:

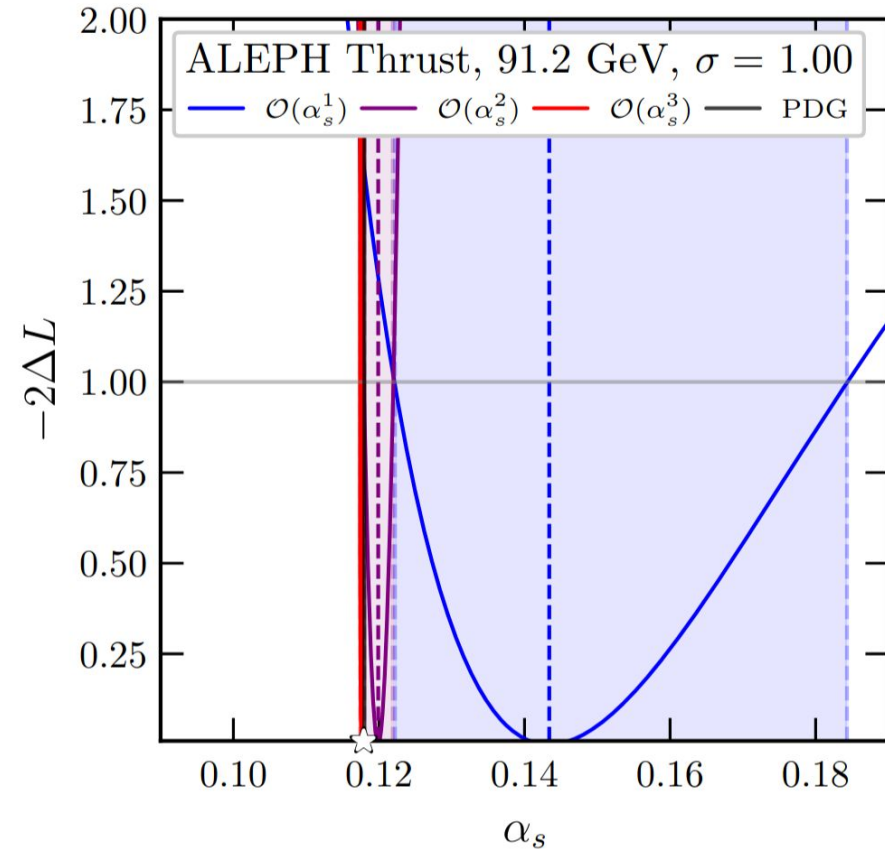
$$\log \mathcal{L}(\alpha) = \log \frac{\mathcal{L}(\alpha, \hat{\nu})}{\mathcal{L}(\hat{\alpha}, \hat{\nu})}$$

Emotionally: Statistical power gets “eaten up” fitting **nuisances**, leading to a “theory uncertainty” on α

Fitting Thrust



Kinematic thresholds are tricky to fit...



Beyond leading order, we recover the PDG value. **Victory!**

Final Results

Just to be thorough, let's try widening and narrowing our prior to see if it drastically changes things. *Pretty robust, at least beyond leading order!*

Order	<i>RDF Prior σ</i>			LEP I+II (From Ref. [40])
	$\sigma = 0.5$	$\sigma = 1.0$	$\sigma = 2.0$	
Thrust $\mathcal{O}(\alpha_s^1)$	$0.1555^{+0.0189}_{-0.0139}$	$0.1434^{+0.0409}_{-0.0212}$	$* 0.19^{+0.0000}_{-0.0595}$	0.1142 ± 0.0297
Thrust $\mathcal{O}(\alpha_s^2)$	$0.1202^{+0.0010}_{-0.0009}$	$0.1199^{+0.0019}_{-0.0020}$	$0.1249^{+0.0067}_{-0.0053}$	0.1152 ± 0.0068
Thrust $\mathcal{O}(\alpha_s^3)$	$0.1202^{+0.0002}_{-0.0001}$	$0.1176^{+0.0002}_{-0.0002}$	$0.1164^{+0.0003}_{-0.0002}$	0.1164 ± 0.0033

For comparison, a proper SCET calculation of thrust fit to the same data (but windowed) with scale variation uncertainties. Same ballpark!

At leading order, it seems there isn't enough information for us to reliably extract α .

*The third order errorbars are smaller than the SCET calculation. This is likely due to how we chose our prior to scale. A little fine tuning might help a lot!
Remember we have *not* considered systematic uncertainties or nonperturbative uncertainties here.

CLAIMER

What we are claiming

- **Claiming:** Given a distribution p , pathological or otherwise, we can find a regulated distribution q that is normalized, smooth, positive, and normalized that is perturbatively matched to p .
- **Claiming:** This is a universal approximator: *any* valid q can be reached with a choice of higher order g , including the “true” one or N^nLL calculations.
- **Claiming:** For *particular* g ’s, we can get meaningful q ’s that look like theory calculations and tame large logs!
- **Claiming:** Higher order g ’s can be used as meaningful nuisance parameters to define theory uncertainties and do fits!

