SPECTER: Efficient Evaluation of the Spectral EMD

Rikab Gambhir

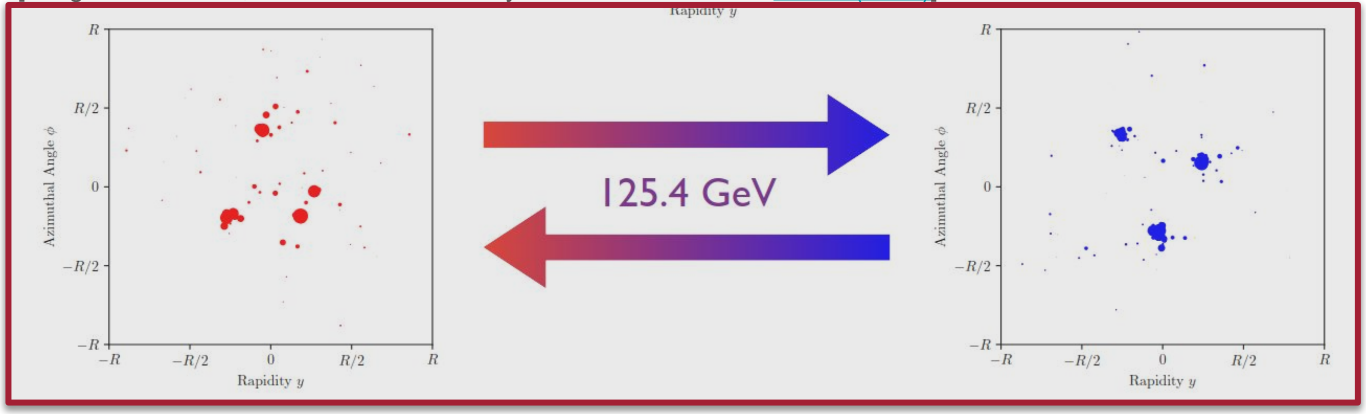
Email me questions at rikab@mit.edu!

Based on [**RG**, Larkoski, Thaler, 24XX.XXXX]

Review: Optimal Transport

The **Wasserstein Metric**, a.ka. **Earth/Energy Mover's Distance (EMD)***
allows us to quantitatively ask "How far are two distributions?"

[Image from Thaler; The Hidden Geometry of Particle Collisions Slides (2019)]



$$\text{EMD}_{\beta,R}(\mathcal{E},\mathcal{E}') = \min_{\{f_{ij}\geq 0\}} \sum_{i=1}^{M}\sum_{j=1}^{M'} f_{ij}\left(\frac{\theta_{ij}}{R}\right)^{\beta} + \left|\sum_{i=1}^{M} E_i - \sum_{j=1}^{M'} E'_j\right|,$$

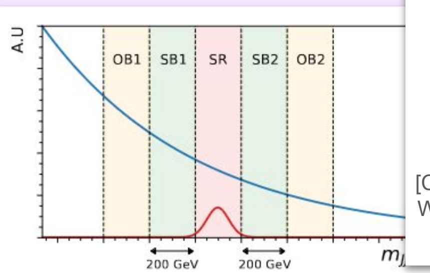
$$\sum_{i=1}^{M} f_{ij} \leq E'_j, \qquad \sum_{j=1}^{M'} f_{ij} \leq E_i, \qquad \sum_{i=1}^{M}\sum_{j=1}^{M'} f_{ij} = \min\left(\sum_{i=1}^{M} E_i, \sum_{j=1}^{M'} E'_j\right)$$

Very useful for LHC collision data and jets, which are distributions of energy!

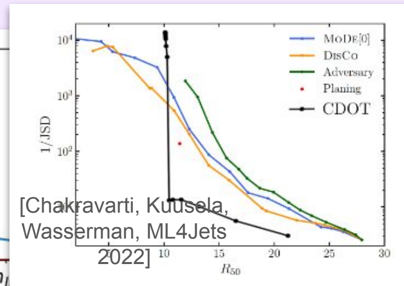
*For this talk, "Optimal Transport", "Wasserstein Metric", "Energy Mover's Distance", and "Earth Mover's Distance" are all synonyms.

The **Wasserstein Metric,** a.ka. **Earth/Energy Mover's Distance (EMD)** has seen increasing interest in jet physics:

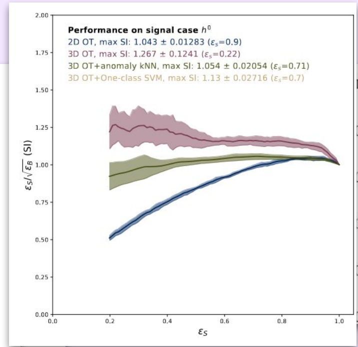
Searches and Anomaly Detection



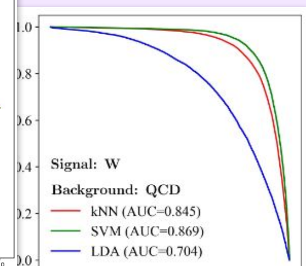
[Raine, Klein, Sengupta, Golling, 2203.09470]



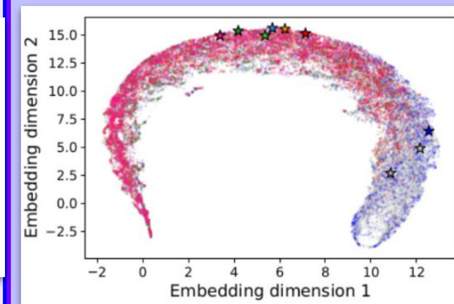
[Chakravarti, Kuusela, Wasserman, ML4Jets 2022]



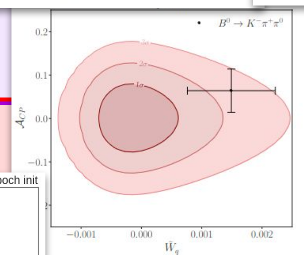
[Craig, Howard, Li, 2401.15542]



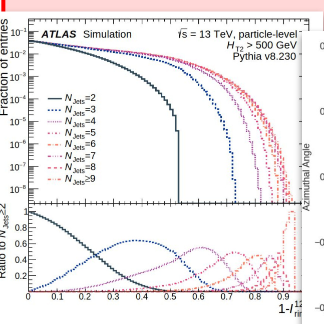
[Cai, Cheng, Craig, Craig, 2008.08604]



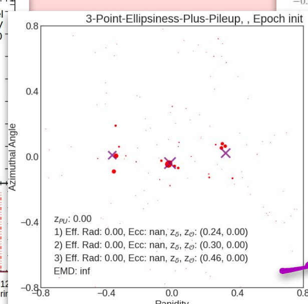
[Romao, Castro, Milhano, Pedro, Vale, 2004.09360]



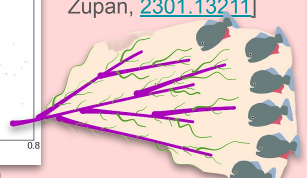
[Davis, Menzo, Youssef, Zupan, 2301.13211]



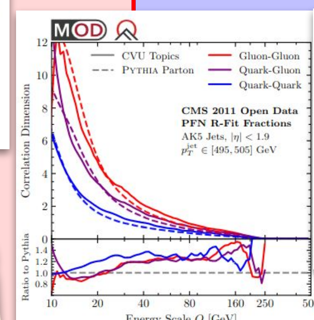
[ATLAS, 2305.16930]



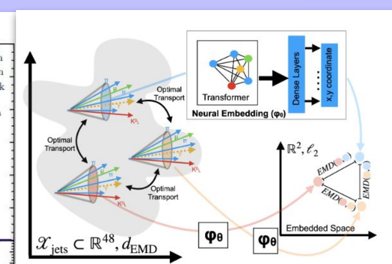
[Ba, Dogra, **RG**, Tasissa, Thaler, 2302.12266]



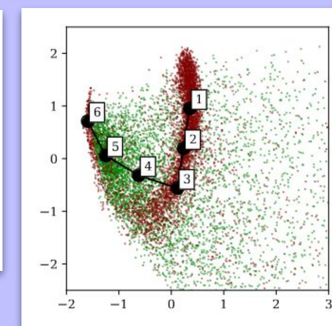
[Alipour-Fard, Komiske, Metodiev, Thaler, 2305.00989]



[Komiske, Kryhin, Thaler, 2205.04459]



[Park, Harris, Ostdiek, 2208.05484]



[Cai, Cheng, Schmitzer, Thorpe, DOI:10.1137/21M1400080]

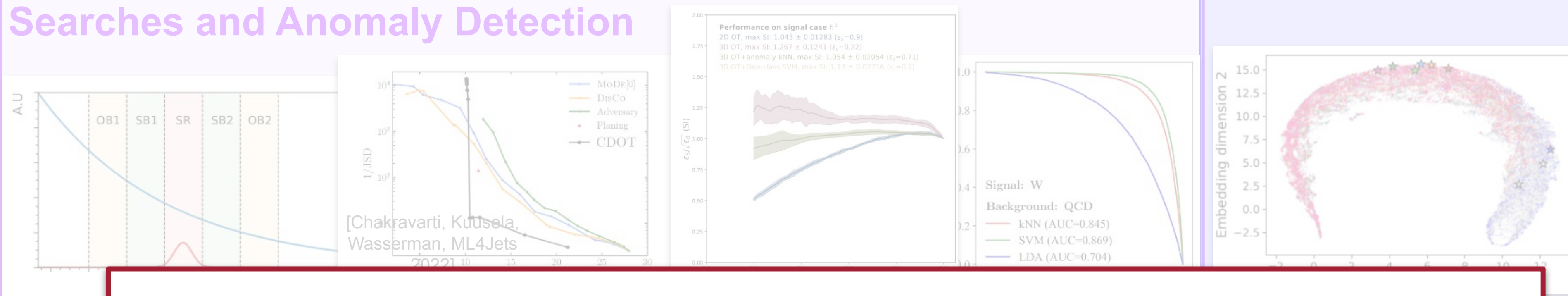
Safe Observables for Colliders + Jets

Correlations and Embeddings

Not an exhaustive list, let me know if I haven't included your recent EMD application!

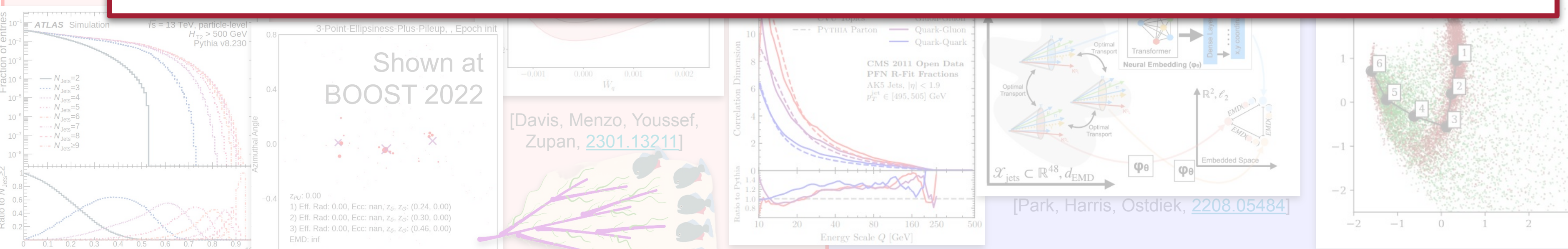
The **Wasserstein Metric**, a.ka. **Earth/Energy Mover's Distance (EMD)** has seen increasing interest in jet physics:

Searches and Anomaly Detection



both computationally, and also in QCD calculations…

But! The EMD is **hard/expensive to calculate**, and even **harder to minimize**…



[ATLAS, 2305.16930]

[Ba, Dogra, **RG**, Tasissa, Thaler, 2302.12266]

[Alipour-Fard, Komiske, Metodiev, Thaler, 2305.00989]

[Davis, Menzo, Youssef, Zupan, 2301.13211]

[Komiske, Kryhin, Thaler, 2205.04459]

[Park, Harris, Ostdiek, 2208.05484]

[Cai, Cheng, Schmitzer, Thorpe, DOI:10.1137/21M1400080]

Safe Observables for Colliders + Jets

Correlations and Embeddings

Not an exhaustive list, let me know if I haven't included your recent EMD application!

