Lecture*: MATRIX hands-on

Luca Buonocore

Advanced School & Workshop on Multiloop Scattering Amplitudes

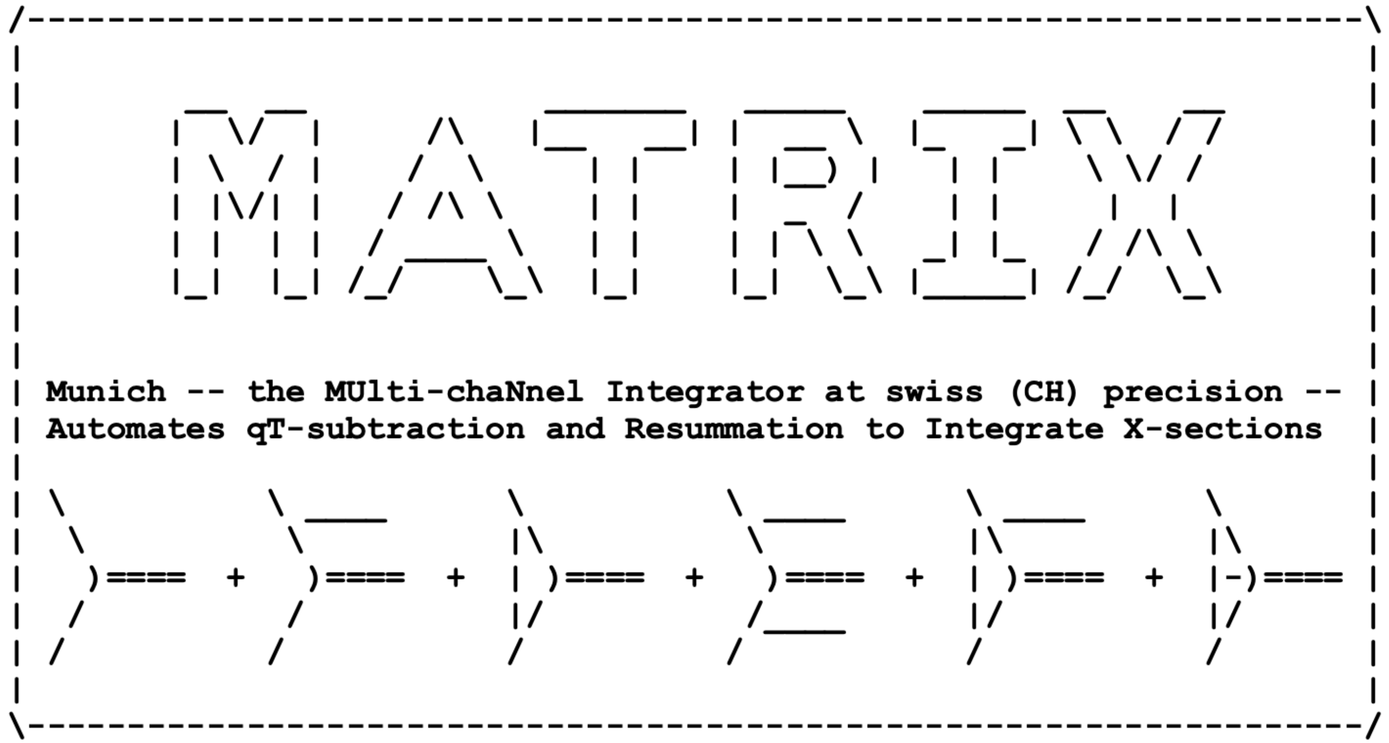
NISER - 15-19 January 2024

Hands-on: MATRIX

Useful links:

https://matrix.hepforge.org/

https://matrix.hepforge.org/manual.html



Hands-on: MATRIX

Getting started

Prerequisites: working installation of LHAPDF (https://lhapdf.hepforge.org/downloads/) such that the program **lhapdf-config** is available to your system

Download the latest stable release MATRIX_vX.Y.Z.tar.gz at https://matrix.hepforge.org/download.html

After extracting the archive MATRIX_vX.Y.Z.tar.gz (for example, tar -xzf MATRIX_vX.Y.Z.tar.gz), the program **matrix** is available in the MATRIX folder MATRIX_vX.Y.Z (from now on $MATRIX_HOME)

./matrix

user-friendly shell with tab completion
(python3 compatible)

Core part in C++

Hands-on: MATRIX

Getting started

Edit the configuration file $MATRIX_HOME/config/MATRIX_configuration

For most of the users, default choices are fine, but

choose your favourite editor!

select the running mode

```
##############################
# MATRIX configuration file #
##############################
# This file contains all parameters to configure MATRIX
# In the run_folders this is the link to the central configuration file
# in MATRIX/config/; you can replace the link by a copy to have individual
# configurations for the different processes

# Editor to be used to edit input files from MATRIX shell
# (default: use the one specified under environmental variable EDITOR)
default_editor = emacs  # eg, emacs, vi, nano, ...

# runmode of MATRIX: 0 -- multicore (default)
#                    1 -- cluster
mode = 1
```