DISCLAIMER

What we are *not* claiming

- **Not Claiming:** The perturbative structure is *easy* to maintain numerically or that numeric artifacts are manageable.
 - But if you restrict to reasonable choice or priors of g , like $g \sim N(0,1)$, then the results seem reasonable!
- **Not Claiming:** The envelopes give a *prior-free* uncertainty on higher orders.
 - The envelopes only tell you what *possible* higher order physics is consistent with positivity and normalizability — it *constrains* physics!
- **Not Claiming:** This is or beats N^nLL resummation.
 - But it is an all-orders resummation, and in special cases really is N^nLL !
 - For the RDF you don’t need to know *any* physics! Potentially useful for joint distributions
 - Even if you have a logarithmic resummation already, the RDF can still help you do theory uncertainties!

Key Point: This is *not* a shortcut to doing QCD, just a new way to parameterize things to guarantee nice properties. There is no free lunch, you still have to do physics!

Conclusion

Try our repo!



Email us questions!
gambhirb@ucmail.uc.edu
rmastand@uchicago.edu
[RG, Mastandrea; 2512.04160]



We introduced the Resummed Distribution Function (RDF) as a method to parameterize non-pathological distributions, that splits **physics we know** from **physics we don't** (high order)

$$p(x|\alpha) = p_0(x) + \alpha p_1(x|\alpha) + \alpha^2 p_2(x|\alpha) + \dots + \alpha^M p_M(x|\alpha) + \mathcal{O}(\alpha^{M+1})$$

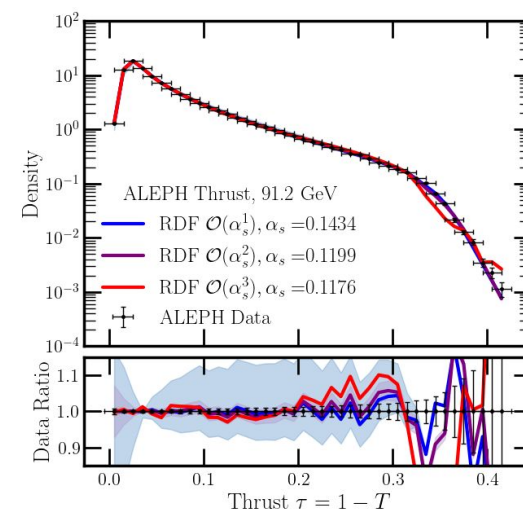
What I know: Fixed-order calculation

Unknown

It is a completely agnostic, novel, and easy way to convert a fixed-order distribution to a unitary resummation, containing all possible all-orders completions

The RDF provides a natural and rigorous way to place theory uncertainties on perturbative calculations!

If you have a fixed-order calculation, zero reason not to apply the RDF to explore higher orders for free!



Appendices

Numerics and Fitting

Bogosort

Article Talk

From Wikipedia, the free encyclopedia

In computer science, **bogosort**^{[1][2]} (also known as **permutation sort** and **stupid sort**^[3]) is a sorting algorithm based on the generate and test paradigm. The

We use a “**0-1-2**” strategy for numeric minimization, based on how many derivatives are considered:

0th Order: “BOGO-init” We randomly initialize all parameters N times. We select the lowest loss. Then we randomly perturb each of those parameters again with lower variance, and repeat many times. *Not too dissimilar from Hamiltonian Monte Carlo*

1st Order: Gradient Descent We use ordinary gradient descent (specifically, **ADAM**) to minimize the parameters.

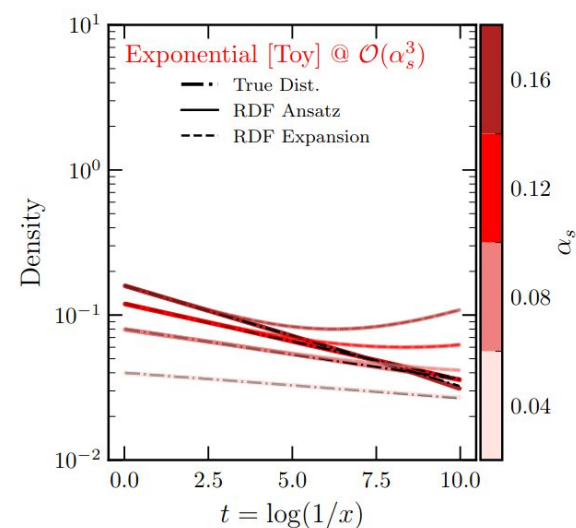
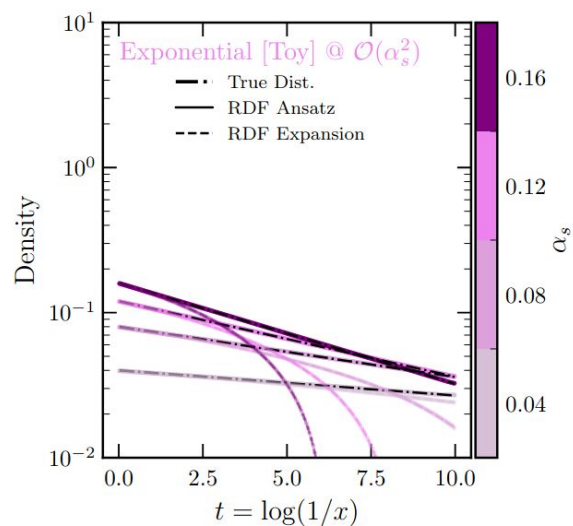
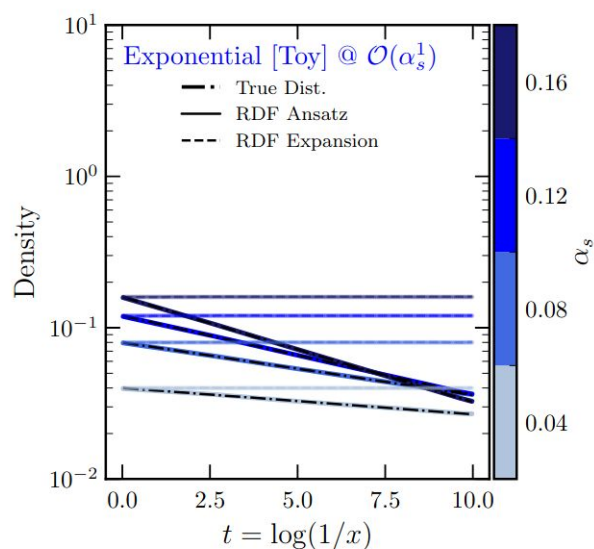
2nd Order: Refinement We use an approximate 2nd order minimizer (**LBFGS**) to refine our estimate.