Today …

An EMD-like metric with associated EMD-like observables that is *easier* and *faster* to calculate using the **Spectral EMD** (SEMD) and ***SPECTER***.

With the **Spectral EMD**, we can now (1) evaluate distances between events in closed form, (2) develop EMD-based observables that are fast to numerically evaluate, and (3) often write closed-form expressions for these observables

SPECTER

Logo made with DALL-E. Preliminary.

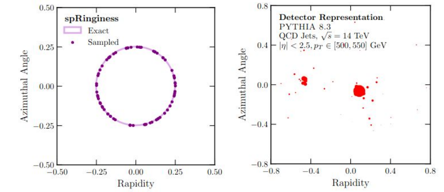
$$\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2$$

$$-\,2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right)$$

$$\times\,\Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),$$

Today …

Old Method: ~ 3 hours
New (Numeric): ~ 5 sec
New (Closed Form): ~ **Practically Instant!**

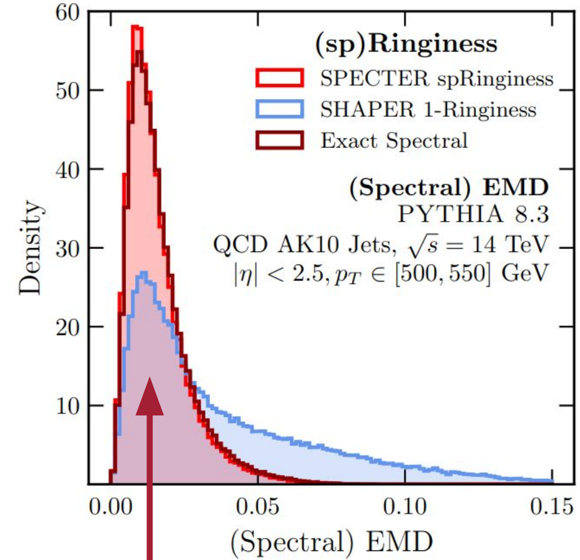


An EMD-like metric with associated EMD-like observables that is *easier* and *faster* to calculate using the **Spectral EMD** (SEMD) and ***SPECTER***.

With the **Spectral EMD**, we can now (1) evaluate distances between events in closed form, (2) develop EMD-based observables that are fast to numerically evaluate, and (3) often write closed-form expressions for these observables

EXAMPLE

How "ring-like" are Jets?



With these tools, we can evaluate the red curve in *seconds*, equivalent to ~10^7 OT problems*!

Or, for rings, we can evaluate using an exact closed form expression instantaneously!

SPECTER

Logo made with DALL-E. Preliminary.

$$\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2$$
$$- 2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right)$$
$$\times\, \Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),$$

*10^7 = 100k events × ~150 epochs

[1][Larkoski, Thaler, 2305.03751]

The Spectral EMD

Similar to the ordinary EMD, the **Spectral EMD (SEMD)**[1] is an IRC-safe metric between events or jets, computed on their **spectral representations**:

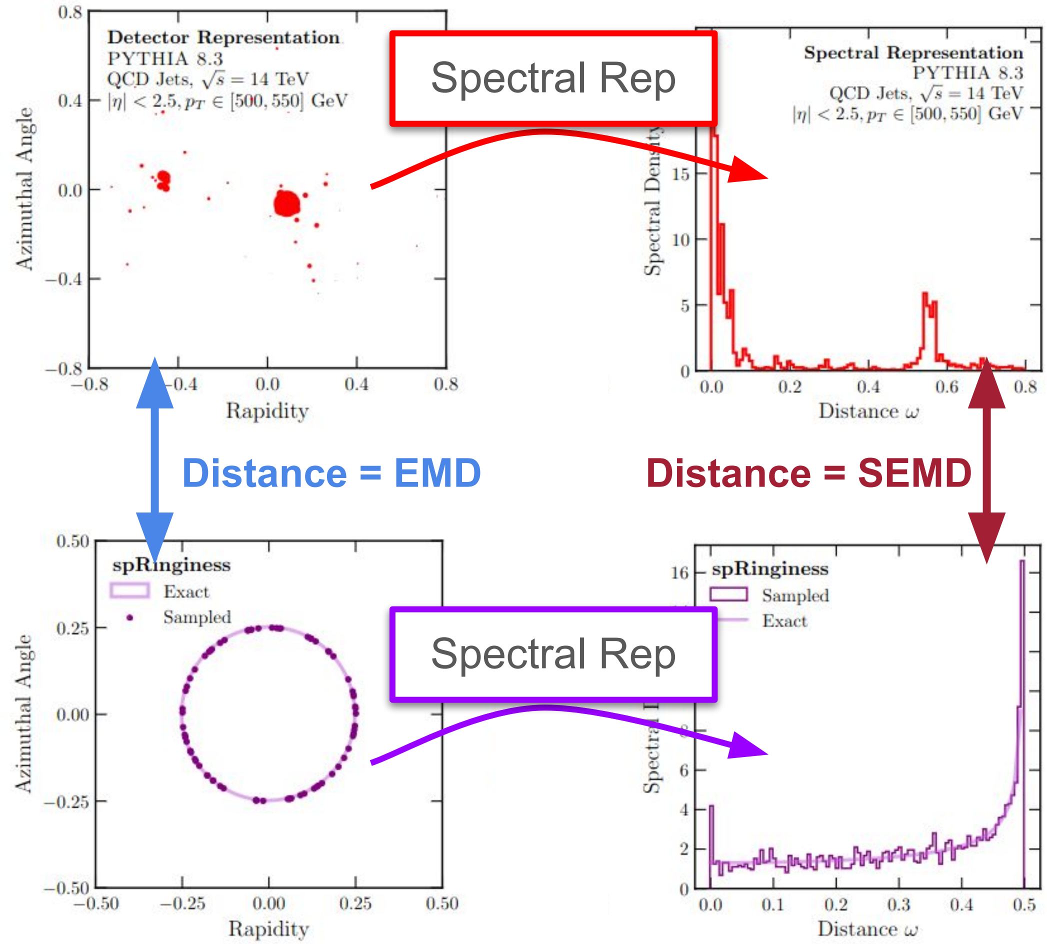
$$s(\omega) \equiv \sum_{i,j\in\mathcal{E}} E_i E_j \, \delta(\omega - \omega_{ij})$$

= list of energy-weighted pairwise distances

Then the *p*-SEMD is given by:

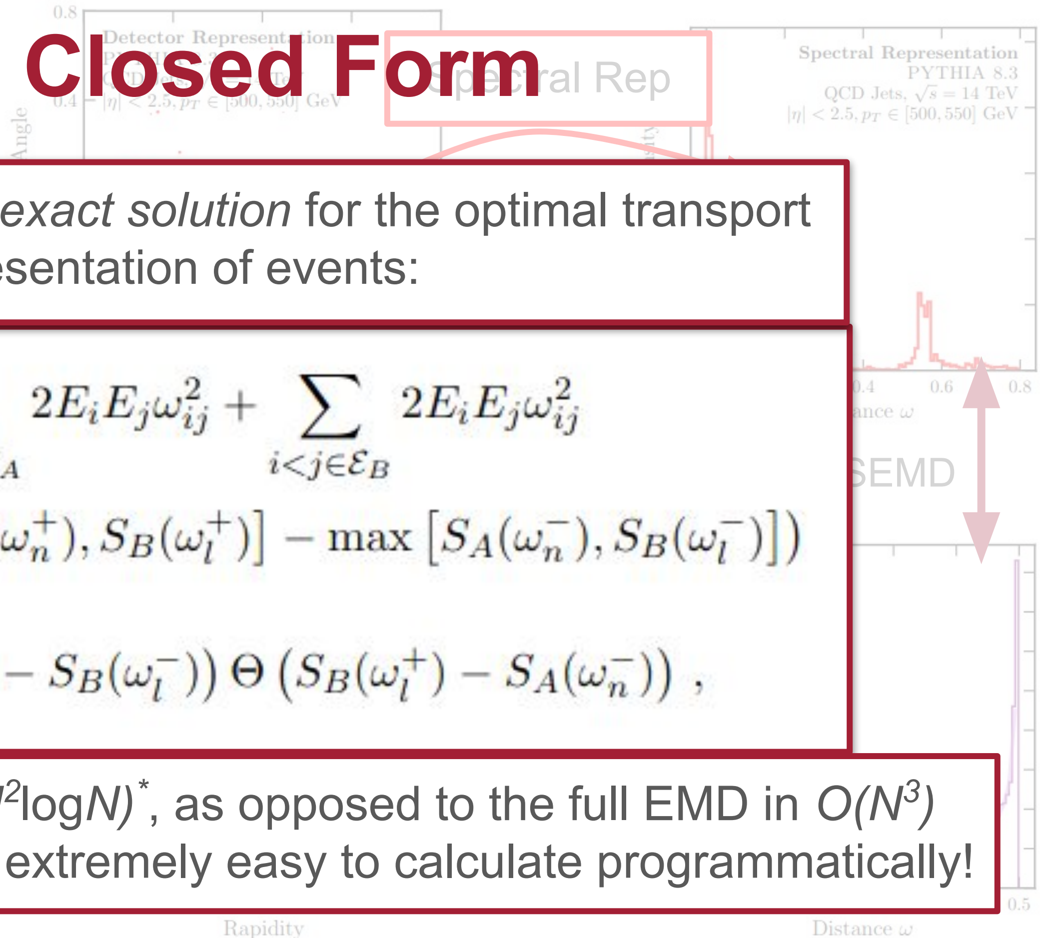
$$\text{SEMD}_p(s_A, s_B) \equiv \int_0^{E_{\text{tot}}^2} dE^2 \, \left| S_A^{-1}(E^2) - S_B^{-1}(E^2) \right|^p$$

Inverse of integral of *s(ω)*



The SEMD automatically respects all isometries of the metric ω. In this case, the SEMD is invariant under rotations or translations of either event or jet

The Spectral EMD: Closed Form



For p = 2, possible to find an *exact solution* for the optimal transport problem on the spectral representation of events:

$$\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2$$
$$-2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l\left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right)$$
$$\times\,\Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),$$

Can be computed *exactly* in *O(N²logN)**, as opposed to the full EMD in *O(N³)*
Closed form, easy derivatives and extremely easy to calculate programmatically!

$$\mathrm{SEMD}_p(s_A, s_B) \equiv \int_0^{E_{\mathrm{tot}}^2} dE^2 \left|S_A^{-1}(E^2) - S_B^{-1}(E^2)\right|^p$$

Inverse of integral of $s(\omega)$

*A 1D OT is usually *O(K log K), where K* is the number of points. Here, *K* ~ *N²* for particle pairs

The Spectral EMD: Closed Form

For p = 2, possible to find an *exact solution* for the optimal transport problem on the spectral representation of events:

$$
\begin{aligned}
\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = &\sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2 \\
&- 2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right) \\
&\times \Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),
\end{aligned}
$$

Can be computed *exactly* in $O(N^2 \log N)$*, as opposed to the full EMD in $O(N^3)$

Closed form, easy derivatives and extremely easy to calculate programmatically!

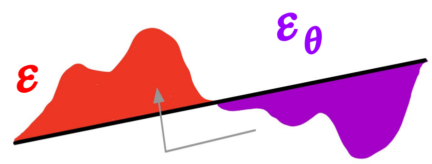
Our framework for doing this, built in Python with *JAX*

SPECTER

See also: Sinkhorn, Sliced Wasserstein, WGANs, Linearized EMDs, …

*A 1D OT is usually $O(K \log K)$, *where* K is the number of points. Here, $K \sim N^2$ for particle pairs

Technical Details …



For p = 2, possible … problem on the sp…

For events A, B, the p **spectral EMD** is defined as (1D OT!):

EMD = Work done to move "dirt" optimally

$$\mathrm{SEMD}_{\beta,p}(s_A, s_B) \equiv \int_0^{E^2_{\text{tot}}} dE^2 \left|S_A^{-1}(E^2) - S_B^{-1}(E^2)\right|^p$$

$$\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2$$
$$-2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right)$$
$$\times\, \Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),$$

S = **cumulative spectral function**
± indicates whether or not to include ω in the sum

The trick: Sum over pairs n of particles within each event.