Hands-on: MATRIX

Getting started

Edit the configuration file $MATRIX_HOME/config/MATRIX_configuration

For most of the users, default choices are fine, but

in Cluster Mode, other important parameters to specify

```
###==================###
## cluster parameter ##
###==================###
# Name of cluster currently supported:
#   slurm, LSF (eg, lxplus), HTcondor, condor_lxplus (special version working on lxplus HTCondor),
#   condor, PBS, Torque, SGE (PSB and Torque/OpenPBS are identical at the moment)
cluster_name = slurm

# Queue/Partition of cluster to be used for running
cluster_queue = grazzini

# Use local scratch directory to run on cluster (speedup for slow shared file systems):
#   condor, PBS, Torque, SGE (PSB and Torque/OpenPBS are identical at the moment)
cluster_name = slurm

# Queue/Partition of cluster to be used for running
cluster_queue = grazzini

# Use local scratch directory to run on cluster (speedup for slow shared file systems):
# 0 -- standard run on shared file system (default)
# 1 -- run in local scratch of nodes; PROVIDE cluster_local_scratch_path BELOW!
# NOT IMPLEMENTED YET: 2 -- run without shared file system; PROVIDE cluster_local_scratch_path BELOW!
cluster_local_run = 0
```

just an example!

Hands-on: MATRIX

Getting started

if you have your own installation of some of the required tools, or standard library installed in non-standard place ...

```
###======================###
## compilation parameter ##
###======================###

# maximum number of cores used for compilation (default: maximal cores available on the machine);
#nr_cores = 16 # when commented the default is used

# you can specify the path to lhapdf-config executable;  not required if lhapdf-config executable
# accessible from command line (will be determined automatically in that case)
#path_to_lhapdf = /PATH/lhapdf-config  # !absolute path!

# if OpenLoops is already installed, you can specify the path to openloops executable; not required
# if openloops executable accessible from command line (will be determined automatically in that case);
# otherwise, OpenLoops will be downloaded and installed automatically
#path_to_openloops = /PATH/openloops  # !absolute path!

#you can specify the path to recola, if already installed locally.
#path_to_recola = /PATH/recola_sm # !absolute path!

#you can specify the path to chaplin, if already installed locally.
#path_to_chaplin = /PATH/chaplin # !absolute path!

# you can specify the path to ginac, if already installed locally; ginac will not be compiled in this case
#path_to_ginac = /PATH/ginac-install/  # !absolute path!

# you can specify the path to cln, if already installed locally; cln will not be compiled in this case
#path_to_cln = /PATH/cln-install/  # !absolute path!

# you can specify the path to 2loop amplitude of ppaaa03 process, if already installed locally;
#path_to_ppaaa03_2loop = /PATH/ppaaa03_2loop-install/  # !absolute path!

# you can specify the path to the libfortran libary, usually found by the system automatically
# NOTE: this path must also be set if the libquadmath library is not found
# NOTE: this path can be also used if other libaries are missing during the compilation process
#path_to_libgfortran = /PATH/x86_64-linux-gnu/  # !absolute path!

# you can specify the path to gsl-config executable;  not required if gsl-config executable
# accessible from command line (will be determined automatically in that case)
#path_to_gsl = /PATH/gsl-config  # !absolute path!
```

Hands-on: MATRIX

Set up a process (first time)

```
$ ./matrix
```

List of available process

```
$>> list
```

Hands-on: MATRIX

Set up a process (first time)

