Building the STXS Parameterisation

Charlotte Knight

26th April 2024

LPC EFT Workshop 2024

Outline

Quick STXS reminder

↓

Derivation of STXS parameterisation
(some highlights)

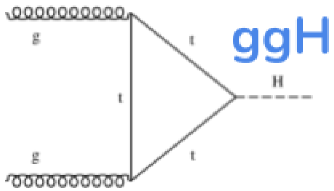
↓

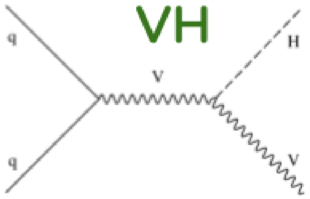
A common STXS parameterisation
+ common EFT parameterisation format
(towards a library)

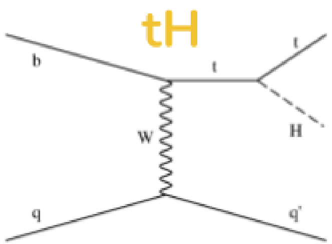
STXS (stage 1.2)

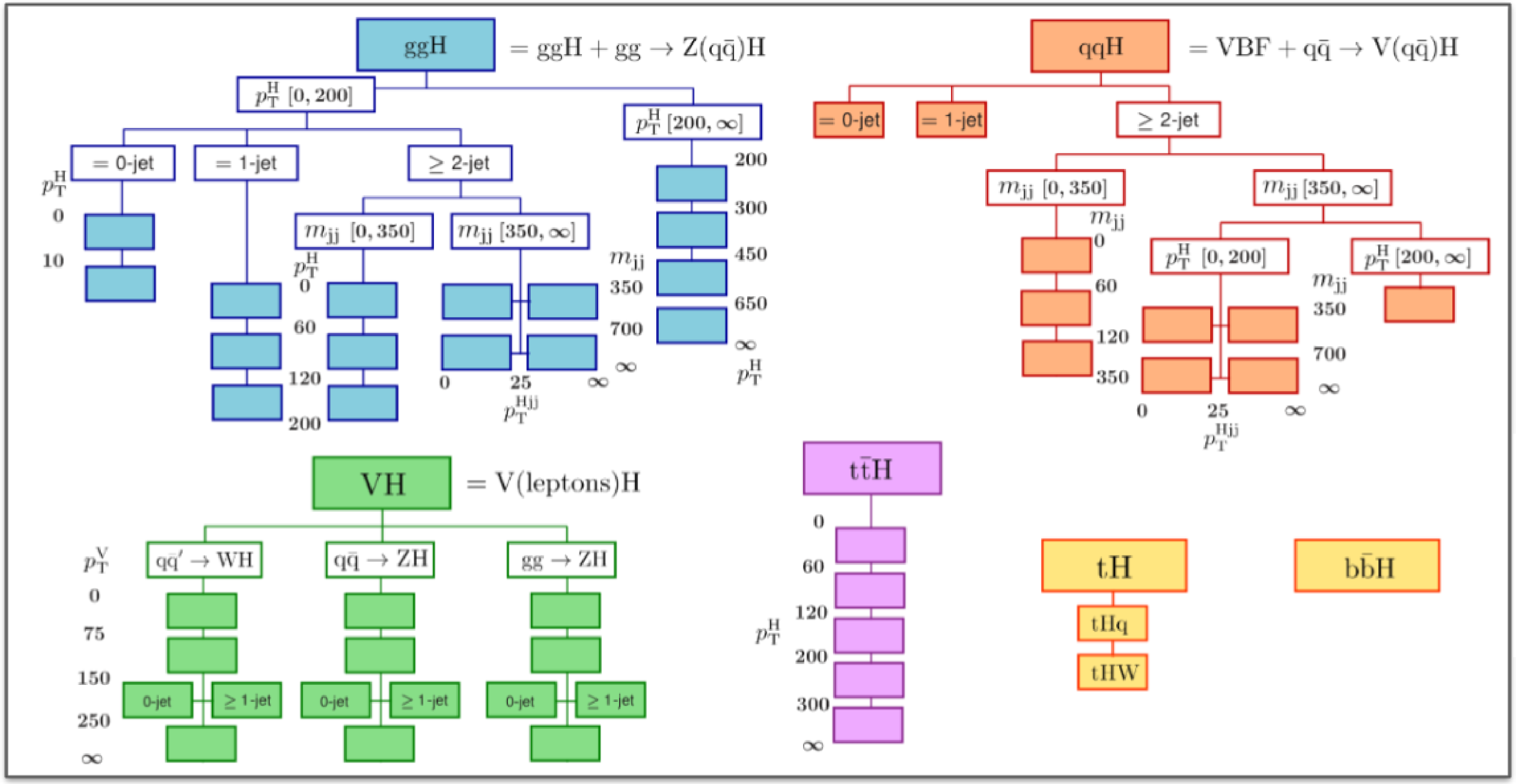
See Jonathon's talk earlier for fitting details

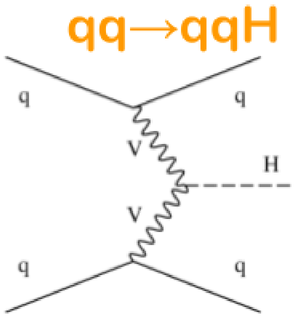
Split by p_T^H, n_{jets}, m_{jj}, p_T^{Hjj}, p_T^V

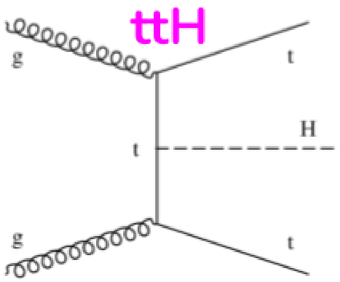












Advantages

- Common scheme across decay channels (eases combination)
- Systematically reduce theory dependence in measurements
- Isolate regions with enhanced BSM sensitivity
- Framework for BSM interpretations (e.g. SMEFT)

Introduction – the parameterisation

Scaling equation, μ, for:

1. Every STXS stage 1.2 bin
2. Partial decay widths
3. Total Higgs width

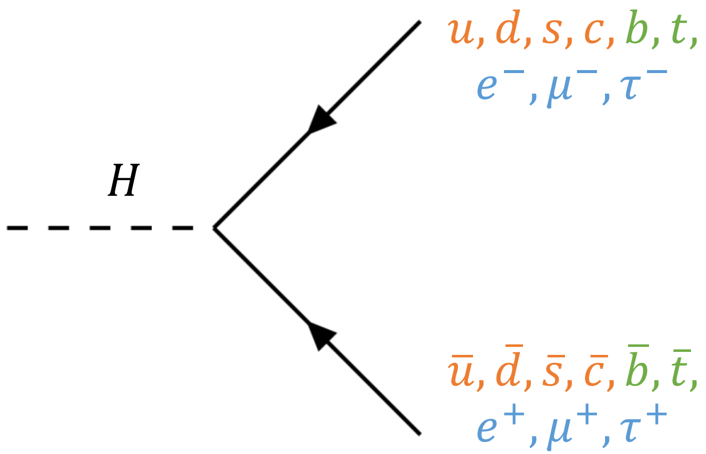
$$\mu = 1 + \sum_i A_i C_i + \sum_{ij} B_{ij} C_i C_j$$