“Machine Learning”?

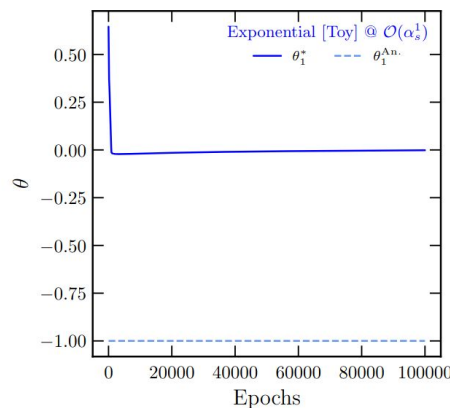
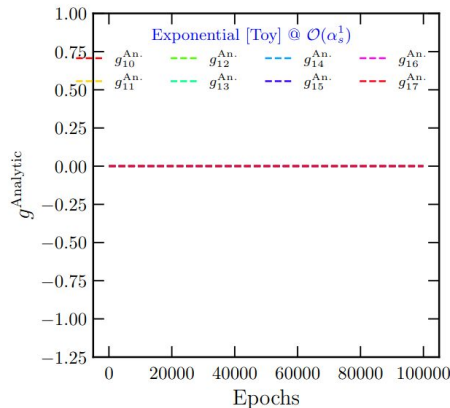
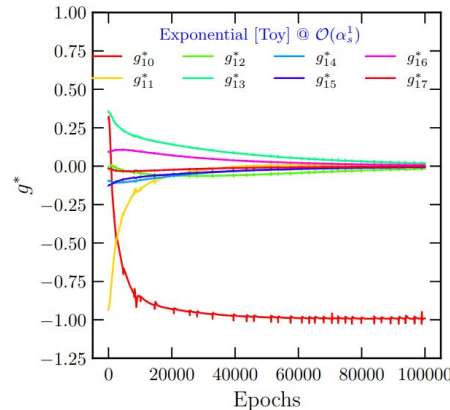
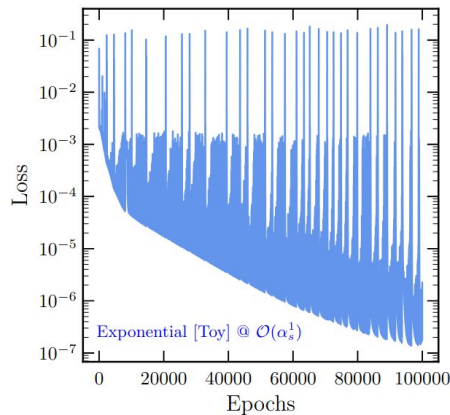
For each α in our scan, we seed the fit with the results of the previous α . We scan back and forth more than once to avoid hysteresis.

Numeric Matching for Toys

Dashed RDF Taylor Expansions = Fixed Order Exponential Expansions, so victory!



Numeric Matching for Toys



For the exponential, we know what the analytic matching coefficients should be, so we can verify the numerics:

$$g^*(t, \alpha) = \alpha$$

$$g^{\text{Analytic}}(t, \alpha) = 0 + \mathcal{O}(\alpha^M)$$

Short Detour: Simultaneous Observables

The RDF can be used to parameterize *multiple* observables at once!

$$q(t_1, t_2, \dots, t_k | \alpha) = f_1(t_1, \alpha) \times \dots \times f_k(t_1, t_2, \dots, t_k, \alpha) \\ \times \exp \left[- \int_0^{t_1} dt'_1 f_1(t'_1, \alpha) \right] \times \dots \times \exp \left[- \int_0^{t_k} dt'_k f_k(t_1, t_2, \dots, t'_k, \alpha) \right]$$

This is just a product of:

1. The RDF of observable t_1 given α
2. The RDF of observable t_2 given α and t_1
3. ...
4. The RDF of observable t_k given α and $t_{k-1}, t_{k-2}, \dots, t_1$

Short Detour: Simultaneous Observables

The RDF can be used to parameterize *multiple* observables at once!