Looks like $O(N^4)$, but with clever sorting & indexing in 1D, reduces to $\boldsymbol{O(N^2)}$!

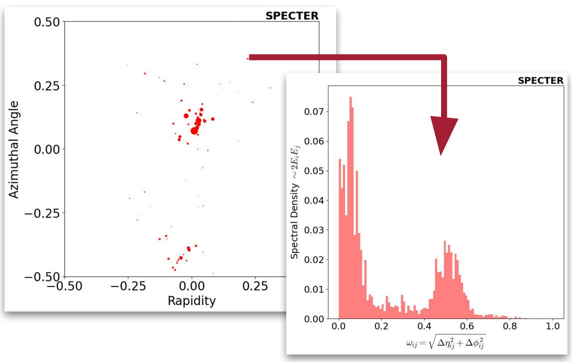
The **spectral density function**

$$s(\omega) = \sum_{i=1}^{N}\sum_{j=1}^{N} E_i\, E_j\, \delta\big(\omega - \omega(\hat{n}_i, \hat{n}_j)\big)$$

Pairwise Distances

Reduces events to 1D, while preserving all* information about the event, up to translations and rotations.

*up to measure 0, but important degeneracies, ask me about this later!



…EMDs, …

[If we have time] The Algorithm

$$\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2$$

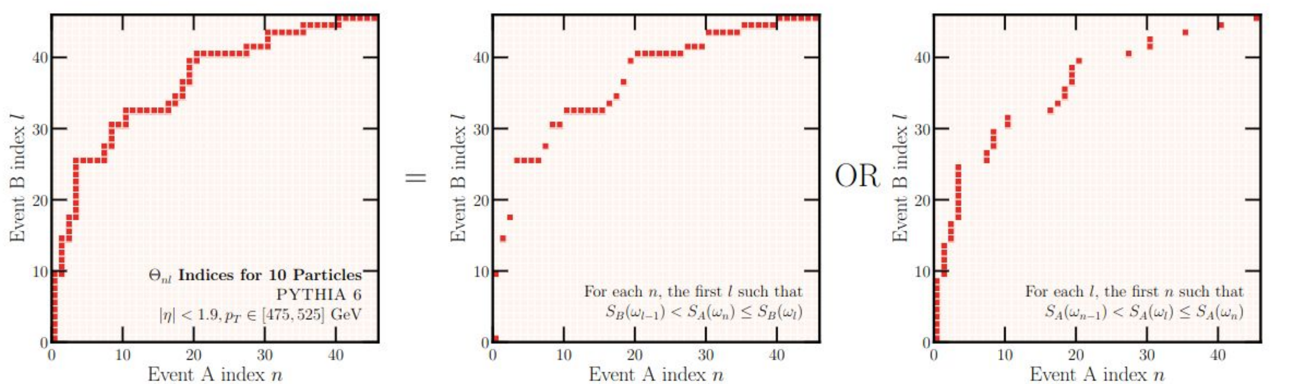
$$\boxed{-2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right) \times \Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),}$$

This sum looks like it goes as $O(N^4)$ as a sum of pairs of pairs, but it turns out only $O(N^2)$ terms survive the Θ-functions!

The Trick: Pre-compute which pairs will activate the Θ-functions by using the fact that in 1D, distances ω can be sorted!

SPECTER

$S_B(\omega_{l-1}) < S_A(\omega_n) \leq S_B(\omega_l)$ OR $S_A(\omega_{n-1}) < S_B(\omega_l) \leq S_A(\omega_n)$



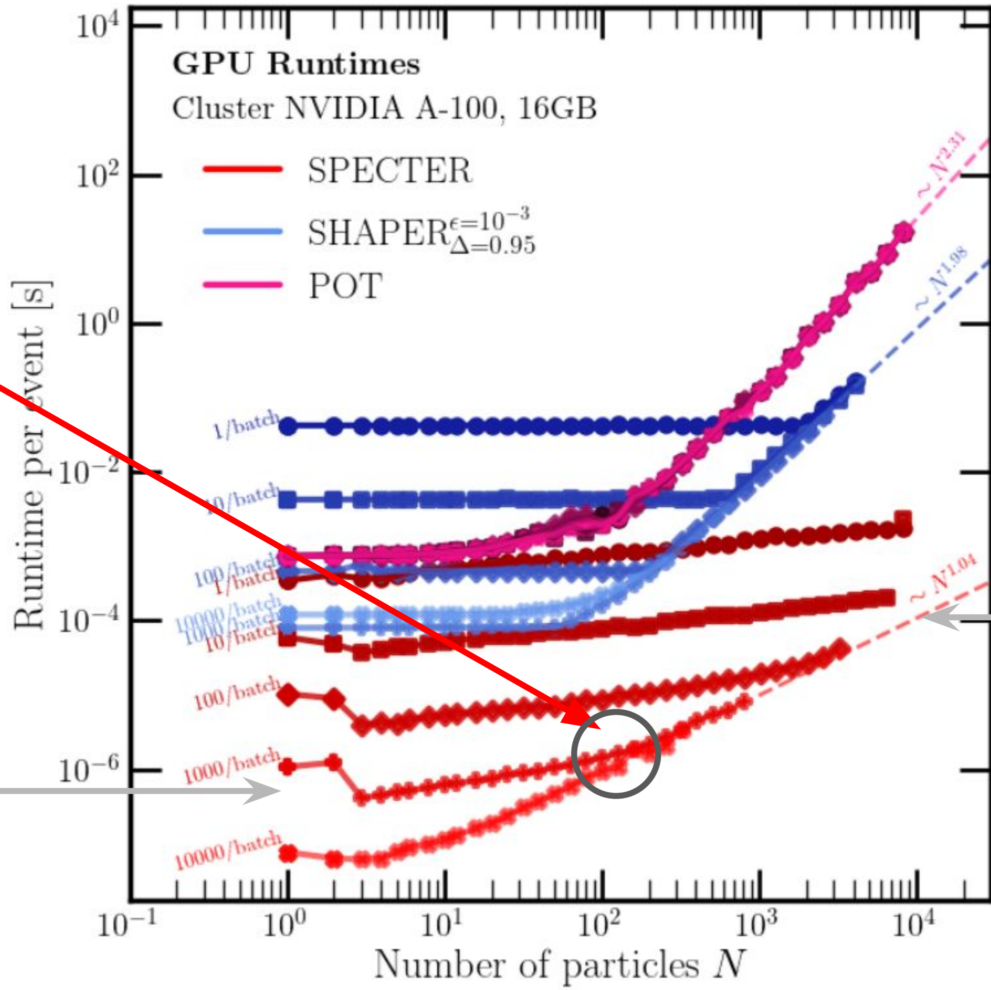
The inequalities can be evaluated efficiently on sorted lists, bringing the total runtime to ***O(N²logN)***.

Ask me afterwards if you want more details on the algorithm and how the code works!

SPECTER is *FAST* (*BOOSTED*)!

Running on a single GPU on my local compute cluster …

SEMDs between jets with ~100 particles can be computed at a rate of **one million SEMDs per second**!



Worst case SEMD scaling is $O(N^2 \log N)$ … *SPECTER* beats it in practical settings!

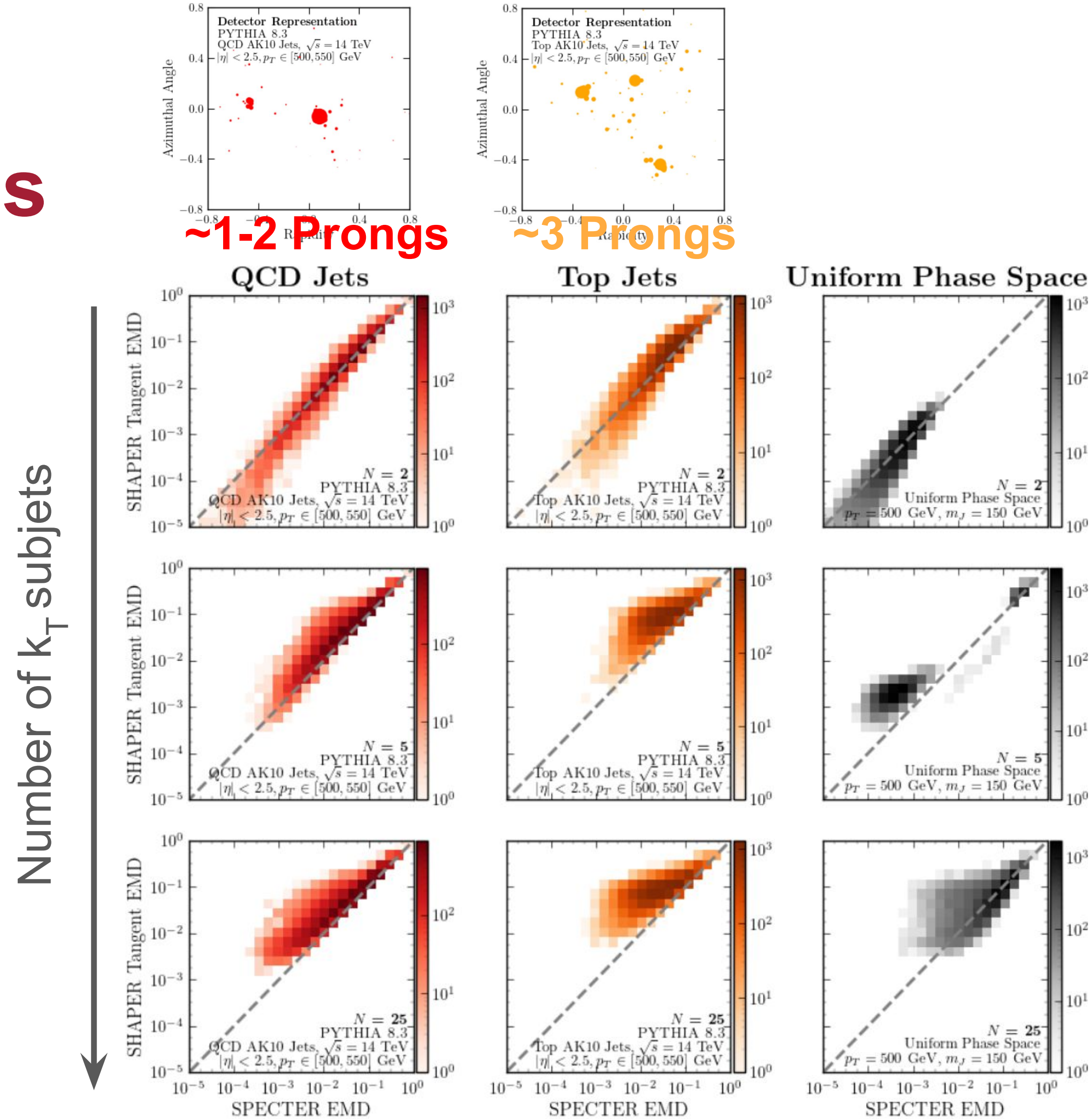
Highly parallelized: Efficiency gains by processing many events at once!

Pairwise (S)EMDs

We can now easily evaluate SEMDs between pairs of events!

The SEMD and EMD are *not* the same metric, but they are correlated, and this correlation can be different for different types of physics!

The SEMD is invariant to translations and rotations of the jets, but the EMD does not and this needs to be minimized over.