- Use only dimension-6 operators, in Warsaw basis
- Single insertions including all CP-even and odd operators
  - Though we only use CP-even in the fit
- Linear in amplitude → keep quadratic terms* from $|M|^2$
- `topU3l` flavour symmetry[1]: $U(2)^3_{q,u,d}U(3)^2_{l,e}$
  - Light quarks (u,d,s,c) and heavy quarks (b,t) have separate Wilson coefficients
  - All leptons share same Wilson coefficients
- $\{G_F, m_Z, m_W\}$ input parameter scheme[1]

$$Q_{uH} = (H^\dagger H)(\bar{q} Y_u^\dagger u \widetilde{H})$$
$$Q_{dH} = (H^\dagger H)(\bar{q} Y_d^\dagger d H)$$

$$Q_{bH} = (H^\dagger H)(\bar{Q} H b)$$
$$Q_{tH} = (H^\dagger H)(\bar{Q} \widetilde{H} t)$$

$$Q_{eH} = (H^\dagger H)(\bar{l}_p e_r H)$$

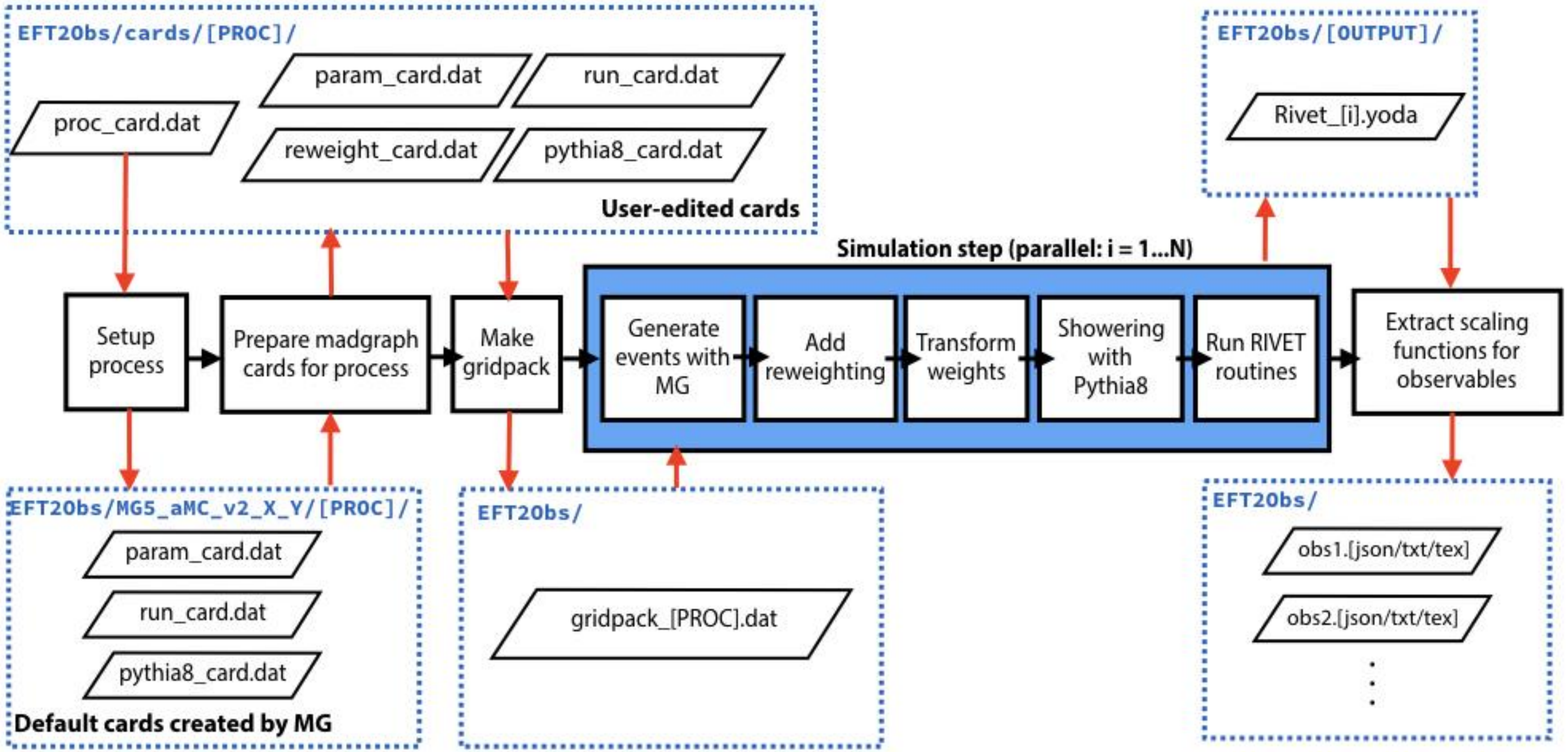


*gives us the *option* to use it…

[1] More details in SMEFTsim manual

EFT2Obs

Wrapper around `MadGraph5_aMC` to help automate derivation of EFT parameterisation



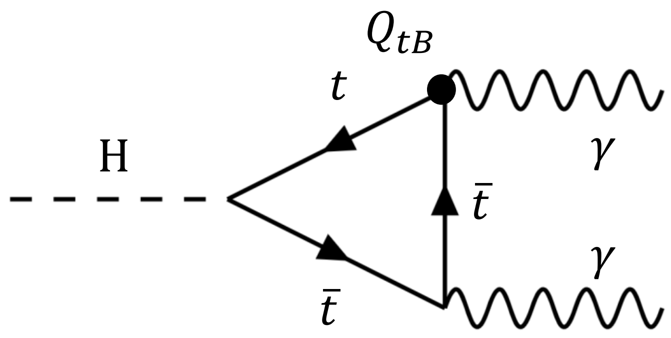
Public tool
[GitHub]

Gen-level parameterisations

- Begin with a gen-level parameterization
  - Derive scaling equations only considering STXS bin definition and no additional selection criteria imposed by analyses
  - Expect to be valid where:
    1. Kinematic-independent effects (e.g. $H \to 2$ body decays)
    2. Bin finely enough in relevant kinematic variables (what the STXS is supposed to be)
  - Tools to derive equations with full selection criteria discussed later

- Analytical equations for $H \to \gamma\gamma$ (1807.11504) and $H \to Z\gamma$ (1801.01136)
  - Include one-loop EW corrections

    $\to$ probe more couplings in loops



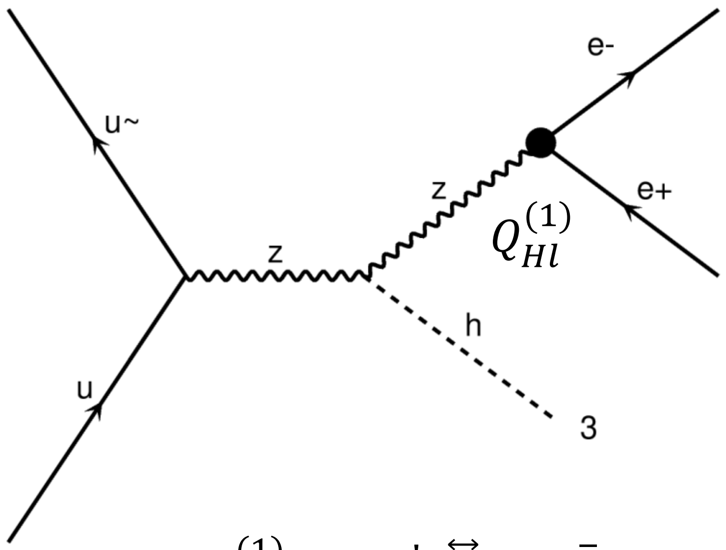
- Use EFT2Obs tool for all other production and decay modes

Propagator corrections

- We have our typical 'vertex' EFT effects
  - e.g. $Q_{Hl}^{(1)}$ in the Zee vertex of $qq\mathrm{ZH}$ $(\mathrm{Z} \to ll)$
- In this diagram, also have term $\sim \frac{1}{q^2 - m_Z^2 + i m_Z \Gamma_Z}$
  - Z width, Γ_Z, is dependent on C_i

  → additional EFT effects introduced via propagators

$$\mu_Z = \frac{\Gamma_Z}{\Gamma_Z^{SM}} = 1 - 0.011 c_{Hl}^{(1)} + \cdots$$



$$Q_{Hl}^{(1)} = (H^\dagger i \overleftrightarrow{D}_\mu H)(\bar{l}_p \gamma^\mu l_r)$$

- Propagator corrections can be significant
  - Without corrections: $\mu_{WHlep} = 1 - 0.039 c_{Hl}^{(1)} + \cdots$