$$q(t_1, t_2, \dots, t_k | \alpha) = f_1(t_1, \alpha) \times \dots \times f_k(t_1, t_2, \dots, t_k, \alpha) \\ \times \exp \left[- \int_0^{t_1} dt'_1 f_1(t'_1, \alpha) \right] \times \dots \times \exp \left[- \int_0^{t_k} dt'_k f_k(t_1, t_2, \dots, t'_k, \alpha) \right]$$

This is just a product of:

1. The RDF of observable t_1 given α
2. The RDF of observable t_2 given α and t_1
3. ...
4. The RDF of observable t_k given α and $t_{k-1}, t_{k-2}, \dots, t_1$

Follows straightforwardly from decomposing multivariate distributions

“**autoregressively**”: $p(t_1, t_2, \dots, t_k | \alpha) = p(t_1 | \alpha) \cdot p(t_2 | t_1, \alpha) \cdot \dots \cdot p(t_k | t_1, t_2, \dots, t_{k-1}, \alpha)$

Short Detour: Simultaneous Observables

The RDF can be used to parameterize *multiple* observables at once!

$$q(t_1, t_2, \dots, t_k | \alpha) = f_1(t_1, \alpha) \times \dots \times f_k(t_1, t_2, \dots, t_k, \alpha) \\ \times \exp \left[- \int_0^{t_1} dt'_1 f_1(t'_1, \alpha) \right] \times \dots \times \exp \left[- \int_0^{t_k} dt'_k f_k(t_1, t_2, \dots, t'_k, \alpha) \right]$$

This is just a product of:

1. The RDF of observable t_1 given α
2. The RDF of observable t_2 given α and t_1
3. ...
4. The RDF of observable t_k given α and $t_{k-1}, t_{k-2}, \dots, t_1$

This *does* depend
on the observable
ordering ... but only
at higher orders!



At the matching order, the labeling
doesn't matter. Good!

Follows straightforwardly from decomposing multivariate distributions

“**autoregressively**”: $p(t_1, t_2, \dots, t_k | \alpha) = p(t_1 | \alpha) \cdot p(t_2 | t_1, \alpha) \cdot \dots \cdot p(t_k | t_1, t_2, \dots, t_{k-1}, \alpha)$

Simultaneous Observables: Joint Angularity

Let's try angularities with two different angular moment weights!

$$\begin{aligned}
 p(t_\alpha, t_\beta) &= \frac{\alpha_s C_F}{\pi(\alpha - \beta)} \\
 p(t_\beta) &= \frac{\alpha_s C_F}{\pi\beta} t_\beta \Theta(1 - e^{-t_\beta}) \\
 p(t_\alpha|t_\beta) &= \frac{\beta}{\alpha - \beta} \frac{1}{t_\beta} \Theta(t_\beta < t_\alpha < \frac{\alpha}{\beta} t_\beta)
 \end{aligned}$$

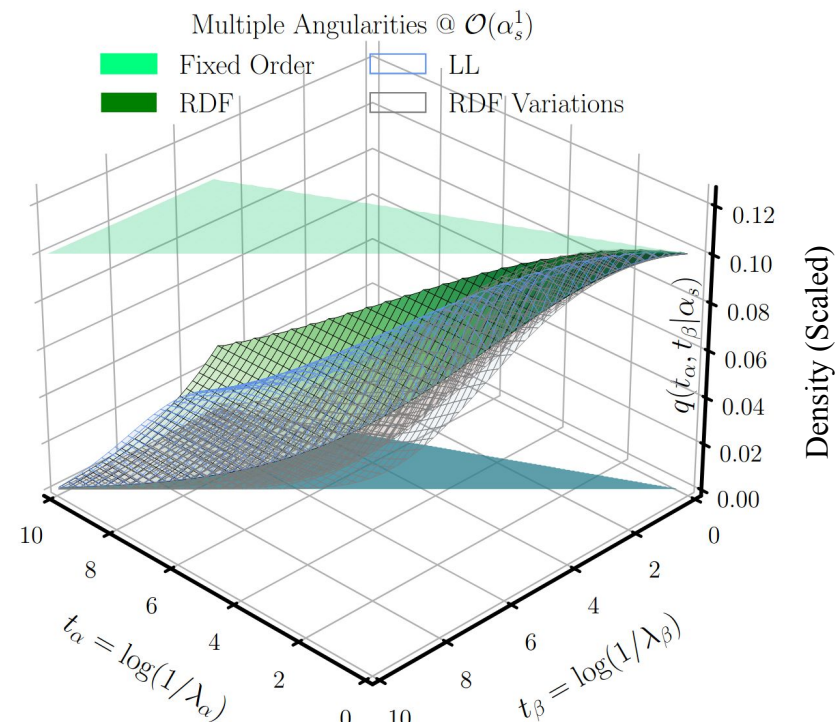
For simplicity,
soft-collinear limit
of angularities to
leading order



Turning the RDF crank...

$$q(t_\alpha, t_\beta|\alpha_s) = \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} \left(1 + \underbrace{\mathcal{O}(\alpha_s)}_{\text{Higher Order unconstrained terms}} e^{-\alpha_s \frac{C_F}{\pi\beta} t_\beta^2} - \mathcal{O}(\alpha_s) \Theta_{\alpha\beta} \right)$$

Joint resummations are generically hard.
But with the RDF, we can at least begin to
parameterize what they might look like!