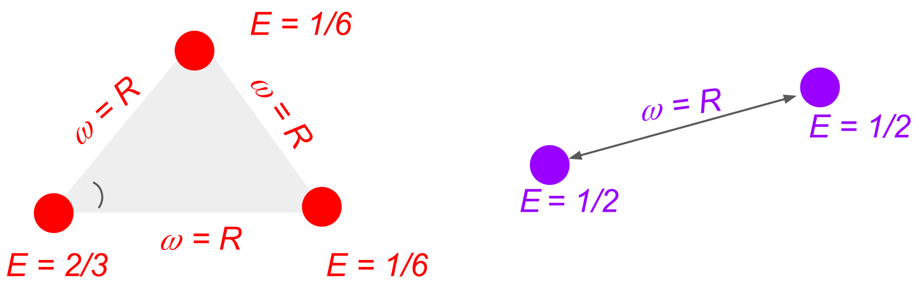


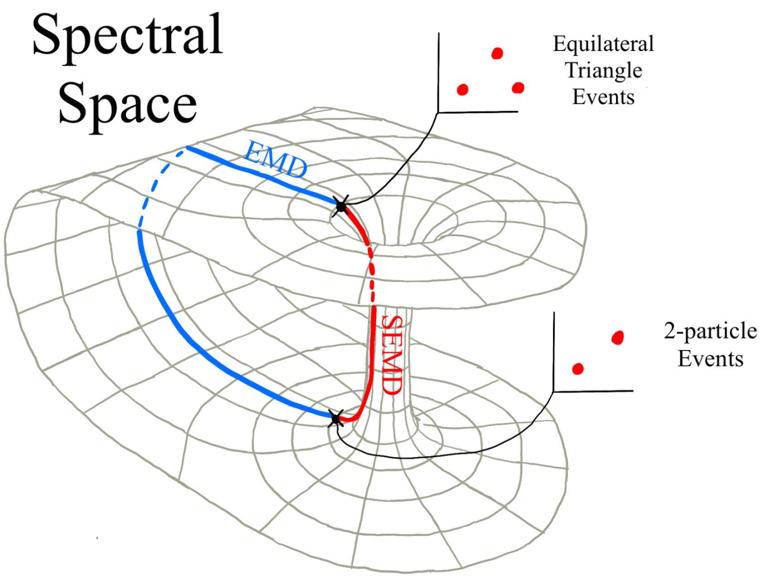
On Diagonal
SEMD ~ EMD

Off Diagonal
SEMD < EMD

The EMD vs. The SEMD

The SEMD is *topologically different* from the ordinary EMD!





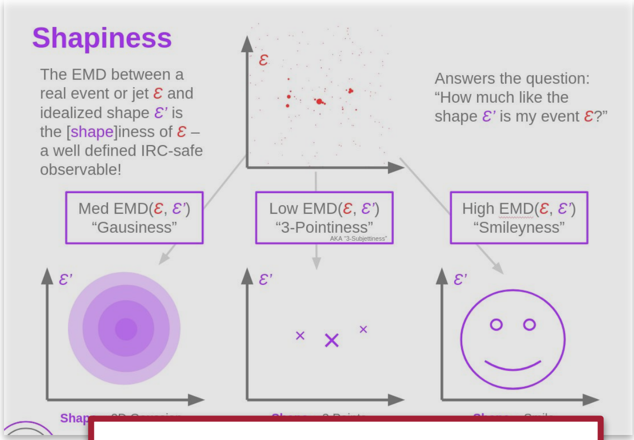
This three particle event looks very different from this two particle event, and thus they have a large EMD.

But their SEMD is zero! The spectral function only cares about pairwise distances and degenerate configurations can occur.

The space of events gets "pinched" at degenerate configurations when looking at only their spectral representations

This can come into play when events have 3 hard prongs, e.g. top jets!

SEMD Observables

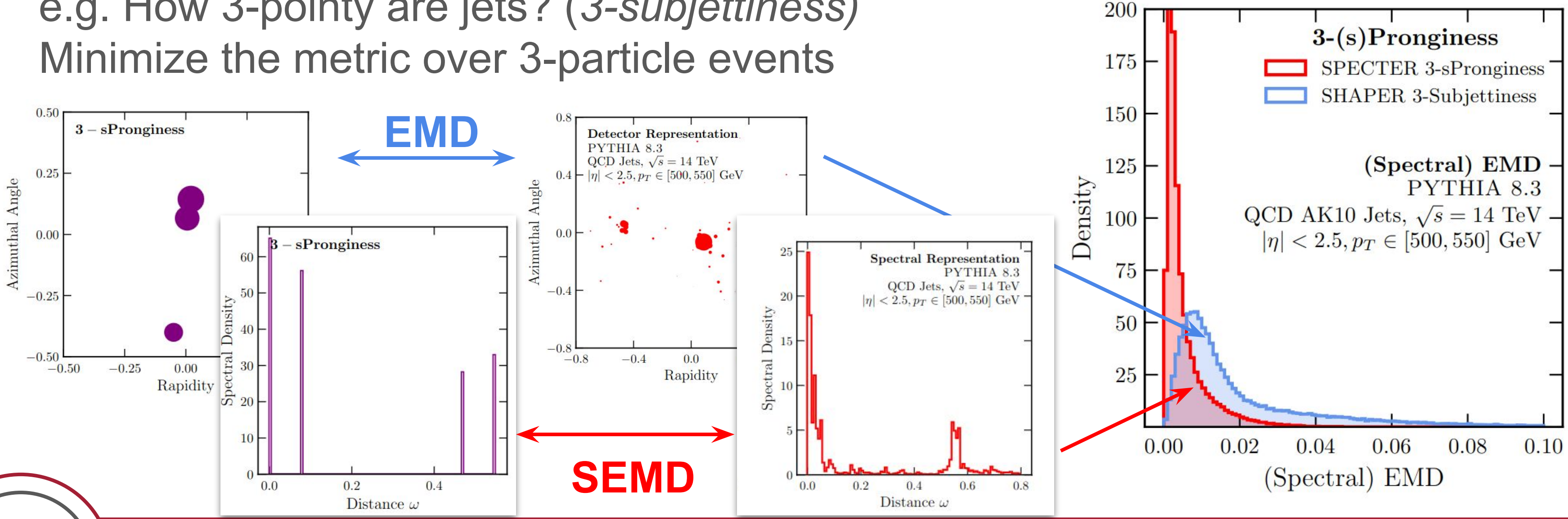


With a geometry based metric, we can now define IRC-safe **shape observables** by finding events that minimize the metric:

$$\boxed{\mathcal{O}(\mathcal{E}) = \min_{\mathcal{E}' \in \mathcal{M}} [\mathrm{SEMD}(s(\mathcal{E}), s(\mathcal{E}'))]}$$

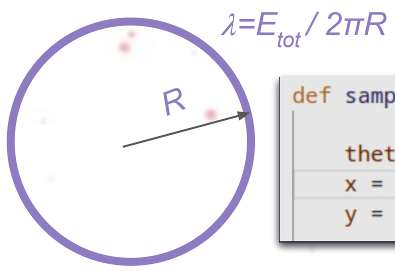
For more details, see my Past IAIFI Talks!

e.g. How 3-pointy are jets? (*3-subjettiness)*
Minimize the metric over 3-particle events



Full Example: How “ring-like” are jets?

Step 1: Define the shape with parameters




```
def sample_circle(params, N, seed):

    thetas = jax.random.uniform(seed, shape
    x = params["Radius"] * jnp.cos(thetas)
    y = params["Radius"] * jnp.sin(thetas)
```

Unlike ordinary EMD, not necessary to specify center / orientation!

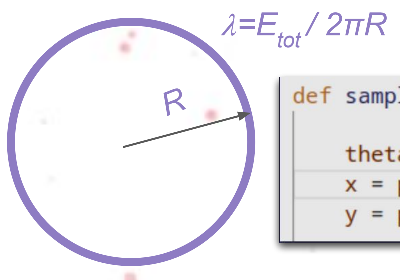
Shapes are parameterized distributions of energy on the detector space.

Many of your favorite observables, like *N*-(sub)jettiness, thrust, and angularities take the form of finding the shape that best fits an event’s energy distribution.

Custom shapes define custom IRC-safe observables – to define a shape, all you need is to define a parameterized energy distribution and how to sample points from it!

Full Example: How “ring-like” are jets?

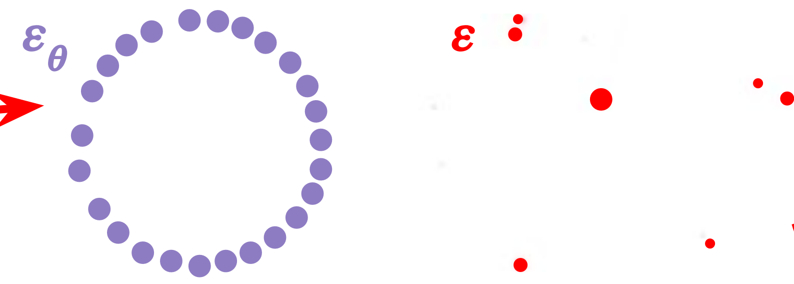
Step 1: Define the shape with parameters




```python
def sample_circle(params, N, seed):

    thetas = jax.random.uniform(seed, shape
    x = params["Radius"] * jnp.cos(thetas)
    y = params["Radius"] * jnp.sin(thetas)
```

Unlike ordinary EMD, not necessary to specify center / orientation!

Step 2: Sample from Parameterized Shapes



Step 3: Calculate the spectral metric between events and shapes

$$\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2 - 2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right) \times \Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),$$

Key difference from previous work: We use the SEMD, *not* the EMD!

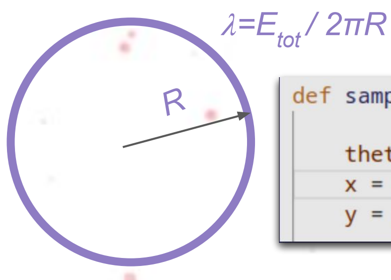
The $p = 2$ spectral EMD between two sets of discrete points has a closed-form solution with only binary discrete minimizations.

We discretize our shape by randomly sampling points from it.

If the spectral functions are sorted, can compute the SEMD in ~$O(N^2 \log N)$ time!

Full Example: How "ring-like" are jets?

Step 1: Define the shape with parameters



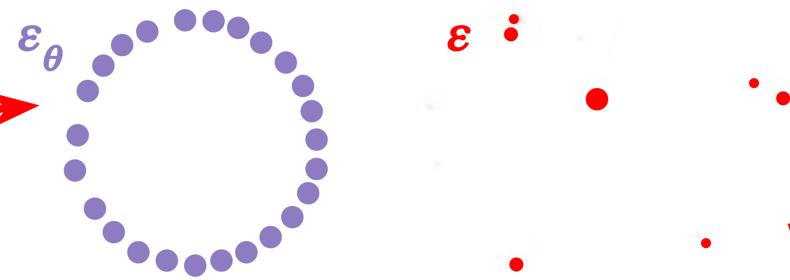

```
def sample_circle(params, N, seed):

    thetas = jax.random.uniform(seed, shape
    x = params["Radius"] * jnp.cos(thetas)
    y = params["Radius"] * jnp.sin(thetas)
```

Unlike ordinary EMD, not necessary to specify center / orientation!

We have an explicit formula for the spectral EMD, so we can automatically differentiate through it

Standard ML procedure: Sample, calculate gradients, gradient descent, repeat! Analogous to WGANS.

Step 2: Sample from Parameterized Shapes

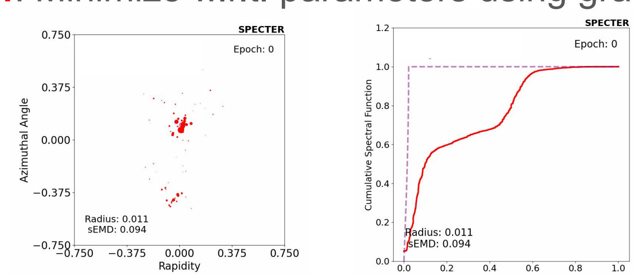


Step 3: Calculate the spectral metric between events and shapes

$$\mathrm{SEMD}_{\beta,p=2}(s_A,s_B) = \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2$$
$$- 2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right)$$
$$\times\, \Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),$$

Key difference from previous work: We use the SEMD, *not* the EMD!

Step 4: Minimize w.r.t. parameters using grads

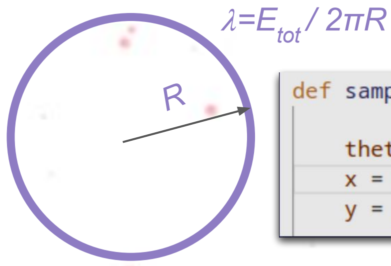


Pictured: Animation of optimizing for the radius R

Full Example: How "ring-like" are jets?