At linear order, take term from μ_Z and subtract

→ 28% reduction

  - With corrections: $\mu_{WHlep} = 1 - 0.028 c_{Hl}^{(1)} + \cdots$
- We apply corrections to processes modelled with `SMEFTsim` (everywhere except ggH and ggZH)

Higgs total width

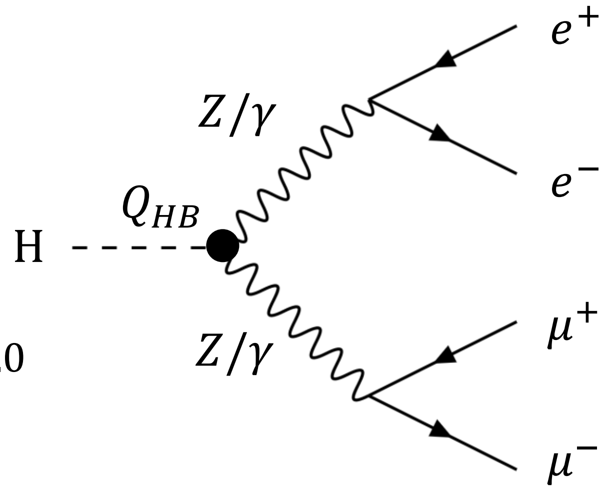
- Total width of the Higgs enters parameterisation of every measurement

$$\mu = \frac{\sigma \times BR}{(\sigma \times BR)_{SM}} = \frac{\sigma}{\sigma_{SM}} \times \left(\frac{\Gamma_i}{\Gamma_i^{SM}} / \frac{\Gamma_H}{\Gamma_H^{SM}} \right)$$

- Analytical solution exists to linear order in Wilson coefficients ([1906.06949]) but we want parameterisation to quadratic order

  → derive total width using MC methods to quadratic order and cross check linear terms as validation

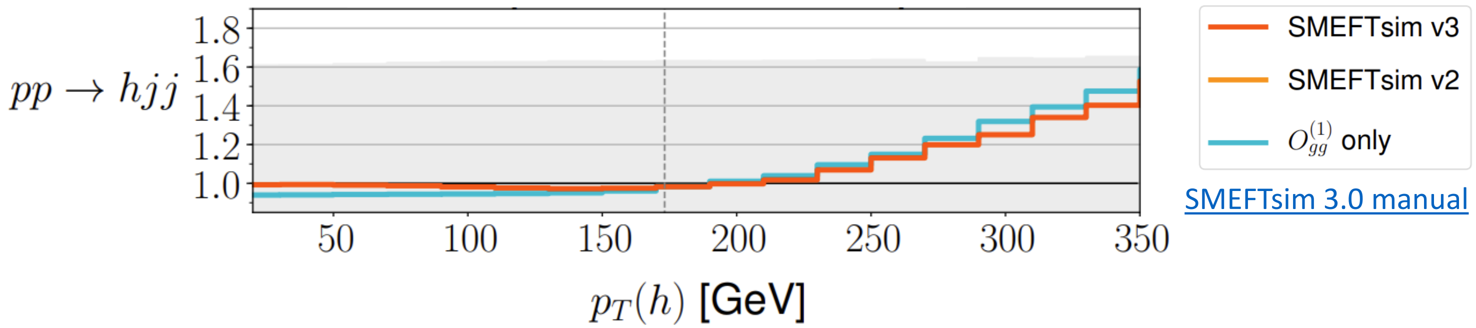
- Derivations for $H \rightarrow 2$ body immediately agreed with analytical result
- Reweighting failed in $H \rightarrow 4f$ for c_{HW}, c_{HB}, c_{HWB}
  - ‘Divergences’ at low m_{ll} due to γ propagator → high uncert. in A_i and B_{ij}
  - Use hybrid approach
    - Derive scaling equations with reweighting for all Wilson coefficients
    - Create dedicated generations for c_{HW}, c_{HB}, c_{HWB} reweighting points, e.g. $c_{HW} = 0{,}0.5{,}1.0$
    - Replace c_{HW}, c_{HB}, c_{HWB} terms with derivations from dedicated generation



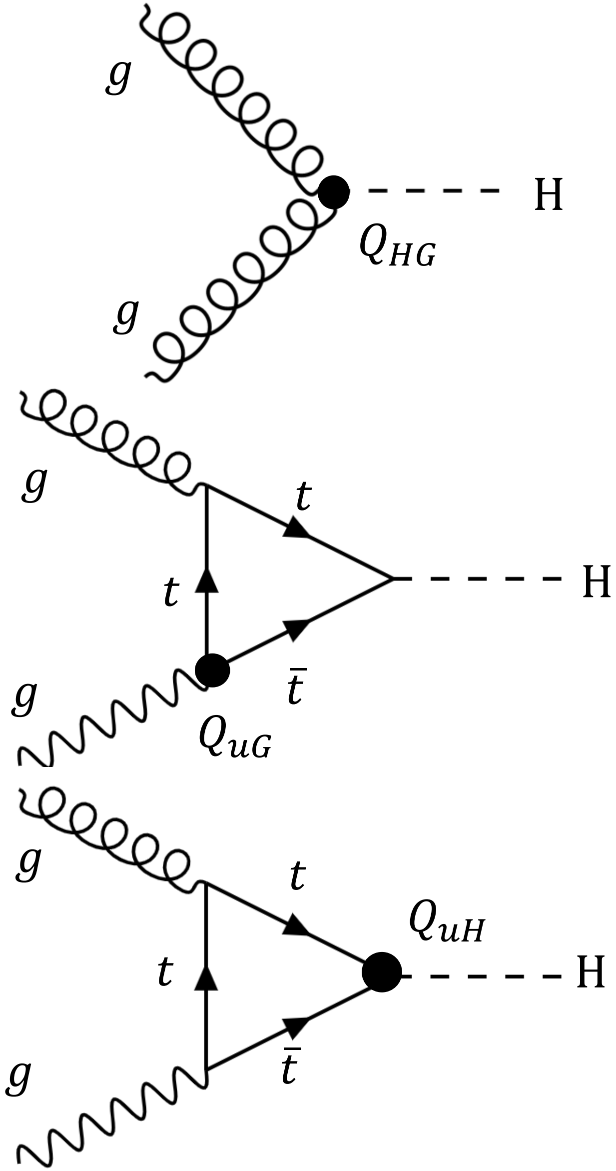
SMEFTsim for ggH

- LO SMEFT calculations (no loop-level)
- Some SM loop-induced processes have EFT tree diagrams
- Effective vertices for $ggH, H\gamma\gamma, HZ\gamma$
  - Allows for approximate parameterisations
  - Invalid for $p_T^H > m_t$

Effective vertex / SM 1-loop



SMEFTsim 3.0 manual



- Only Q_{HG} - No access to Q_{tG}, Q_G and $Q_{tH}\ Q_{H\Box}, Q_{HD}, Q_{Hl}^{(3)}, Q_{ll}$

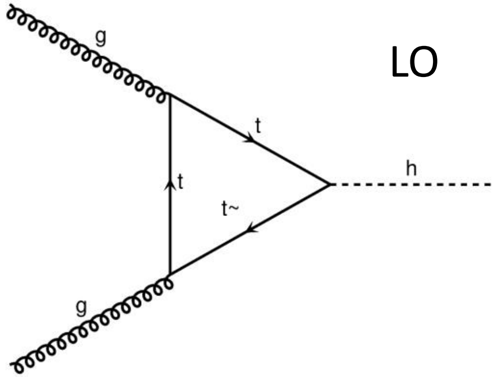
Kinematic dependence (Q_{tG}, Q_G) Flat rescaling ($Q_{tH}\ Q_{H\Box}, Q_{HD}, Q_{Hl}^{(3)}, Q_{ll}$)