*Need to be extremely careful with Theta-Functions!

Simultaneous Observables: Joint Angularity

Let's try angularities with two different angular moment weights!

$$\begin{aligned}
 p(t_\alpha, t_\beta) &= \frac{\alpha_s C_F}{\pi(\alpha - \beta)} \\
 p(t_\beta) &= \frac{\alpha_s C_F}{\pi\beta} t_\beta \Theta(1 - e^{-t_\beta}) \\
 p(t_\alpha|t_\beta) &= \frac{\beta}{\alpha - \beta} \frac{1}{t_\beta} \Theta(t_\beta < t_\alpha < \frac{\alpha}{\beta} t_\beta)
 \end{aligned}$$

For simplicity,
soft-collinear limit
of angularities to
leading order



Turning the RDF crank...

$$q(t_\alpha, t_\beta|\alpha_s) = \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} (1 + \mathcal{O}(\alpha_s)) e^{-\alpha_s \frac{C_F}{\pi\beta} t_\beta^2 - \mathcal{O}(\alpha_s)} \Theta_{\alpha\beta}$$

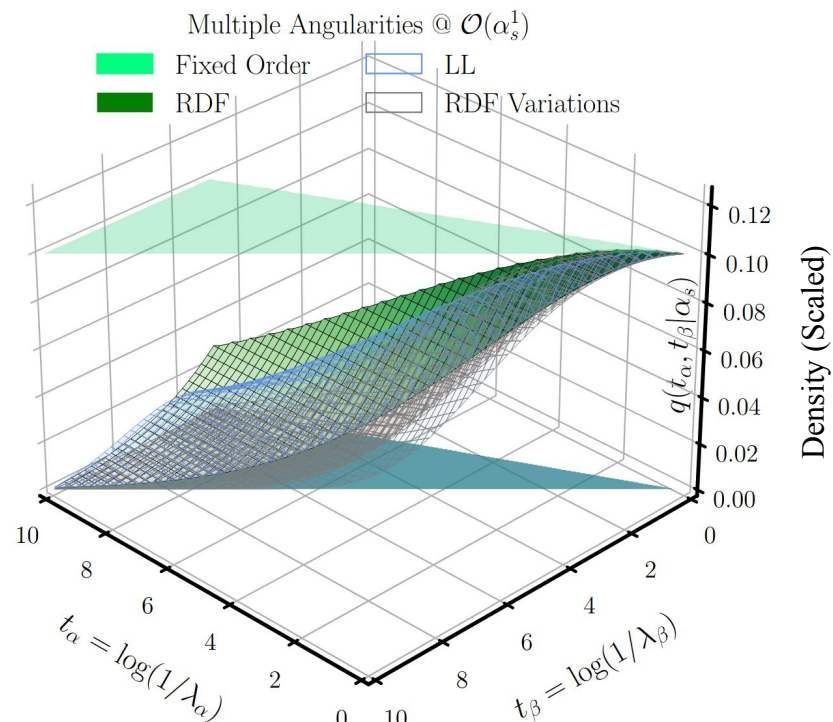
Higher Order unconstrained terms

Let's look at the marginals:

$$q(t_\beta|\alpha_s) = \frac{2\alpha_s C_F}{\pi\beta} t_\beta e^{-\frac{\alpha_s C_F}{\pi\beta} t_\beta^2}$$

$$q(t_\alpha|\alpha_s) = \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} \frac{\sqrt{\pi}}{2\sqrt{\frac{\alpha_s C_F}{\pi\beta}}} \left[\operatorname{erf}\left(\sqrt{\frac{\alpha_s C_F}{\pi\beta}} t_\alpha\right) - \operatorname{erf}\left(\frac{\beta}{\alpha} \sqrt{\frac{\alpha_s C_F}{\pi\beta}} t_\alpha\right) \right]$$

Not an exact match ... but its ok to $O(\alpha^1)$! We can make the match perfect by fiddling with higher-order g to cancel out these effects



Simultaneous Observables: Joint Angularity

Let's try angularities with two different angular moment weights!

$$\begin{aligned}
 p(t_\alpha, t_\beta) &= \frac{\alpha_s C_F}{\pi(\alpha - \beta)} \\
 p(t_\beta) &= \frac{\alpha_s C_F}{\pi\beta} t_\beta \Theta(1 - e^{-t_\beta}) \\
 p(t_\alpha|t_\beta) &= \frac{\beta}{\alpha - \beta} \frac{1}{t_\beta} \Theta(t_\beta < t_\alpha < \frac{\alpha}{\beta} t_\beta)
 \end{aligned}$$

For simplicity,
soft-collinear limit
of angularities to
leading order

Turning the RDF crank...