SPECTER Our code framework for these calculations

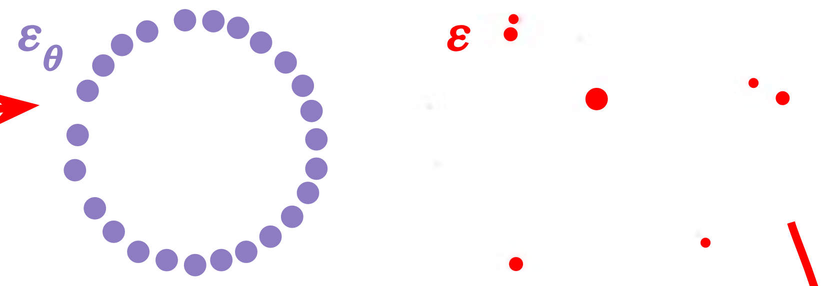
Step 1: Define the shape with parameters




```
def sample_circle(params, N, seed):

    thetas = jax.random.uniform(seed, shape
    x = params["Radius"] * jnp.cos(thetas)
    y = params["Radius"] * jnp.sin(thetas)
```

Unlike ordinary EMD, not necessary to specify center / orientation!

Step 2: Sample from Parameterized Shapes

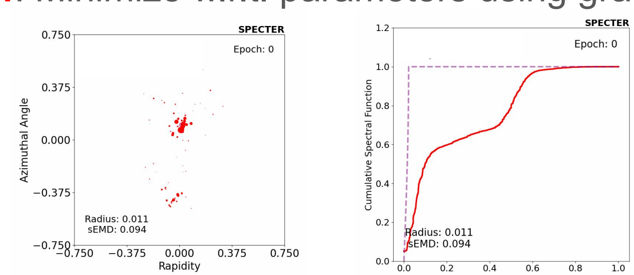


Step 3: Calculate the spectral metric between events and shapes

$$\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2$$
$$-2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l\left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right)$$
$$\times\,\Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),$$

Key difference from previous work: We use the SEMD, *not* the EMD!

Step 4: Minimize w.r.t. parameters using grads



Pictured: Animation of optimizing for the radius R

SPECTER is our code interface for performing these steps: sampling from user-defined shapes, calculating spectral functions and differentiable EMDS, and optimizing over parameters.

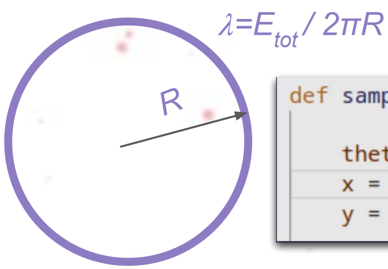
Built in highly parallelized and compiled JAX

SPECTER is a "sequel" to SHAPER, introduced last ML4Jets. SPECTER is *not* an acronym, don't ask me what it stands for.

Full Example: How “ring-like” are jets?

SPECTER Our code framework for these calculations

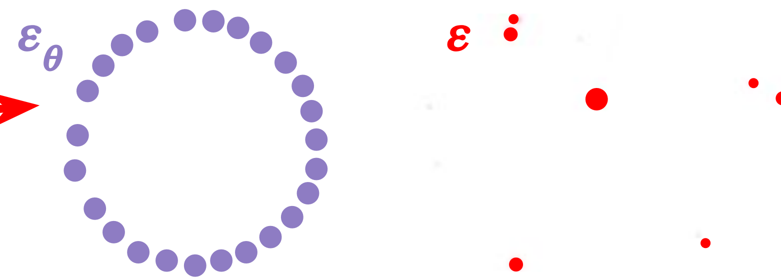
Step 1: Define the shape with parameters




```
def sample_circle(params, N, seed):

    thetas = jax.random.uniform(seed, shape
    x = params["Radius"] * jnp.cos(thetas)
    y = params["Radius"] * jnp.sin(thetas)
```

Unlike ordinary EMD, not necessary to specify center / orientation!

Step 2: Sample from Parameterized Shapes

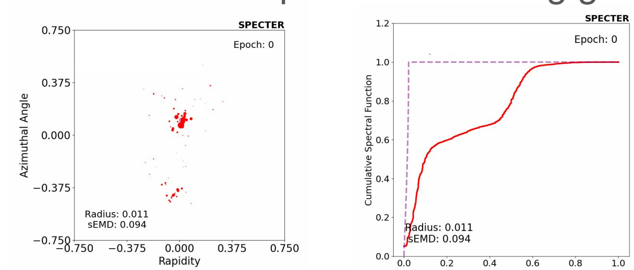


Step 3: Calculate the spectral metric between events and shapes

$$
\begin{aligned}
\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) = &\sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2 \\
&- 2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right) \\
&\times \Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),
\end{aligned}
$$

Key difference from previous work: We use the SEMD, *not* the EMD!

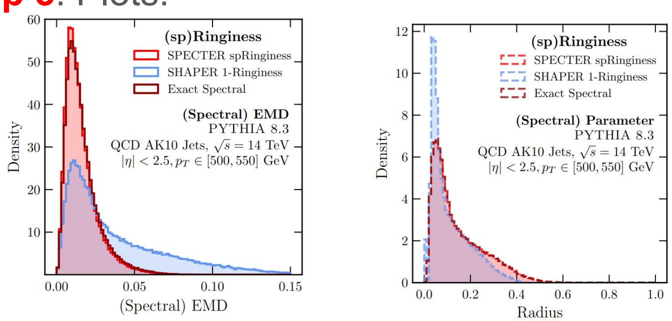
Step 4: Minimize w.r.t. parameters using grads



Pictured: Animation of optimizing for the radius R

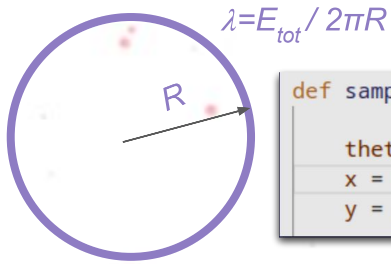
Pictured: 100k Jets, PYTHIA 8 QCD Jets

Step 5: Plots!



Full Example: How "ring-like" are jets?

Step 1: Define the shape with parameters



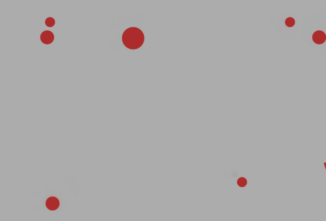

```
def sampl
    theta
    x = p
    y = p
```

Unlike ordinary EMD, not necessary to s

SPECTER Our code framework for these calculations

SPECTER

Step 2: Sample from Parameterized Shapes



Alternatively…

The spectral EMD, *and its optimization*, are often partially or ***completely solvable*** in closed form!

$$R_{\text{opt}} = \frac{2}{\pi} \sum_{\substack{n \in \mathcal{E}^2 \\ \omega_n < \omega_{n+1}}} \omega_n \left[\sin\left(\frac{\pi}{2E_{\text{tot}}^2} \sum_{\substack{n \leq m \in \mathcal{E}^2 \\ \omega_m < \omega_{m+1}}} (2EE)_m \right) - \sin\left(\frac{\pi}{2E_{\text{tot}}^2} \sum_{\substack{n < m \in \mathcal{E}^2 \\ \omega_m < \omega_{m+1}}} (2EE)_m \right) \right]$$

$$\text{SEMD}_{\beta, p=2}(s, s_{\text{jet ring}}) = \sum_{i<j \in \mathcal{E}} 2E_i E_j \omega_{ij}^2 - 2E_{\text{tot}}^2 R_{\text{opt}}^2$$

For many shapes, we can completely short circuit having to perform expensive optimization over an optimal transport problem entirely!

e spectral metric between

$$\ldots_{\mathcal{E}_A} 2E_i E_j \omega_{ij}^2 + \sum_{i<j \in \mathcal{E}_B} 2E_i E_j \omega_{ij}^2$$

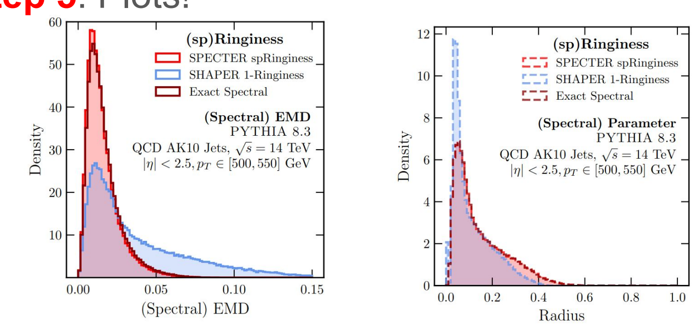
$$\ldots(\omega_n^+), S_B(\omega_l^+)] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right])$$

$$\ldots - S_B(\omega_l^-))\, \Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right),$$

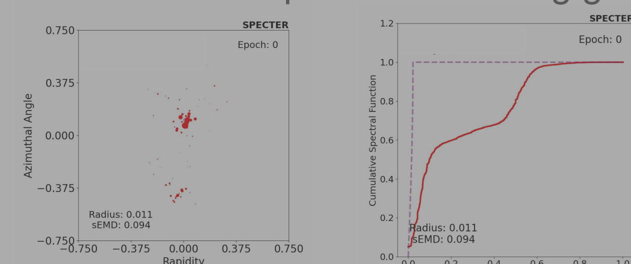
Key difference from previous work: We use the SEMD, *not* the EMD!

Pictured: 100k Jets, PYTHIA 8 QCD Jets

Step 5: Plots!



Step 4: Minimize w.r.t. parameters using grads



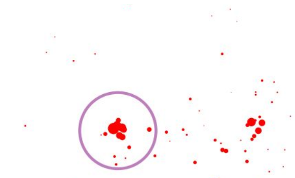
Pictured: Animation of optimizing for the radius R

[1][Ba, Dogra, **RG**, Tasissa, Thaler, 2302.12266]

To distinguish SEMD observables from EMD observables, I will add "s" or "sp"

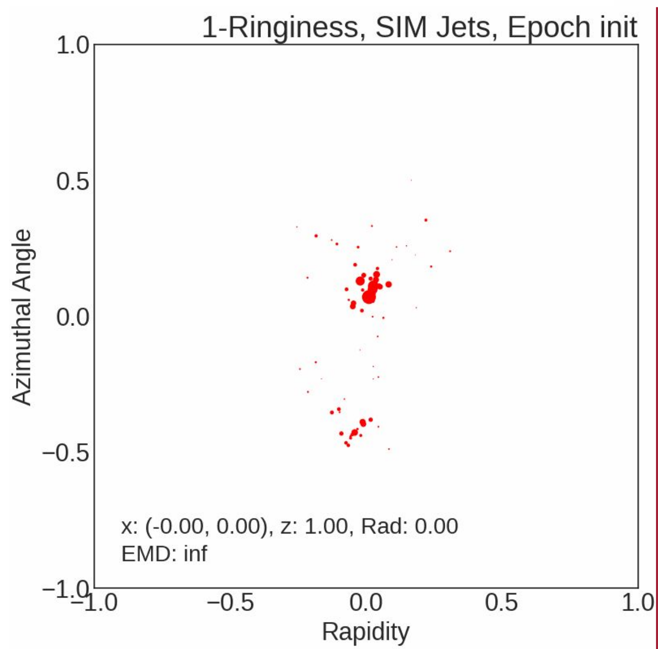
Hearing Jets (sp)Ring

Small R Jet



Large R Jet

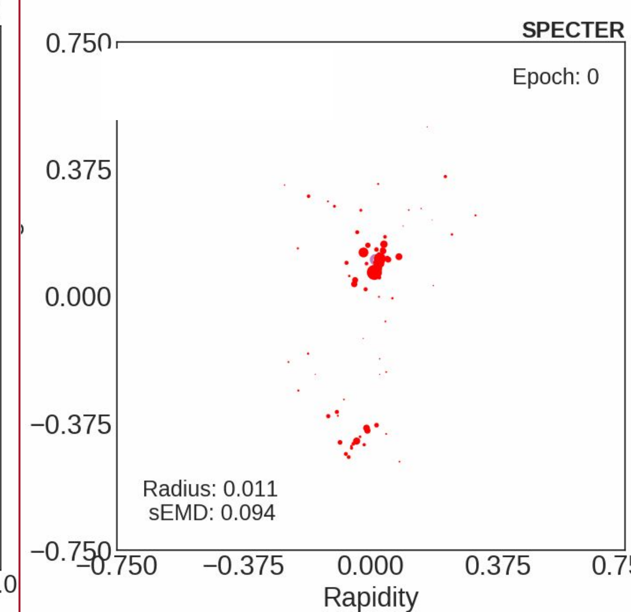
EMD

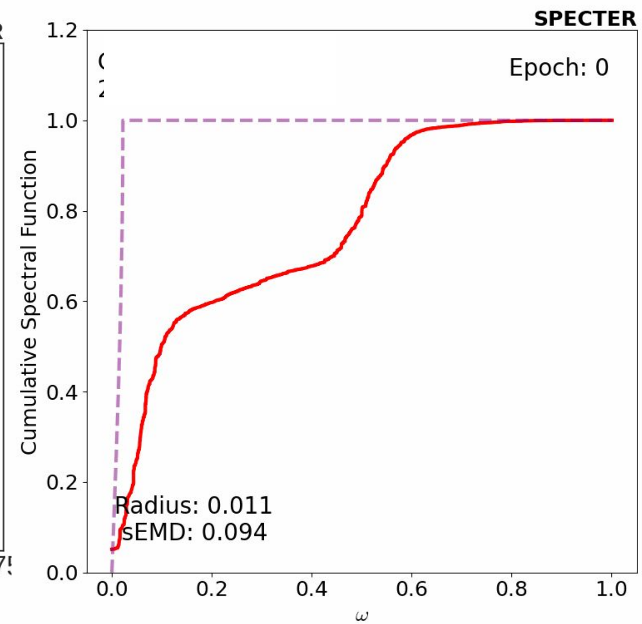


Calculated using SHAPER[1]