- No effective vertex for $ggZH$ → no parameterisation at all

ggH with SMEFT@NLO

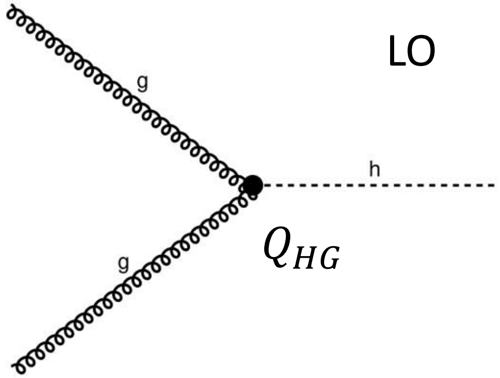
SMEFT@NLO: SMEFT at one-loop in QCD

- Solves our physics problems… but creates technical challenge
- The loop-induced diagram for ggH is LO in SM
- We are not deriving a NLO EFT parameterization
  - Just using the NLO machinery (handling of loops)

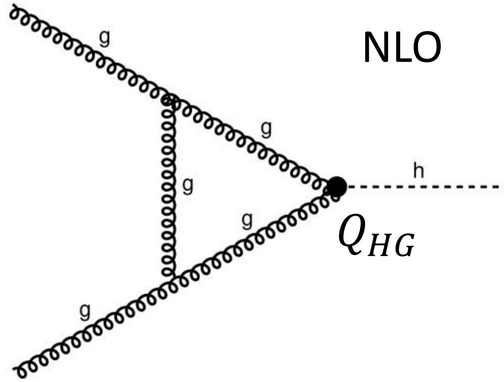


- Generating ggH at loop-level

```
generate p p > h NP=2 QED=1 QCD=2 [virt=QCD]
```

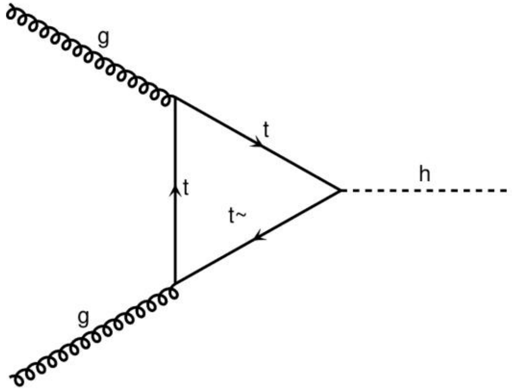


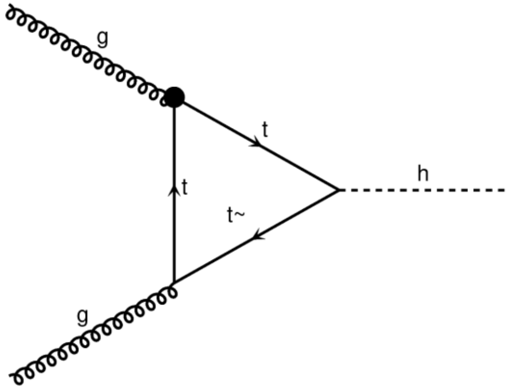
- Get LO and NLO diagrams from Q_{HG}
  - Want to exclude the NLO diagrams
  - Cannot exclude NLO Q_{HG} whilst keeping everything else using a single process
    - → splitting contributions

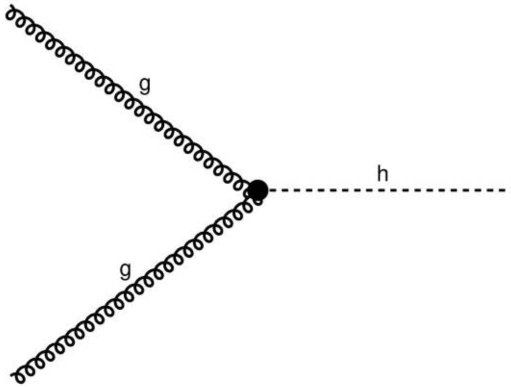


What contributions do we need?

$$\mathcal{M} = \mathcal{M}_{loop}^{SM} + \mathcal{M}_{loop}^{EFT} + \mathcal{M}_{tree}^{EFT}$$



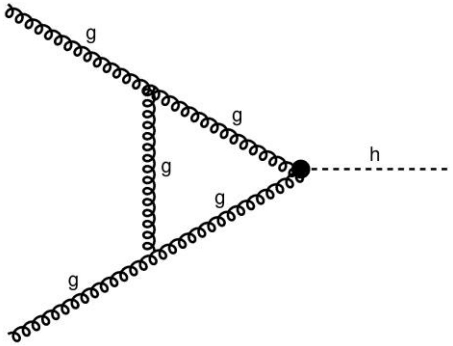




$$|\mathcal{M}|^2 = \left|\mathcal{M}_{loop}^{SM}\right|^2 + \left|\mathcal{M}_{loop}^{EFT}\right|^2 + 2Re\left(\mathcal{M}_{loop}^{SM}\mathcal{M}^{*EFT}_{loop}\right) + \left|\mathcal{M}_{tree}^{EFT}\right|^2 + 2Re\left(\mathcal{M}_{loop}^{SM}\mathcal{M}^{*EFT}_{tree}\right) + 2Re\left(\mathcal{M}_{loop}^{EFT}\mathcal{M}^{*EFT}_{tree}\right)$$

Unwanted contribution

$\mathcal{M}_{loop}^{EFT,NLO}$



Contributions

- Four different processes
- Derive separate scaling equations for each
- Combine equations (add terms)

Loop

Exclude Q_{HG} via restrict card

```
import model SMEFTatNLO-ggH_no_cpg
p p > h NP=2 QED=1 QCD=2 [virt=QCD]
```

$|\mathcal{M}^{SM}_{loop}|^2$ $|\mathcal{M}^{EFT}_{loop}|^2$ $\mathcal{M}^{SM}_{loop}\mathcal{M}^{*EFT}_{loop}$

Tree

```
p p > h NP=2 QED=1 QCD=0
```

$|\mathcal{M}^{EFT}_{tree}|^2$

Tree_loop_2

```
p p > h NP=2 QCD=0 QED=1 QCD^2==2 NP^2==2
[virt=QCD]
```

$\mathcal{M}^{SM}_{loop}\mathcal{M}^{*EFT}_{tree}$

Tree_loop_4

```
p p > h NP=2 QCD=0 QED=1 QCD^2==2 NP^2==4
[virt=QCD]
```

Unwanted contribution

$\mathcal{M}^{EFT}_{loop}\mathcal{M}^{*EFT}_{tree}$ $\mathcal{M}^{EFT,NLO}_{loop}\mathcal{M}^{*EFT}_{tree}$