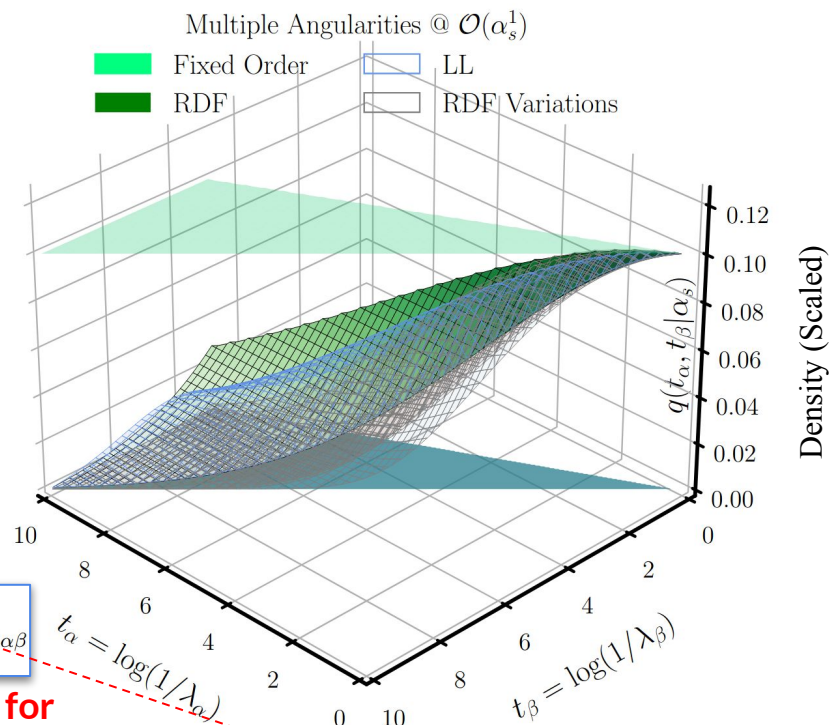
$$q(t_\alpha, t_\beta|\alpha_s) = \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} (1 + \mathcal{O}(\alpha_s)) e^{-\alpha_s \frac{C_F}{\pi\beta} t_\beta^2} \mathcal{O}(\alpha_s) \Theta_{\alpha\beta}$$

Let's compare to the known LL:

$$q^{\text{LL}}(t_\alpha, t_\beta|\alpha_s) = \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} \left(1 + \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} \frac{(t_\beta - t_\alpha)(\beta t_\alpha - \alpha t_\beta)}{\beta} \right) e^{-\alpha_s \frac{C_F}{\pi} \left(\frac{t_\beta^2}{\beta} + \frac{(t_\alpha - t_\beta)^2}{\alpha - \beta} \right)} \Theta_{\alpha\beta}$$

**This log is
unaccounted for**

**These logs
match**



The RDF is *not* guaranteed to be a full logarithmic resummation! **This log** is only first visible at $\mathcal{O}(\alpha^2)$, so the RDF matched to $\mathcal{O}(\alpha^1)$ missed it.

Simultaneous Observables: Joint Angularity

Let's try angularities with two different angular moment weights!

$$\begin{aligned} p(t_\alpha, t_\beta) &= \frac{\alpha_s C_F}{\pi(\alpha - \beta)} \\ p(t_\beta) &= \frac{\alpha_s C_F}{\pi\beta} t_\beta \Theta(1 - e^{-t_\beta}) \\ p(t_\alpha|t_\beta) &= \frac{\beta}{\alpha - \beta} \frac{1}{t_\beta} \Theta(t_\beta < t_\alpha < \frac{\alpha}{\beta} t_\beta) \end{aligned}$$

For simplicity,
soft-collinear limit
of angularities to
leading order



Turning the RDF crank...

$$q(t_\alpha, t_\beta|\alpha_s) = \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} (1 + \mathcal{O}(\alpha_s)) e^{-\alpha_s \frac{C_F}{\pi\beta} t_\beta^2} \mathcal{O}(\alpha_s) \Theta_{\alpha\beta}$$

Let's compare to the known LL:

$$q^{\text{LL}}(t_\alpha, t_\beta|\alpha_s) = \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} \left(1 + \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} \frac{(t_\beta - t_\alpha)(\beta t_\alpha - \alpha t_\beta)}{\beta} \right) e^{-\alpha_s \frac{C_F}{\pi} \left(\frac{t_\beta^2}{\beta} + \frac{(t_\alpha - t_\beta)^2}{\alpha - \beta} \right)} \Theta_{\alpha\beta}$$

**This log is
unaccounted for**

These logs
match

But the LL *does* live within the RDF's
undetermined part!
If we choose this higher-order wiggle

$$f_\alpha^{\text{LL}} = \frac{\beta}{\alpha t_\beta - \beta t_\alpha} \Theta_{\alpha\beta} + \frac{2\alpha_s C_F}{\pi(\alpha - \beta)} (t_\alpha - t_\beta) \Theta_{\alpha\beta}$$

We would reproduce the LL.

So, it lives within the “theory
uncertainty” of the RDF.

We'll make theory uncertainties more
precise in a second!

The RDF is *not* guaranteed to be a full logarithmic resummation! **This log** is only first visible at $\mathcal{O}(\alpha^2)$, so the RDF matched to $\mathcal{O}(\alpha^1)$ missed it.

Theory Uncertainties and Priors

Philosophically, all theory uncertainties attempt to model our ignorance about higher-order terms.

Scale Variations attempt to do this by varying renormalization scales, which affects higher order terms. But it assumes that higher orders look like lower orders.

TNPs make some broad (and probably reasonable) assumptions about the form of these higher order terms

Other approaches [Cacciari, Houdeau; [1105.5152](#)] use boundedness / truncation properties.