Position of ring must be optimized – can use as jet algorithm

Spectral EMD





Calculated using ***SPECTER***

Translationally invariant – no need to optimize over position

Secretly a 1D optimal transport problem over the spectral function

Hearing Jets (sp)Ring

Runtimes (NVIDIA A100 GPU):

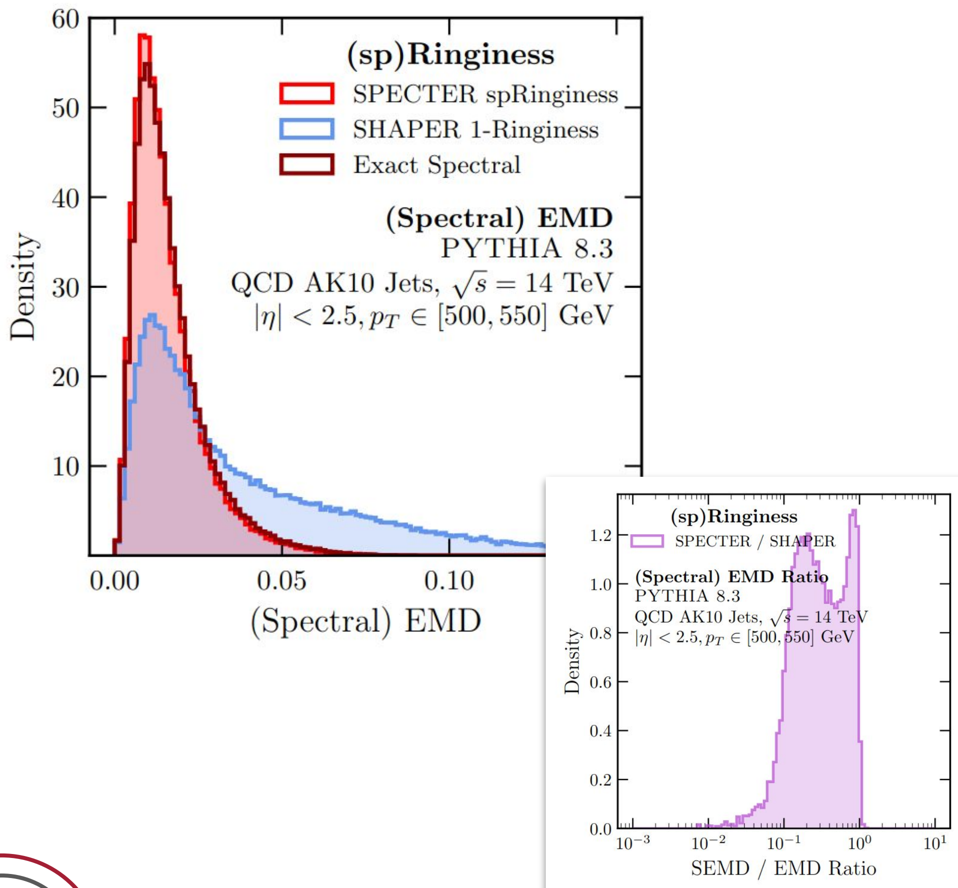
SHAPER (EMD): ~ 36 hours / 100k events

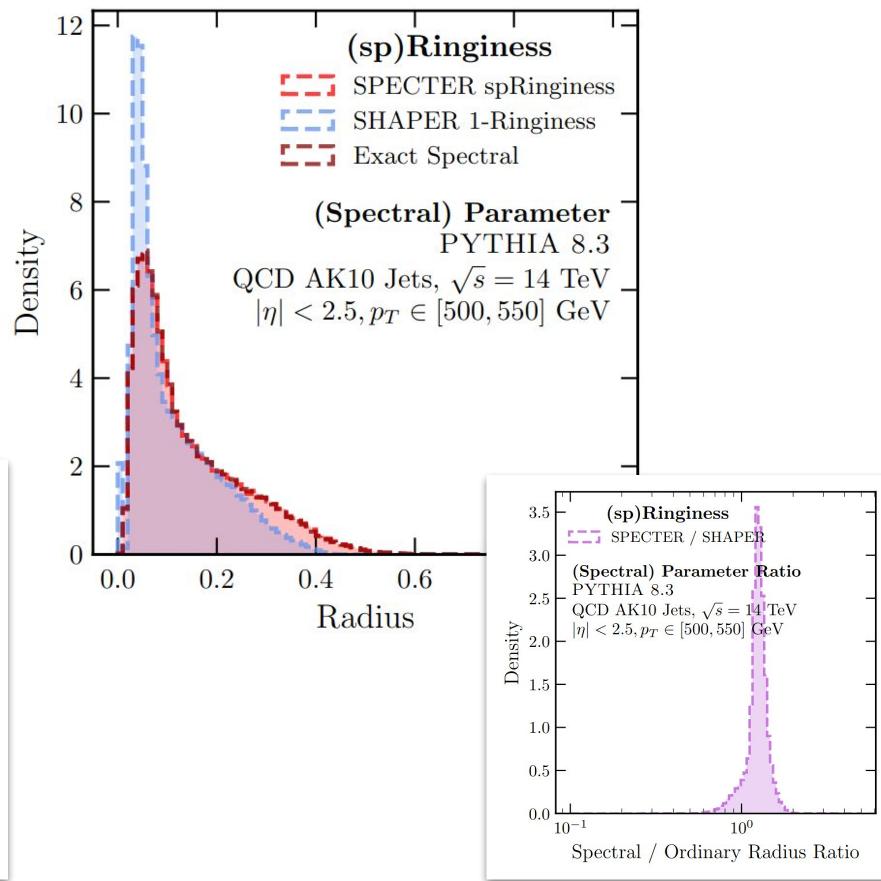
Generalized *SPECTER*: ~55 seconds / 100k events

Closed Form *SPECTER*: ~ < 0.1 seconds / 100k events

The SEMD and EMD are qualitatively different, but give similar radii!

They probe the same event length scale





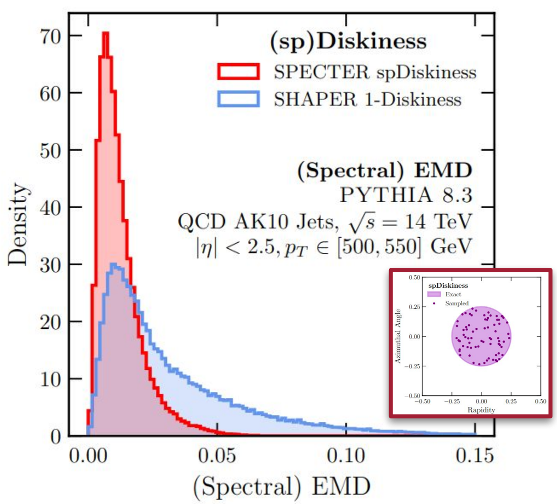
Lots of Observables!

Event and jet shape observables can be defined as the (S)EMD between events and *any* parameterized set of ideal events!

Everything I said today applies to full events on the celestial sphere as well localized jets! Different topologies are possible!

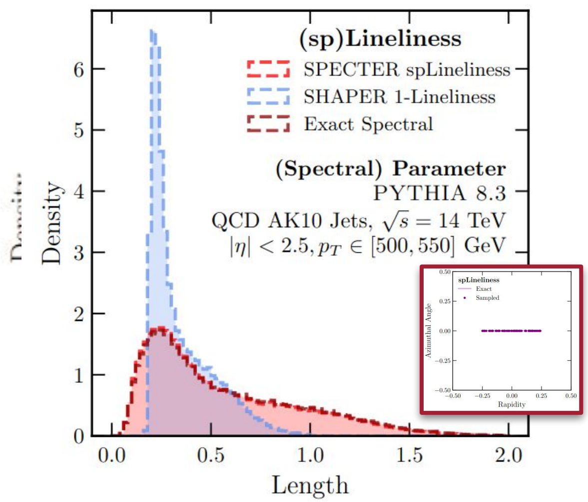
Some examples …

How disk-like are jets?



[No closed form for sDiskiness – not all observables have closed forms]

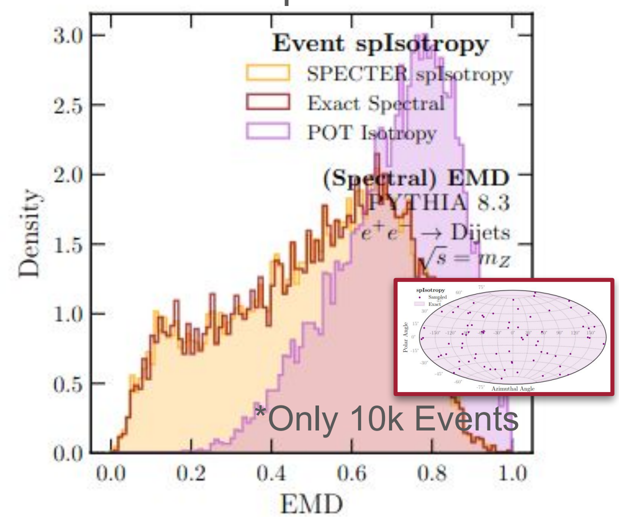
If QCD jets were lines, how long would they be?



$$L_{\text{opt}}(s_\mathcal{E}) = \frac{6}{E_{\text{tot}}^2} \sum_{\substack{n \in \mathcal{E}^2 \\ \omega_n < \omega_{n+1}}} \omega_n \Big[S^+(\omega_n) + \frac{2}{3E_{\text{tot}}} \left(E_{\text{tot}}^2 - S^+(\omega_n)\right)^{3/2} - S^-(\omega_n) - \frac{2}{3E_{\text{tot}}} \left(E_{\text{tot}}^2 - S^-(\omega_n)\right)^{3/2} \Big]$$

Closed-form lines!

How isotropic[1] are events?



$$\mathcal{O}_{\text{Isotropy}}(s_\mathcal{E}) = \sum_{i<j \in \mathcal{E}} 2E_iE_j\omega_{ij}^2 + \frac{\pi^2-4}{2}E_{\text{tot}}^2 - 2 \sum_{\substack{n \in \mathcal{E}^2 \\ \omega_n < \omega_{n+1}}} \omega_n \left[f^+(n) - f^-(n)\right]$$

$$f^\pm(n) = \sqrt{S^\pm(\omega_n)}\sqrt{E_{\text{tot}}^2 - S^\pm(\omega_n)} + S^\pm(\omega_n)\cos^{-1}\left(1 - 2\frac{S^\pm(\omega_n)}{E_{\text{tot}}^2}\right) - E_{\text{tot}}^2 \sin^{-1}\left(\frac{\sqrt{S^\pm(\omega_n)}}{E_{\text{tot}}}\right)$$

Closed-form isotropy!