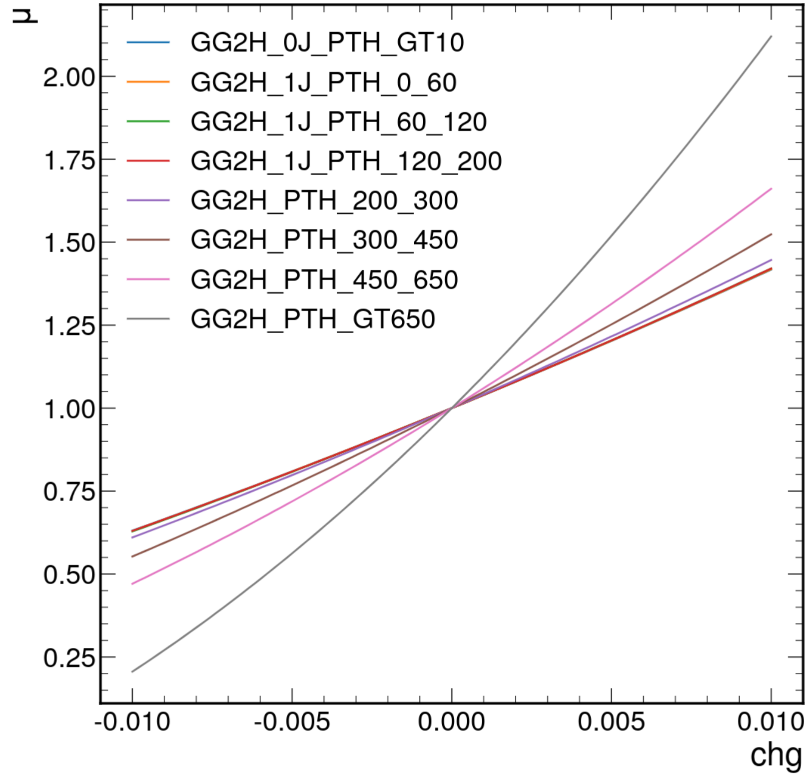
$c_i c_{HG}\ (i \neq HG)$ c^2_{HG}

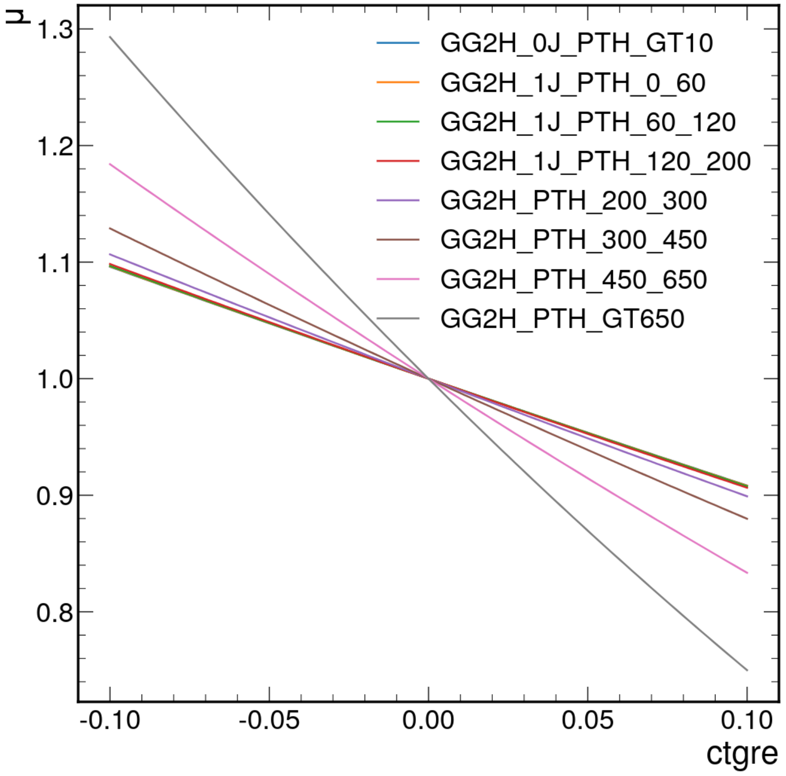
Remove unwanted contribution by removing all c^2_{HG} terms from `tree_loop_4`.

ggH results

Scaling equation for Stage 0 ggH:

$$\mu = 1 - 0.12c_{Hl}^{(3)} + 0.12c_{H\Box} + 0.06c'_{ll} - 0.03c_{HDD} + 40c_{HG} - 0.95Re(c_{tG}) - 0.12Re(c_{tH}) + O(c_i^2)$$

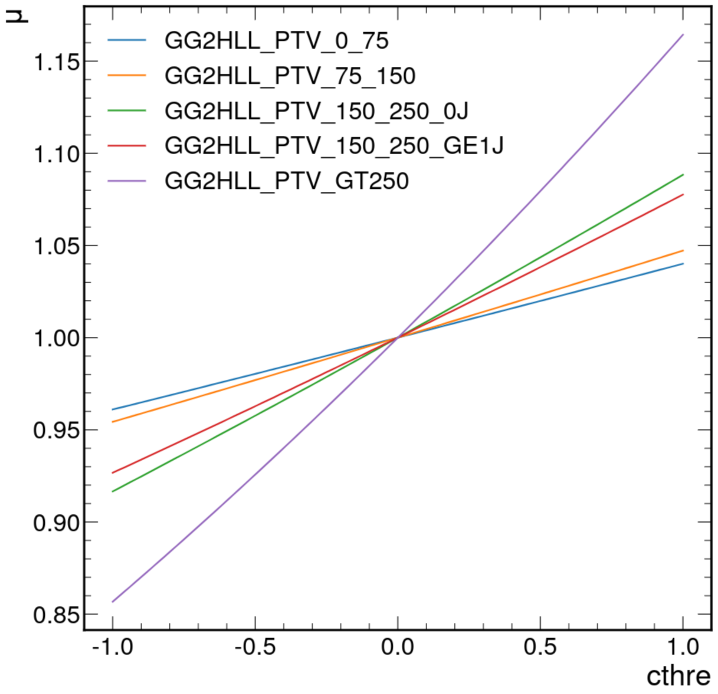


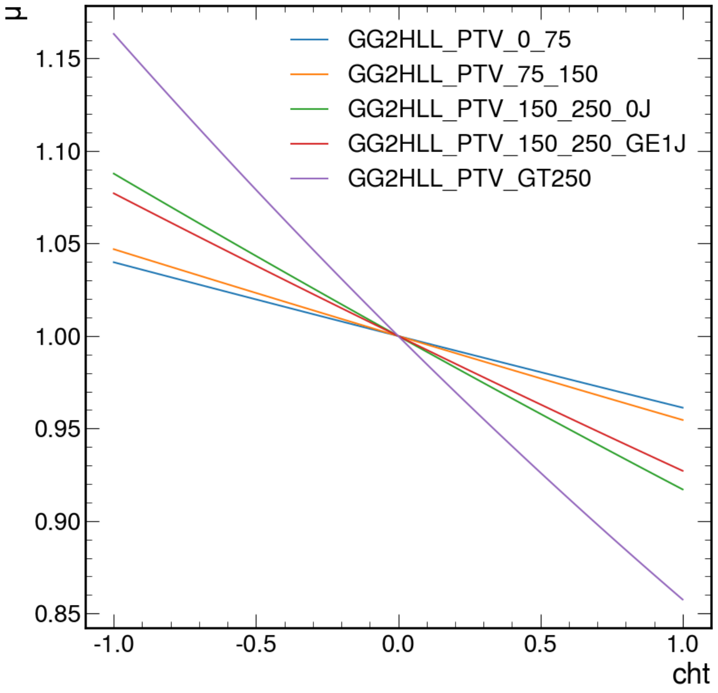


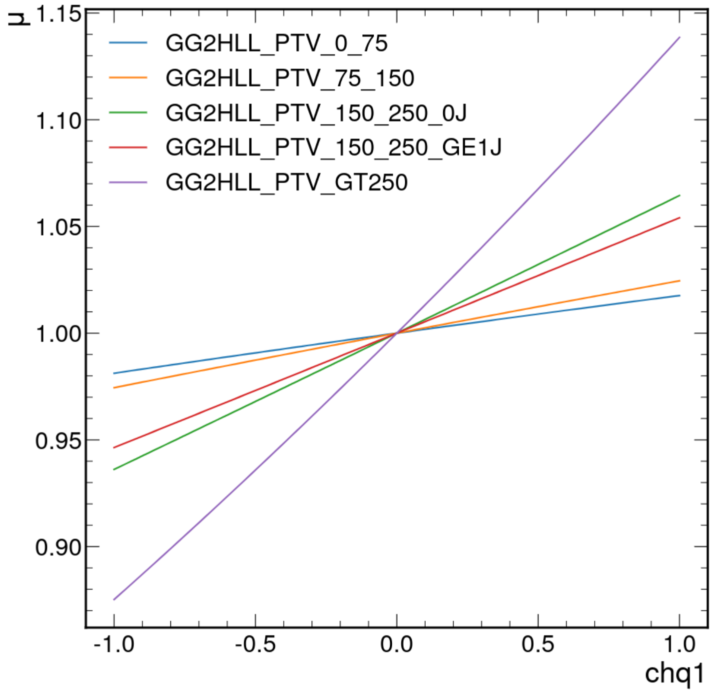
$ggZH$ results

Scaling equation for Stage 0 $ggZH$

$$\mu = 1 - 0.08c_{Hj}^{(3)} + 0.01c_{Hd} - 0.34c_{HG} - 0.31Re(c_{tG}) - 0.15c_{Hl}^{(3)} - 0.06c_{Ht} - 0.09c_{Hq}^{(3)} + 0.07Re(c_{tH}) + 0.03c_{HDD} - 0.02c_{Hu} + 0.12\, c'_{ll} + 0.04\, c_{Hq}^{(1)} - 0.01c_{Hj}^{(1)} + 0.12c_{H\Box} + 0.14c_{Hl}^{(1)} - 0.01c_{HWB} - 0.10c_{He} + O(c_i^2)$$