RDFs make no assumptions^{*}, but if the higher order terms can truly be anything, all statistical power gets absorbed away — *not even wrong!*

We need a regulator^{**} or a prior on coefficients to break degeneracies.

$$\frac{1}{2} \sum_k \frac{|\nu_k - \mu_k|^2}{\sigma_k^2}$$
$$\sigma_k = \frac{\sigma}{m!n!} \quad \mu_k = 0$$

Let's just put a reasonable Gaussian prior on g and not think about it too hard or fine-tune^{***} it.

Pontificating: Theory uncertainties are inherently Bayesian as higher-order terms aren't probabilistic. That's why I said "semi-frequentist" earlier.

^{*}Other than unitarity

^{**}The fact that the RDF only has finitely many parameters is a kind of regularization, but it is extremely weak.

^{***}You can always fine-tune a prior to give almost any answer you want. We consider it a feature of the RDF that simple unit Gaussians work well.

NⁿLL Resummed Distribution Functions (?)

The RDF is *an* all-orders resummation that matches the F.O. calculation. It is not *the* logarithmic resummation – but they relate!

$$P(x|\alpha) = \left(1 + \sum_{m=1}^{\infty} C_m \alpha^m\right) \times \exp \left[- \sum_{m=1}^{\infty} \sum_{n=1}^{m+1} F_{mn} \alpha^m \log^n(1/x) \right] + \cancel{R(x)}_{\text{Non-Singular}}$$

Leading Log: All terms $F_{n,n+1}$ included

Next-to-Leading Log: All terms $F_{n,n}$ included

Maps directly onto the RDF *only if* (a) f is polynomial and (b) all relevant logs are included in the F.O. calculation to match (including running coupling, NGLs, etc)

$$Q(x|\alpha) = \exp[-F(\log(1/x), \alpha)], \quad \text{where } F(t, \alpha) = \int_0^t dt' f(t', \alpha)$$

Higher order unconstrained g = “guesses” of the $N^{n>N}$ LL logarithms, can set to zero

[Details] Fitting

Try our repo!



We use a χ^2 binned likelihood:

$$-\log \mathcal{L}(\alpha, \nu) = \frac{1}{2} \sum_{\text{Bin}_i} \frac{|\text{RDF}(\text{Bin}_i, \alpha, \phi, \nu) - \text{data}(\text{Bin}_i)|^2}{\text{error}(\text{Bin}_i)^2} + \frac{1}{2} \sum_k \frac{|\nu_k - \mu_k|^2}{\sigma_k^2}$$

For each α in a fine-grained scan, we fit for the optimal **order-(M+1) g nuisance parameters** that minimize the likelihood, keeping the matched **order-(M) and below g parameters** frozen.

The best fit α is then the one that minimizes the profiled log likelihood.

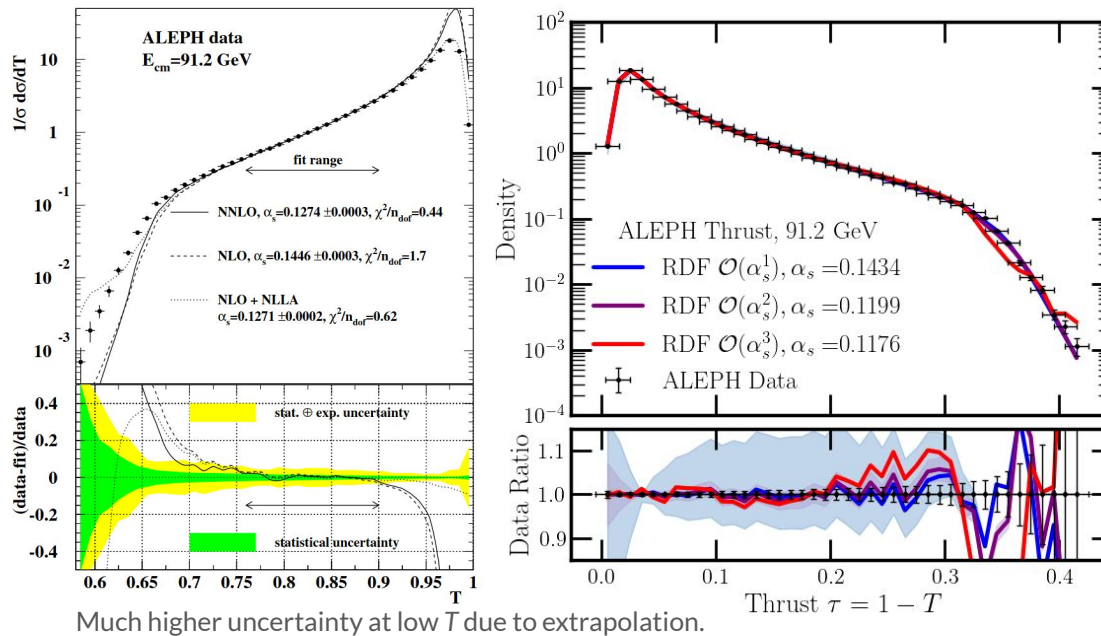
We can construct confidence intervals in the region:

$$-2 \log \mathcal{L}(\alpha) < 1$$

We don't check that our intervals are calibrated / check for Wilke's Theorem.
But in a real analysis, you should!