Things to think about:

- **Speed**: I am not a great computer scientist; *SPECTER* could probably be made *even faster* with more clever and better programming.
- **Degeneracies and Topology**: The EMD and SEMD are different, especially for equilateral triangle configurations – how often do these configurations occur in different theories?
- **Closed form Observables**: Not every shape has a completely closed-form solution, but it is usually possible to partially simplify and reduce the problem to 1D minimization, 1D root finding, or simple 1D numeric integrals. Can we understand this better?
- **Perturbative Calculations**: Closed-form and simple expressions means perturbative calculations may be possible – can we predict the radius of a jet to LO, NLO, LL, NLL, …?
- **Theory Space**: There have been proposals to use the (S)EMD between events as a ground metric for an OT distance between theories. With *SPECTER*, this could now be numerically viable!

Happy to talk with you about any and all of these afterwards!

Conclusion

SPECTER

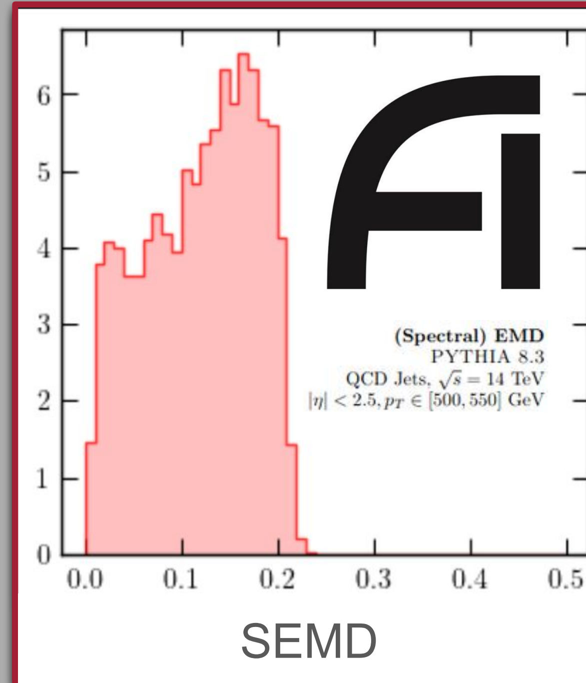
Pictured: The **spectral-IAIFI-ness** of QCD Jets!

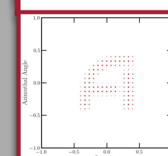
The **spectral EMD** can be used as an alternative to the EMD. It is **fast** and **easy to minimize**.

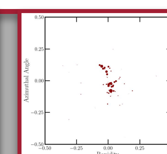
SPECTER is a code package for efficiently evaluating the spectral EMD and calculating shape observables.

With the spectral EMD, many jet observables can be understood in **closed form**.

SPECTER will be *pip*-installable!
Coming soon!







More questions? Email me at rikab@mit.edu

Appendices

The EMD

Definition:

$$\mathrm{EMD}_{\beta,R}(\mathcal{E}_A, \mathcal{E}_B) = \min_{\{f_{ab}\}} \sum_{a\in J_A} \sum_{b\in J_B} f_{ab} \frac{\Omega(\hat{n}_a, \hat{n}_b)^\beta}{R^\beta} + \left| \sum_{a\in J_A} E_a - \sum_{b\in J_B} E_b \right|$$

such that

$$f_{ab} \geq 0\,, \qquad \sum_{b\in J_B} f_{ab} \leq E_a\,, \qquad \sum_{a\in J_A} f_{ab} \leq E_b\,, \qquad \sum_{a\in J_A} \sum_{b\in J_B} f_{ab} = \min\left(\sum_{a\in J_A} E_a, \sum_{b\in J_B} E_b \right)$$

Ground Metrics

For local jets on the rapidity-azimuth plane:

$$\omega_{ij} = \sqrt{(\phi_i - \phi_j)^2 + (y_i - y_j)^2}$$

For global events on the sphere:

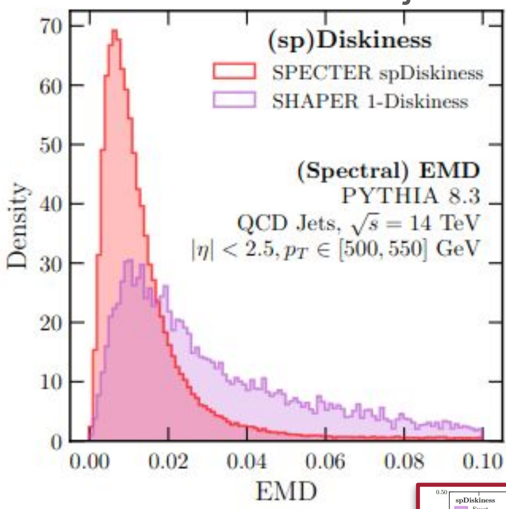
$$\begin{aligned} \omega_{ij} &= |\theta_{ij}| \\ &= \left|\cos^{-1}\left(1 - \frac{p_i \cdot p_j}{E_i E_j}\right)\right| \end{aligned}$$

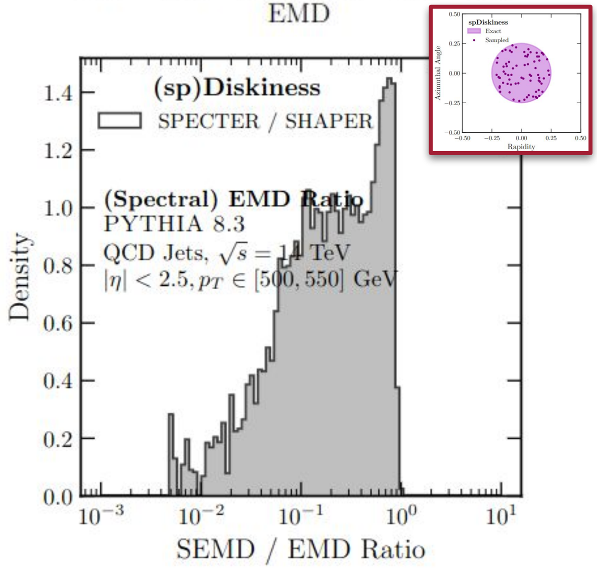
In principle, could have picked chord length rather than arc length

[1][Cesarotti, Thaler, 2004.06125]

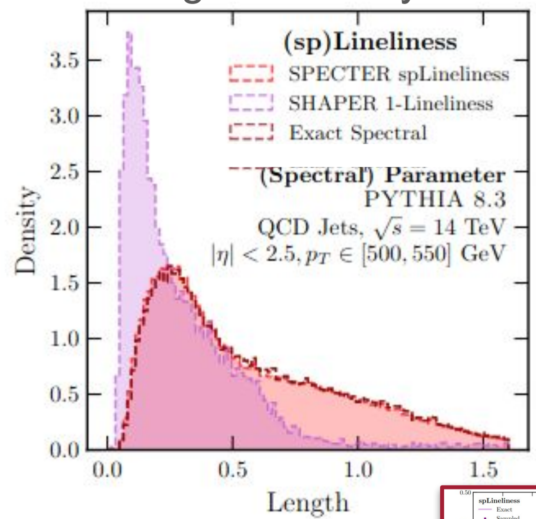
SEMD to EMD Ratios

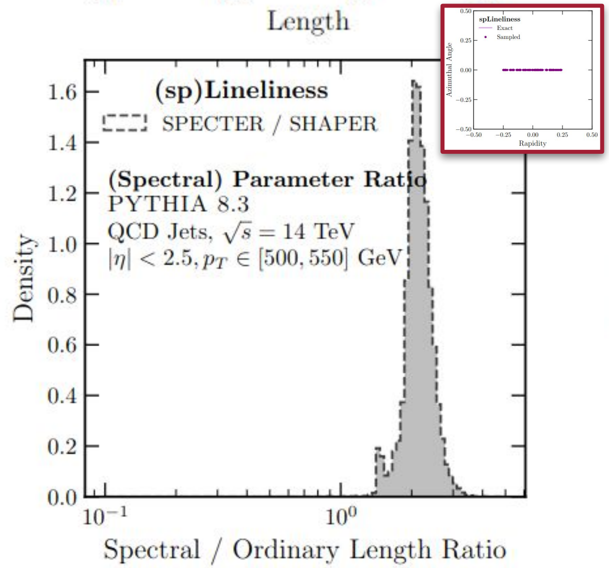
How disk-like are jets?



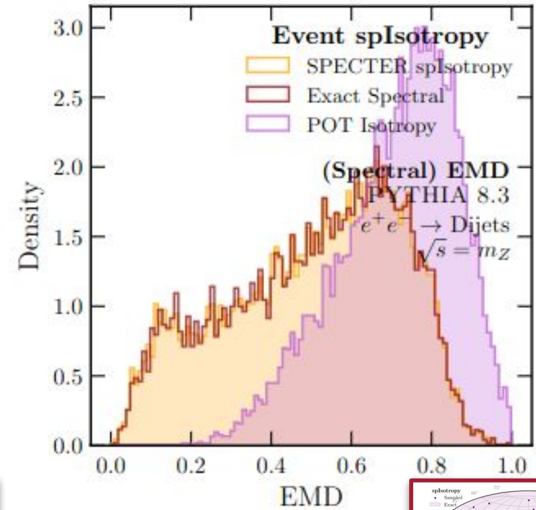


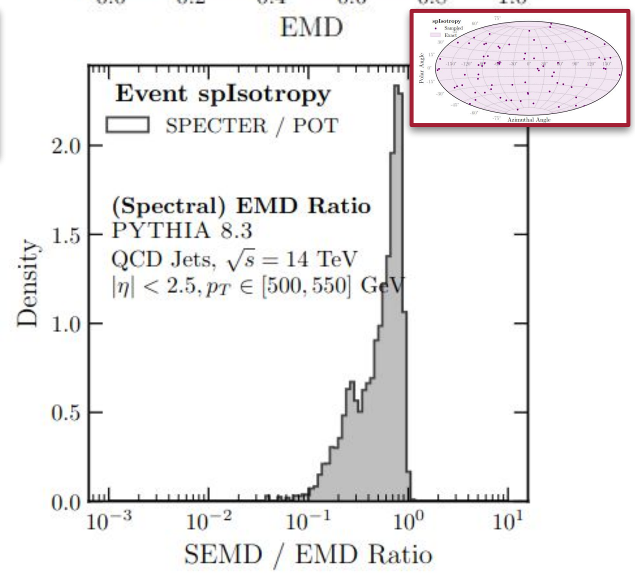
If QCD jets were lines, how long would they be?



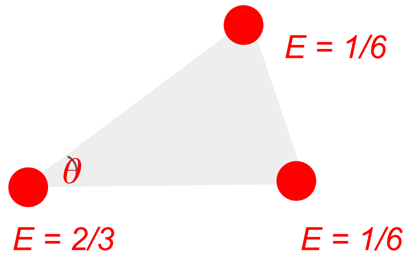


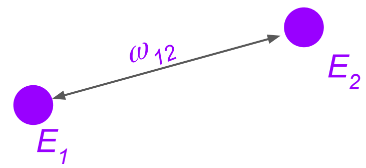
How isotropic[1] are events?