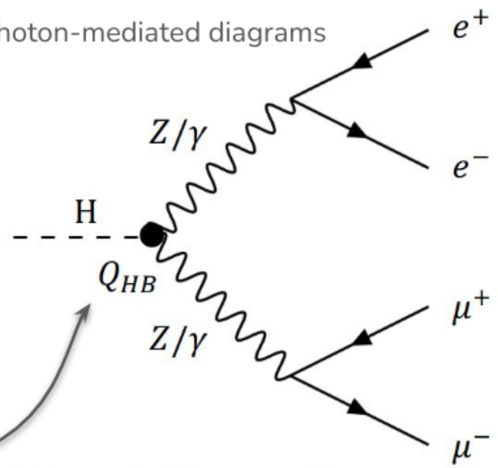




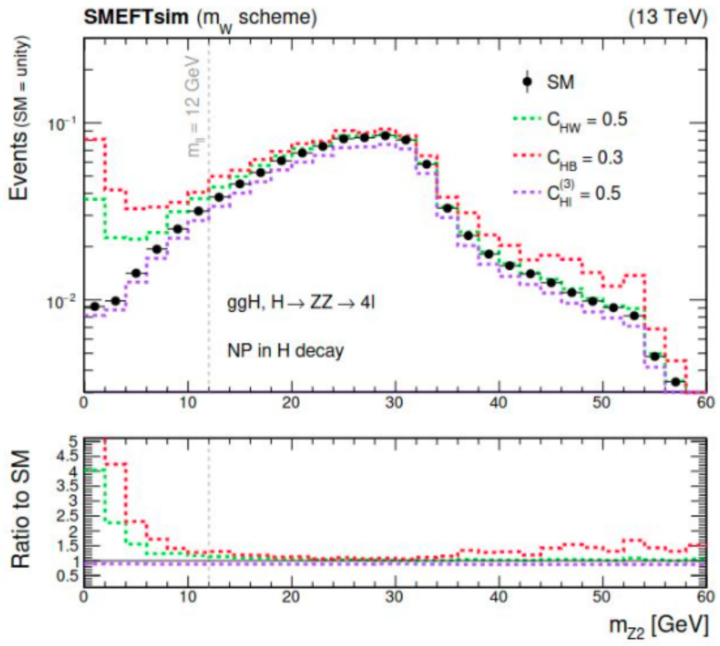
Acceptance corrections

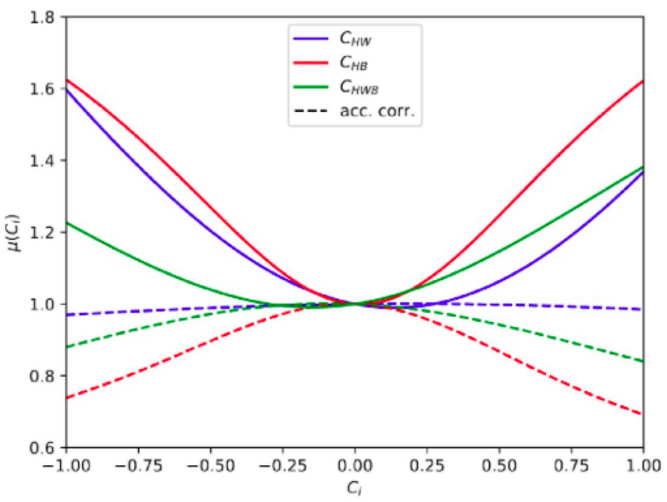
- EFT dependence in experimental phase space ≠ EFT dependence in inclusive phase space
  - EFT effects can depend on analysis acceptance/selection
  - Exacerbated by fact that STXS has **no fiducial selection on Higgs boson decay products**
- Problem for Higgs four-body decays e.g. H → ZZ* → 4l
  - Analysis places cut on invariant mass of subleading lepton pair: m_{Z2} > 12 GeV
  - Removes phase space with largest EFT effects → washes out the dependence in this channel

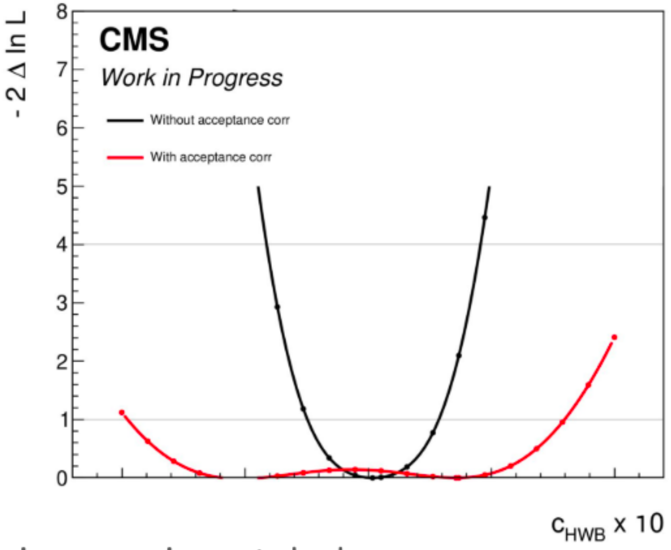
Pesky photon-mediated diagrams



Q_{HW}, Q_{HB}, Q_{HWB}







We add corrections to model EFT dependence in experimental phase space

Useful to introduce some fiducial-like selection in STXS definition?

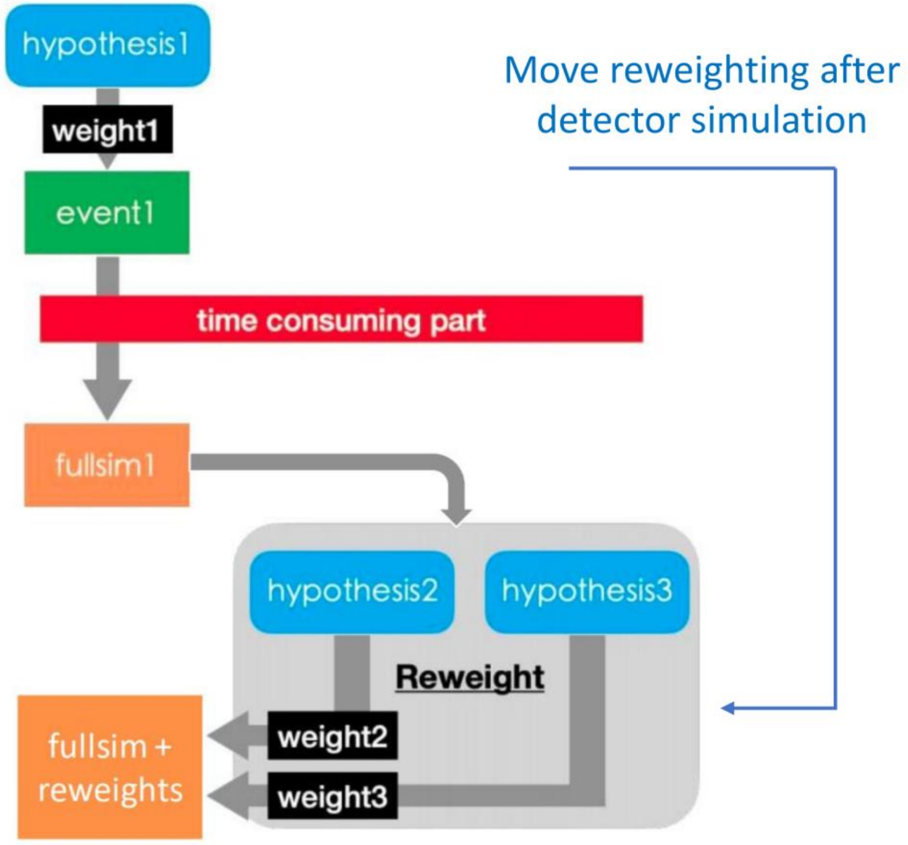
24

Standalone reweighting

- Developed nanoAOD-tools module to reweight existing SM nanoAOD events:
  - standalone reweighting module from MadGraph using EFT2Obs interface
  - use gen-level information to reweight event to any point in EFT parameter space