For Comparison – Just Fixed Order

[Dissertori, Gehrmann–De Ridder, Gehrmann, Glover, Heinrich, Stenzel; [0712.0327](#)]

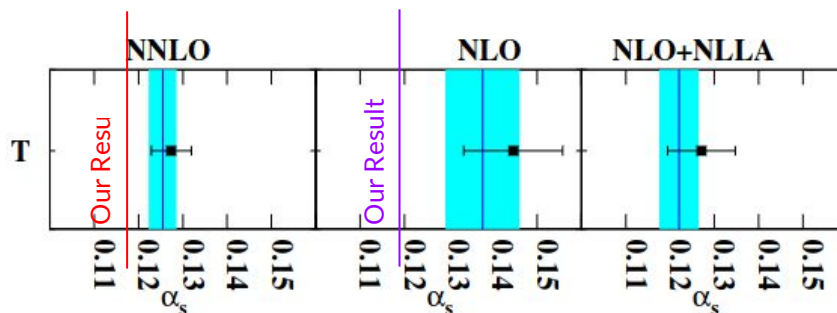


To fit just a pure fixed order calculation, use only a window.

Pure fixed order results are very different! α goes to 0.125 or even higher.

The **RDF**, *essentially for free*, obtains results consistent with the PDG average, *still only using fixed order info!*

All we did was re-organize the fixed order info in a motivated way.



[[0712.0327](#). Figure slightly modified to remove other observables]

Resummed Distribution Functions

The RDF works by re-organizing the perturbative series from expanding in α^m to expanding in

$$\alpha^{m-m^*} \frac{p_m(t)}{p_{m^*}(t)} \ll 1$$

← Analogous, and sometimes equal to, to expanding in $\alpha^m \log^n$

Recall: α^{m^*} and p_{m^*} given by the leading order term

This is *not necessarily* the usual logarithmic NⁿLL resummation in QCD, just a **generic all-orders re-summation** consistent with unitarity.

The p_m may contain large logs, and then these will be resummed in g , but *only those particular logarithms implied by the fixed-order input* and no others.

No RG structure or factorization used whatsoever. Completely physics agnostic!

Future Directions and Ideas [Cont'd]

We took the **physics we know** to be a fixed order calculation, and the **physics we don't know** to be higher orders.

$$g_{mn} = \begin{pmatrix} \begin{matrix} g_{00} & g_{01} & g_{02} & g_{03} & g_{04} & g_{05} & g_{06} & g_{07} & g_{08} & g_{09} \\ g_{10} & g_{11} & g_{12} & g_{13} & g_{14} & g_{15} & g_{16} & g_{17} & g_{18} & g_{19} \\ g_{20} & g_{21} & g_{22} & g_{23} & g_{24} & g_{25} & g_{26} & g_{27} & g_{28} & g_{29} \\ g_{30} & g_{31} & g_{32} & g_{33} & g_{34} & g_{35} & g_{36} & g_{37} & g_{38} & g_{39} \\ g_{40} & g_{41} & g_{42} & g_{43} & g_{44} & g_{45} & g_{46} & g_{47} & g_{48} & g_{49} \end{matrix} \end{pmatrix}$$

Powers of logs t →

Known

Unknown

Orders of α ↓

Surely there are other ways to split what we know and what we don't! Perhaps we know some logs (fitting Pythia or Panscales) and don't know others?

$$g_{mn} = \begin{pmatrix} \begin{matrix} g_{00} & g_{01} & g_{02} & g_{03} & g_{04} & g_{05} & g_{06} & g_{07} & g_{08} & g_{09} \\ g_{10} & g_{11} & g_{12} & g_{13} & g_{14} & g_{15} & g_{16} & g_{17} & g_{18} & g_{19} \\ g_{20} & g_{21} & g_{22} & g_{23} & g_{24} & g_{25} & g_{26} & g_{27} & g_{28} & g_{29} \\ g_{30} & g_{31} & g_{32} & g_{33} & g_{34} & g_{35} & g_{36} & g_{37} & g_{38} & g_{39} \\ g_{40} & g_{41} & g_{42} & g_{43} & g_{44} & g_{45} & g_{46} & g_{47} & g_{48} & g_{49} \end{matrix} \end{pmatrix}$$

Powers of logs t →

Known

Unknown

Orders of α ↓

Future Directions and Ideas

- (1) We chose the simplest possible polynomial parameterization and the simplest priors

$$g^*(t, \alpha), \quad g_{\text{Analytic}}(t, \alpha) = \sum_{m=0, n=0}^{M, N} g_{mn} \alpha^m t^n \Theta(t - \theta_{mn})$$

$$\frac{1}{2} \sum_k \frac{|\nu_k - \mu_k|^2}{\sigma_k^2}$$

Surely its possible to do something smarter with a little thought!

- (2) We explicitly ignored resurgence / asymptotic series / nonperturbative effects.

$$g_{\text{RDF}} \stackrel{?}{=} \sum_m \alpha^m g_m + \cancel{\frac{1}{\alpha} g_{\text{asympt.}}} + \cancel{g_{\text{non-pert.}}}$$

Surely there is a clever way to parameterize these nicely!

- (3) We didn't talk about enforcing RG equations or SCET factorization with or within the RDF. Surely you can inject some QCD knowledge this way!