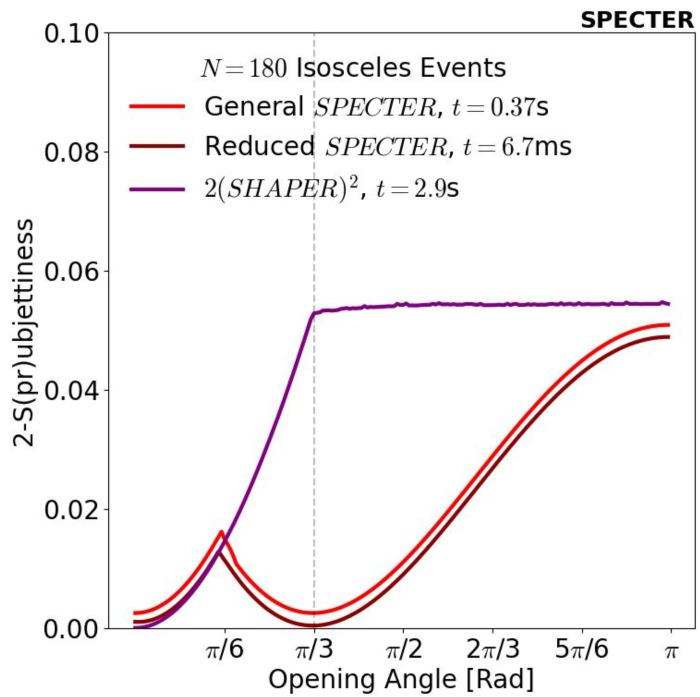




Degeneracies (Continued)







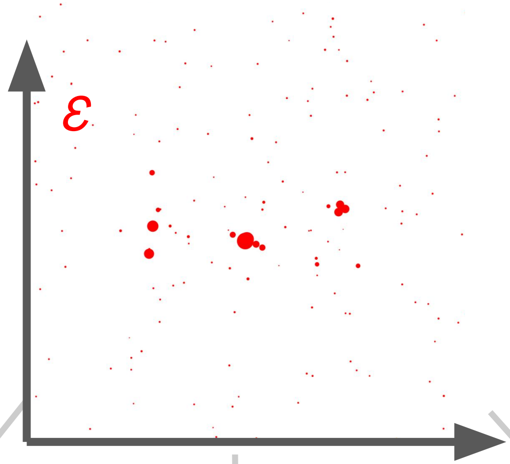
For this precise energy configuration, equilateral triangles are *exactly* degenerate with 2 particle events – so the spectral EMD only sees 2 particles!

Only measure 0 configuration of events – but events *near* this give spectral EMDs *near* zero against 2 particle events.

*with the right energy weights.

Shapiness

The EMD between a real event or jet $\mathcal{E}$ and idealized shape $\mathcal{E}'$ is the [shape]iness of $\mathcal{E}$ – a well defined IRC-safe observable!

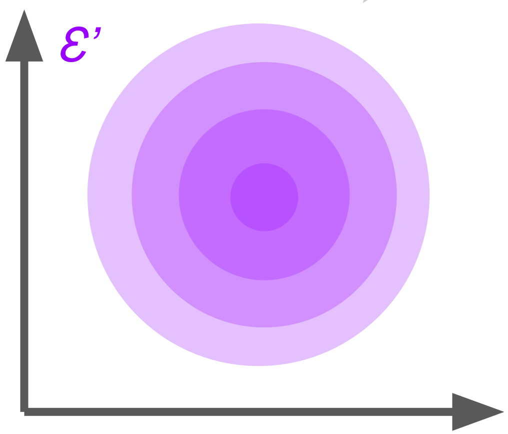


Answers the question: “How much like the shape $\mathcal{E}'$ is my event $\mathcal{E}$?”

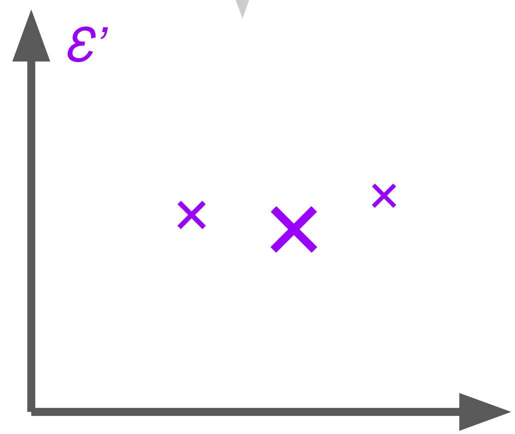
Med EMD($\mathcal{E}$, $\mathcal{E}'$)
“Gausiness”

Low EMD($\mathcal{E}$, $\mathcal{E}'$)
“3-Pointiness”
AKA “3-Subjettiness”

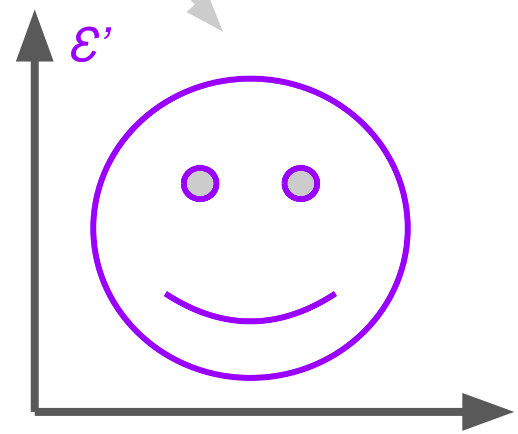
High EMD($\mathcal{E}$, $\mathcal{E}'$)
“Smileyness”



Shape = 2D Gaussian



Shape = 3 Points



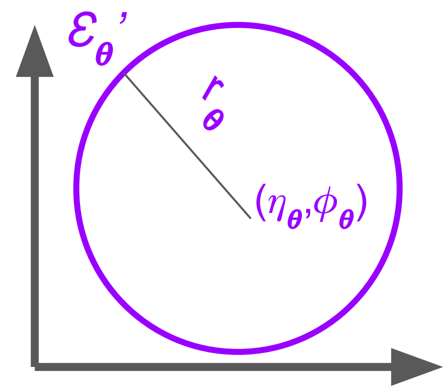
Shape = Smile

Mathematical Details - Shapiness

Rather than a single shape, consider a **parameterized manifold** $\mathcal{M}$ of **energy flows**.

e.g. The manifold of uniform circle energy flows:

$$\mathcal{E}_\theta'(y) = \begin{cases} \frac{1}{2\pi r_\theta} & |\vec{y} - \vec{y}_\theta| = r_\theta \\ 0 & |\vec{y} - \vec{y}_\theta| \neq r_\theta \end{cases}$$



Then, for an event $\mathcal{E}$, define the **shapiness** $\mathcal{O}_\mathcal{M}$ and **shape parameters** $\theta_\mathcal{M}$, given by:

$$\mathcal{O}_\mathcal{M}(\mathcal{E}) \equiv \min_{\mathcal{E}_\theta \in \mathcal{M}} \text{EMD}^{(\beta,R)}(\mathcal{E}, \mathcal{E}_\theta)$$

$$\theta_\mathcal{M}(\mathcal{E}) \equiv \underset{\mathcal{E}_\theta \in \mathcal{M}}{\text{argmin}}\, \text{EMD}^{(\beta,R)}(\mathcal{E}, \mathcal{E}_\theta)$$

[P. Komiske, E. Metodiev, and J. Thaler, 2004.04159;
J. Thaler, and K. Van Tilburg, 1011.2268;
I. W. Stewart, F. J. Tackmann, and W. J. Waalewijn, 1004.2489.;
S. Brandt, C. Peyrou, R. Sosnowski and A. Wroblewski, PRL 12 (1964) 57-61;
C. Cesarotti, and J. Thaler, 2004.06125]

Observables ⇔ Manifolds of Shapes

Observables can be specified solely by defining a **manifold of shapes**:

$$\mathcal{O}_{\mathcal{M}}(\mathcal{E}) \equiv \min_{\mathcal{E}_\theta \in \mathcal{M}} \mathrm{EMD}^{(\beta,R)}(\mathcal{E}, \mathcal{E}_\theta),$$

$$\theta_{\mathcal{M}}(\mathcal{E}) \equiv \operatorname*{argmin}_{\mathcal{E}_\theta \in \mathcal{M}} \mathrm{EMD}^{(\beta,R)}(\mathcal{E}, \mathcal{E}_\theta),$$

Many well-known observables* already have this form!

Observable	**Manifold of Shapes**
N-Subjettiness	Manifold of *N*-point events
N-Jettiness	Manifold of *N*-point events with floating total energy
Thrust	Manifold of back-to-back point events
Event / Jet Isotropy	Manifold of the single uniform event

… and more!

All of the form "How much like **[shape]** does my **event** look like?"
Generalize to *any* shape.

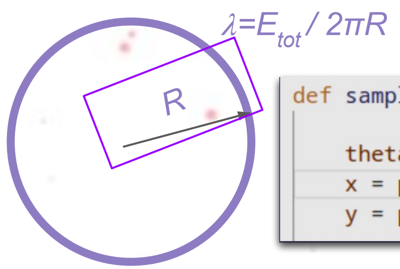
*These observables are usually called event shapes or jet shapes in the literature – we are making this literal!

Full Example: How “ring-like” are jets?

SPECTER Our code framework for these calculations

SPECTER

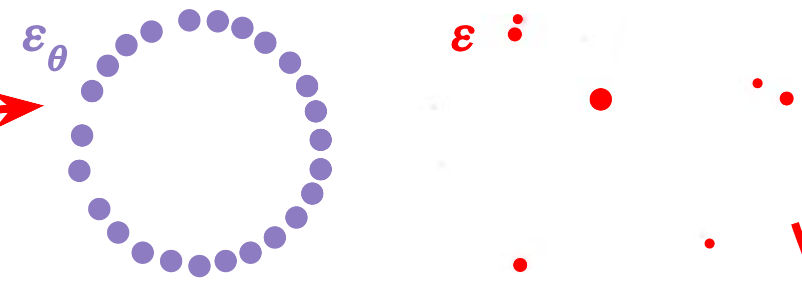
Step 1: Define the shape with parameters




```
def sample_circle(params, N, seed):

    thetas = jax.random.uniform(seed, shape
    x = params["Radius"] * jnp.cos(thetas)
    y = params["Radius"] * jnp.sin(thetas)
```

Unlike ordinary EMD, not necessary to specify center / orientation!

Step 2: Sample from Parameterized Shapes

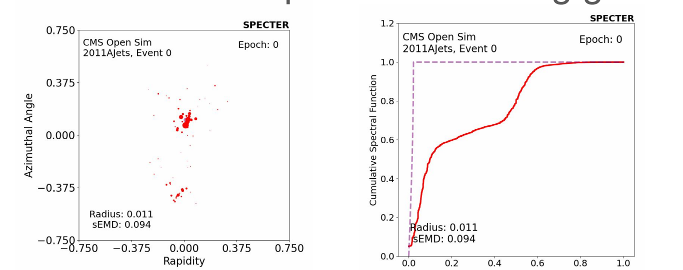


Step 3: Calculate the spectral metric between events and shapes

$$
\begin{aligned}
\mathrm{SEMD}_{\beta,p=2}(s_A, s_B) &= \sum_{i<j\in\mathcal{E}_A} 2E_iE_j\omega_{ij}^2 + \sum_{i<j\in\mathcal{E}_B} 2E_iE_j\omega_{ij}^2 \\
&- 2\sum_{n\in\mathcal{E}_A^2,\, l\in\mathcal{E}_B^2} \omega_n\omega_l \left(\min\left[S_A(\omega_n^+), S_B(\omega_l^+)\right] - \max\left[S_A(\omega_n^-), S_B(\omega_l^-)\right]\right) \\
&\times \Theta\left(S_A(\omega_n^+) - S_B(\omega_l^-)\right)\Theta\left(S_B(\omega_l^+) - S_A(\omega_n^-)\right) ,
\end{aligned}
$$

Key difference from previous work: We use the SEMD, *not* the EMD!

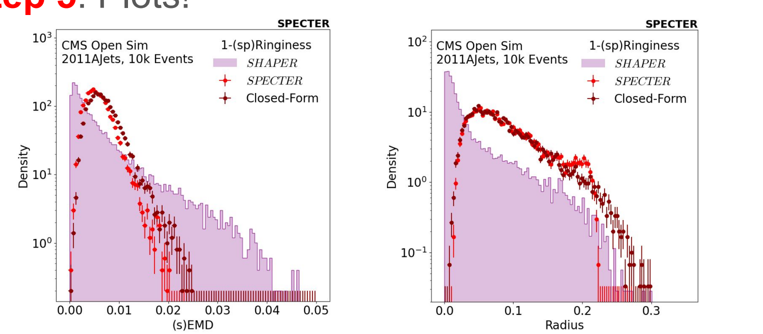
Step 4: Minimize w.r.t. parameters using grads



Pictured: Animation of optimizing for the radius R

Pictured: 10k Jets, CMS 2011AJets Open Sim

Step 5: Plots!



CPU Runtimes