  → EFT samples with full reconstruction and detector effects

- Event ID skimming: input analyses provide list of MC event ID's landing in each analysis category
  - skim events in nanoAOD → apply standalone reweighting
  - obtain EFT effects only for events which enter analysis
  - extremely useful tool to study acceptance/selection effects and shape effects on fitted observables within experiment



See Jonathon's talk for subsequent studies (acceptance, selection and shape effects)

nanoAOD = root-based data tier in CMS

A common STXS parameterisation

Idea: create a SMEFT parameterization of the STXS which is public and free to use by CMS, ATLAS and theorists

Motivation: efficiency and accuracy/validity

- CMS, ATLAS and theorists derive their own SMEFT parameterisations
  - Repetition is good for validation sake but we should also try to reduce duplicated work
- Quite a bit of crosstalk between experiment and theory already, e.g. support for `SMEFT@NLO` and `SMEFTsim`
  - Theorists spend time telling both experiments how to do the same thing
- Encourages collaboration between experiment and theory → more accurate interpretations
  - From theory: newest models, analytical equations, checking input parameters, theoretical discussions such as linear vs quadratic order
  - From experiment: acceptance corrections, frameworks such as EFT2Obs (incl. matching & merging)
- Use opportunity to develop a common EFT parameterization format as well

Practical stuff

- Joint effort between LHC Higgs WG2 and LHC EFT WG with mixture of CMS, ATLAS and theory members

I. Brivio, K. Mimasu, C. Knight, J. Langford, E. Rossi, A. Cueto

I. Brivio, K. Mimasu	C. Knight, J. Langford	E. Rossi, A. Cueto
SMEFTsim and SMEFTatNLO authors	CMS	ATLAS

- Treat the CMS numbers as the nominal parameterisation to publish & compare with ATLAS
  - Validate different tools/approaches (generation/reweighting)
  - Many handles that can be different: process lines, input parameters, run settings (gen cuts), scales,…
  - Long arduous process… there are $\sim 17K\ A_i$ and B_{ij} terms in total
  - We are finally starting to converge! ☺

- Will publish the parameterisation with a note describing the tools and choices made
- Include instructions to run parameterisation with EFT2Obs

Common EFT parameterisation format (.json)

- Propose a standard json format for publishing EFT parameterisations
  - More "plug and play" when incorporating new parameterisations
  - To be used for analytical and MC-based derivations alike
- Eventually, we could imagine a library/database of such files to search within
- "metadata" field
  - Information on shape of observable (number/list/matrix) & coeffs on which it depends

```
"metadata": {
  "coefficients": [ "chb", "chbox", "chd", "chl3", "chw", "chwb", "cll1", "ctbre", "cthre", "ctwre", "cw" ],
  "observable_shape": "(1,)",
  "observable_names": [ "example" ],
```

example.json on indico

- "data" field:
  - contains monomial coefficients + errors ○ More than one error can be stored (MC, PDF, scale, …)

```
"data": {
  "central": {
    "SM": [ 10.0 ],
    "a_chb": [ 1.0 ],
    "a_chbox": [ 1.0 ],
```

Main prediction: A_i and B_{ij} coefficients

```
"u_MC": {
  "SM": [ 0.01 ],
  "a_chb": [ 0.01 ],
  "a_chbox": [ 0.01 ]
```

MC stat. uncertainty on A_i and B_{ij}

More metadata fields → reproducibility

```
{
  "metadata": {
    "coefficients": [ "chb", "chbox", "chd", "chl3", "chw", "chwb", "cll1", "ctbre", "cthre", "ctwre", "cw" ],
    "observable_shape": "(1,)",
    "observable_names": [ "example" ],
    "author": "Jane Bloggs",
    "contact": "j.bloggs@cern.ch",
    "date [DD/MM/YY]": "15/11/2023",
    "description": "An example SMEFT parametrisation json file for an observable such as a decay BR, represente
    monomials, respectively",
    "documentation": [ "https://mydocumentation.page.com", "https://arxiv.org/abs/2311.XXXXX" ],
    "tool_version": "MG5_aMC_v2_X_Y",
    "basis": "warsaw",
    "flavor_scheme": "topU3l",
    "inputs": {
      "Lambda": 1000,
      "MW": 91.1876,
      "GF": 1.16638e-05,
      "aS": 0.1181,
      "MH": 125.0,
      "MB": 3.237,
      "MT": 173.2
    },
    "EW_input_scheme": "MW_MZ_GF",
    "EFT_order": "quadratic",
    "scale_choice": 125.09,
    "pertubative_order_QCD": "LO",
    "pertubative_order_QED/EW": "LO",
    "method": "reweighting"
  },
```

Please let us know if you have ideas/feedback on the current format!

- Basis choice
- EW input schemes,
- Flavour assumptions
- Links to documentation
- Tools, versions,...

- MC settings (scale choice,...)
- Orders (EFT, perturbative,...)
- Method used
- Free-form fields

Preserving the code / EFT2Obs instructions

- At the very least, we will provide instructions to reproduce the parameterisation with EFT2Obs
- Preferably, we will integrate EFT2Obs into a workflow tool
  - Reduce the number of buttons you need to push to $O(1)$

Preserving the code / EFT2Obs instructions

- At the very least, we will provide instructions to reproduce the parameterisation with EFT2Obs
- Preferably, we will integrate EFT2Obs into a workflow tool
  - Reduce the number of buttons you need to push to $O(1)$
- This was a topic of the hackathon earlier this week!

snakemake

  - EFT2Obs provided in a docker container → usable anywhere
  - Created a small workflow (WH_{lep} only) from beginning to end
  - Still a work in progress… but you can give it a go already

EFT2Obs-Workflow

[GitHub]

To create the snakemake environment:

```
curl -Ls https://micro.mamba.pm/api/micromamba/linux-64/latest | tar -xvj bin/micromamba
mv bin snakemake_env
export MAMBA_ROOT_PREFIX=${PWD}/snakemake_env
eval "$(./snakemake_env/micromamba shell hook -s posix)"

micromamba create -c conda-forge -c bioconda -n snakemake snakemake
micromamba activate snakemake
```

To source the environment from a fresh terminal later do:

```
export MAMBA_ROOT_PREFIX=${PWD}/snakemake_env
eval "$(./snakemake_env/micromamba shell hook -s posix)"
micromamba activate snakemake
```

I assume that you have apptainer installed on your system. For the time being, we need to increase the size of the temporary overlay (so you can write inside the container):

```
sudo apptainer config global --set "sessiondir max size" "1024"
```

More details about overlays: https://apptainer.org/docs/user/main/persistent_overlays.html

One could now run the snakemake command and it will internally pull the docker (converted by apptainer) container. However, I recommend pulling it with apptainer first

```
apptainer pull charlotteknight/eft2obs
```

To run the workflow first source some environment variables with `source env_vars.sh`. Then run:

```
snakemake --cores 1 --sdm apptainer --apptainer-args "--writable-tmpfs " -p results/WH_lep_SMEI
```

Thanks to Joseph Mariano for joining me

Running this command will do everything! →

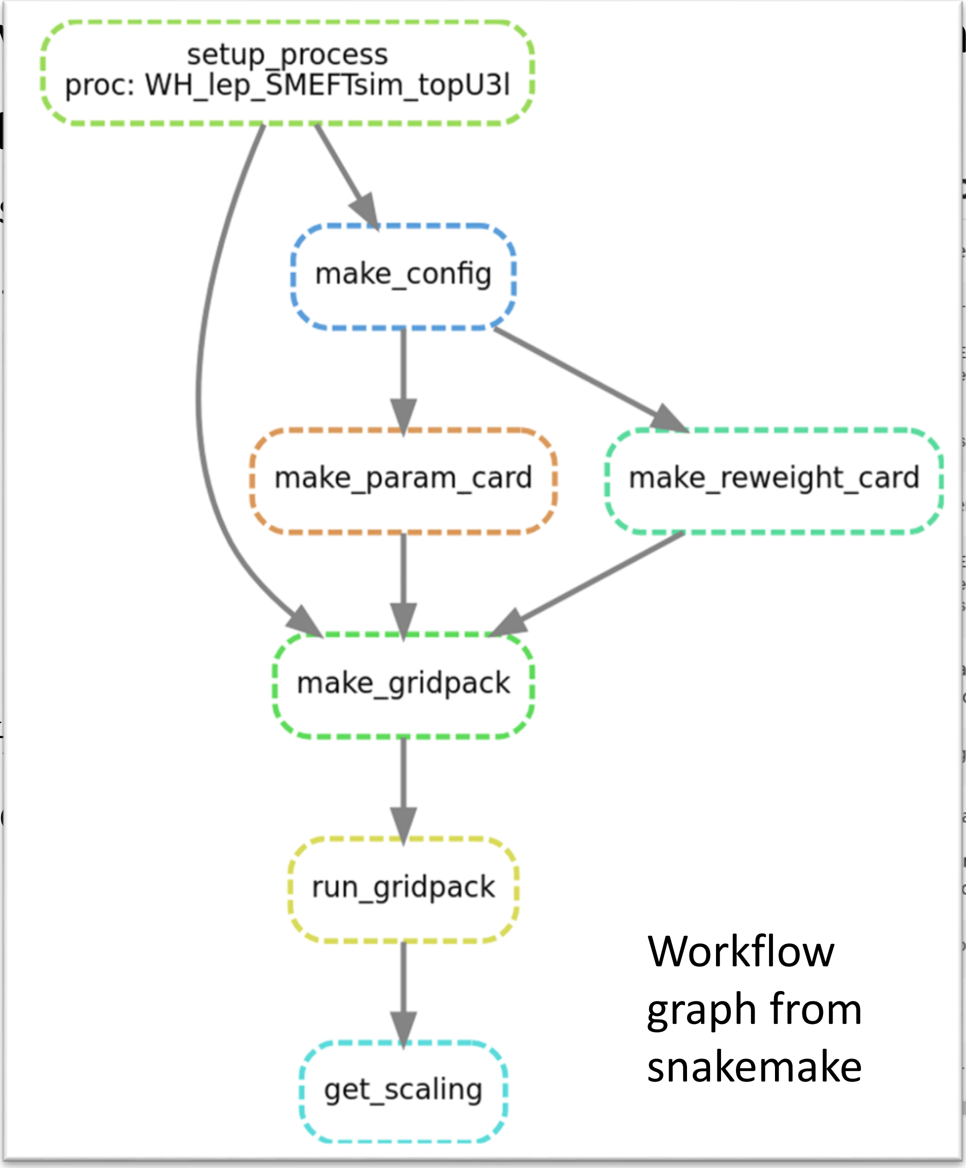
Preserving the code / EFT2Obs instructions

- At the very least, we will prov ... eterisation with EFT2Obs
- Preferably, we will integrate I
  - Reduce the number of button
- This was a topic of the hacka

snake

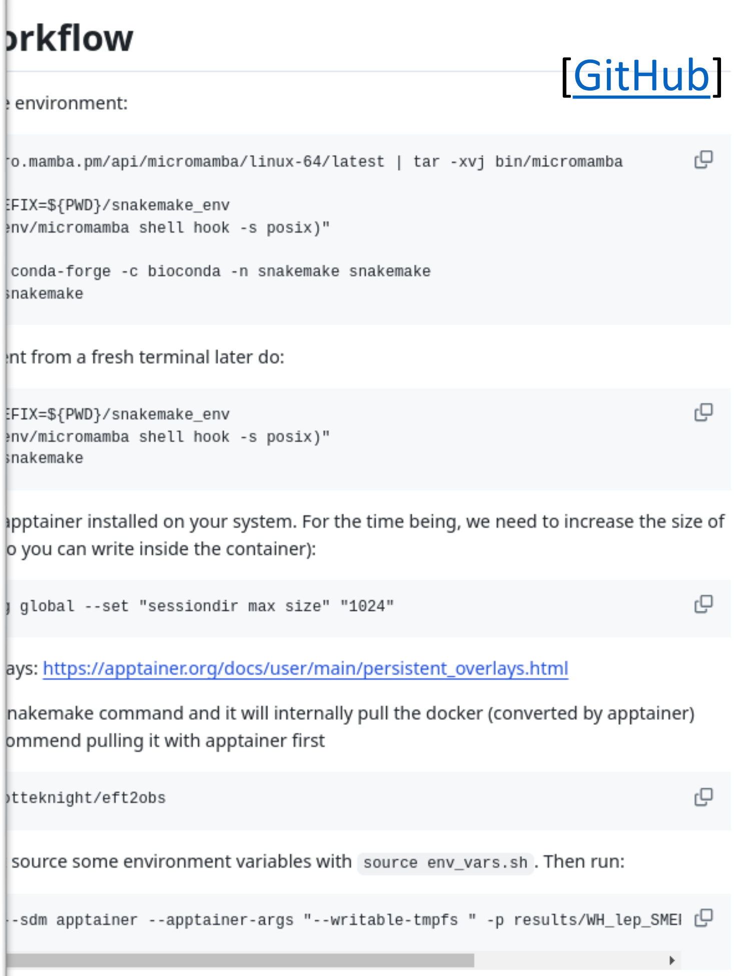
  - EFT2Obs provided in a docker
  - Created a small workflow (WH
  - Still a work in progress… but y

Thanks to Joseph Mariano for joining me



Workflow graph from snakemake

[GitHub]



Summary

- Deriving SMEFT interpretation of STXS to quadratic order and including all CP-even and CP-odd operators
- Propagator corrections for SMEFTsim processes included
- Total width derived to quadratic order and validated against analytical result
- SMEFT@NLO now used for $gg\mathrm{H}$ and $gg\mathrm{ZH}$
- Post-generation reweighting tools used to study acceptance, selection, and shape effects

- Publishing a common STXS parameterization collaborating with CMS, ATLAS and theorists
  - Outputs: the parameterization, an accompanying note, instructions to reproduce
- Propose a common parameterization format (json)
  - Publish STXS parameterization in this format
  - Encourage others to follow suit → start collecting a library of predictions

Propagator corrections in quadratic parameterisations

- Corrections to width of propagators $A \sim \frac{1}{\Gamma+\delta\Gamma} = \frac{1}{\Gamma}\left(1 - \frac{\delta\Gamma}{\Gamma} + \underbrace{\frac{(\delta\Gamma)^2}{\Gamma^2} + \cdots}\right)$ $\qquad \delta\Gamma = \delta\Gamma(C_i)$

  Higher-order terms neglected in `SMEFTsim`

- Square the amplitude: $A^2 = \left[\frac{1}{\Gamma}\left(1 - \frac{\delta\Gamma}{\Gamma}\right)\right]^2 \frac{1}{\Gamma^2}\left(1 - \underbrace{\frac{2\delta\Gamma}{\Gamma}}_{A_i} + \underbrace{\frac{(\delta\Gamma)^2}{\Gamma^2}}_{B_{ij}}\right)$

- But what about $A^2 = \left[\frac{1}{\Gamma}\left(1 - \frac{\delta\Gamma}{\Gamma} + \frac{(\delta\Gamma)^2}{\Gamma^2}\right)\right]^2 = \frac{1}{\Gamma^2}\left(1 - \frac{2\delta\Gamma}{\Gamma} + \frac{3(\delta\Gamma)^2}{\Gamma^2} + \cdots\right)$
- In `SMEFTsim`, $\delta\Gamma(\mathrm{C_i})$ is also only given to linear order in C_i
- Very unclear what we can do about propagator effects at quadratic level
- With current tools, no way to calculate these terms correctly