process_id		process		description
pph21	>>	p p --> H	>>	on-shell Higgs production (NNLO)
ppz01	>>	p p --> Z	>>	on-shell Z production (NNLO,NLO EW)
ppw01	>>	p p --> W^-	>>	on-shell W- production with CKM (NNLO)
ppwx01	>>	p p --> W^+	>>	on-shell W+ production with CKM (NNLO)
ppeex02	>>	p p --> e^- e^+	>>	Z production with decay (NNLO,NLO EW)
ppnenex02	>>	p p --> v_e^- v_e^+	>>	Z production with decay (NNLO,NLO EW)
ppenex02	>>	p p --> e^- v_e^+	>>	W- production with decay and CKM (NNLO,NLO EW)
ppexne02	>>	p p --> e^+ v_e^-	>>	W+ production with decay and CKM (NNLO,NLO EW)
ppaa02	>>	p p --> gamma gamma	>>	gamma gamma production (NNLO)
ppeexa03	>>	p p --> e^- e^+ gamma	>>	Z gamma production with decay (NNLO)
ppnenexa03	>>	p p --> v_e^- v_e^+ gamma	>>	Z gamma production with decay (NNLO)
ppenexa03	>>	p p --> e^- v_e^+ gamma	>>	W- gamma production with decay (NNLO)
ppexnea03	>>	p p --> e^+ v_e^- gamma	>>	W+ gamma production with decay (NNLO)
ppzz02	>>	p p --> Z Z	>>	on-shell ZZ production (NNLO)
ppwxw02	>>	p p --> W^+ W^-	>>	on-shell WW production (NNLO)
ppemexmx04	>>	p p --> e^- mu^- e^+ mu^+	>>	ZZ production with decay (NNLO,NLO gg,NLO EW)
ppeeexex04	>>	p p --> e^- e^- e^+ e^+	>>	ZZ production with decay (NNLO,NLO gg,NLO EW)
ppeexnmnmx04	>>	p p --> e^- e^+ v_mu^- v_mu^+	>>	ZZ production with decay (NNLO,NLO gg,NLO EW)
ppemxnmnex04	>>	p p --> e^- mu^+ v_mu^- v_e^+	>>	WW production with decay (NNLO,NLO gg,NLO EW)
ppeexnenex04	>>	p p --> e^- e^+ v_e^- v_e^+	>>	ZZ/WW production with decay (NNLO,NLO gg,NLO EW)
ppemexnmx04	>>	p p --> e^- mu^- e^+ v_mu^+	>>	W-Z production with decay (NNLO,NLO EW)
ppeeexnex04	>>	p p --> e^- e^- e^+ v_e^+	>>	W-Z production with decay (NNLO,NLO EW)
ppeexmxnm04	>>	p p --> e^- e^+ mu^+ v_mu^-	>>	W+Z production with decay (NNLO,NLO EW)
ppeexexne04	>>	p p --> e^- e^+ e^+ v_e^-	>>	W+Z production with decay (NNLO,NLO EW)
ppttx20	>>	p p --> top anti-top	>>	on-shell top-pair production (NNLO)
ppaaa03	>>	p p --> gamma gamma gamma	>>	gamma gamma gamma production (NNLO)

new in MATRIX 2.1.0

Hands-on: MATRIX

Set up a process (first time)

our favourite example: NC DY

```
------------------------------------------------------------------------------------------------------
process_id     ||   process                                ||   description
------------------------------------------------------------------------------------------------------
pph21          >>   p p --> H                              >>   on-shell Higgs production (NNLO)
ppz01          >>   p p --> Z                              >>   on-shell Z production (NNLO,NLO EW)
ppw01          >>   p p --> W^-                            >>   on-shell W- production with CKM (NNLO)
ppwx01         >>   p p --> W^+                            >>   on-shell W+ production with CKM (NNLO)
ppeex02        >>   p p --> e^- e^+                        >>   Z production with decay (NNLO,NLO EW)
ppnenex02      >>   p p --> v_e^- v_e^+                    >>   Z production with decay (NNLO,NLO EW)
ppenex02       >>   p p --> e^- v_e^+                      >>   W- production with decay and CKM (NNLO,NLO EW)
ppexne02       >>   p p --> e^+ v_e^-                      >>   W+ production with decay and CKM (NNLO,NLO EW)
ppaa02         >>   p p --> gamma gamma                    >>   gamma gamma production (NNLO)
ppeexa03       >>   p p --> e^- e^+ gamma                  >>   Z gamma production with decay (NNLO)
ppnenexa03     >>   p p --> v_e^- v_e^+ gamma              >>   Z gamma production with decay (NNLO)
ppenexa03      >>   p p --> e^- v_e^+ gamma                >>   W- gamma production with decay (NNLO)
ppexnea03      >>   p p --> e^+ v_e^- gamma                >>   W+ gamma production with decay (NNLO)
ppzz02         >>   p p --> Z Z                            >>   on-shell ZZ production (NNLO)
ppwxw02        >>   p p --> W^+ W^-                        >>   on-shell WW production (NNLO)
ppemexmx04     >>   p p --> e^- mu^- e^+ mu^+              >>   ZZ production with decay (NNLO,NLO gg,NLO EW)
ppeeexex04     >>   p p --> e^- e^- e^+ e^+                >>   ZZ production with decay (NNLO,NLO gg,NLO EW)
ppeexnmnmx04   >>   p p --> e^- e^+ v_mu^- v_mu^+          >>   ZZ production with decay (NNLO,NLO gg,NLO EW)
ppemxnmnex04   >>   p p --> e^- mu^+ v_mu^- v_e^+          >>   WW production with decay (NNLO,NLO gg,NLO EW)
ppeexnenex04   >>   p p --> e^- e^+ v_e^- v_e^+            >>   ZZ/WW production with decay (NNLO,NLO gg,NLO EW)
ppemexnmx04    >>   p p --> e^- mu^- e^+ v_mu^+            >>   W-Z production with decay (NNLO,NLO EW)
ppeeexnex04    >>   p p --> e^- e^- e^+ v_e^+              >>   W-Z production with decay (NNLO,NLO EW)
ppeexmxnm04    >>   p p --> e^- e^+ mu^+ v_mu^-            >>   W+Z production with decay (NNLO,NLO EW)
ppeexexne04    >>   p p --> e^- e^+ e^+ v_e^-              >>   W+Z production with decay (NNLO,NLO EW)
ppttx20        >>   p p --> top anti-top                   >>   on-shell top-pair production (NNLO)
ppaaa03        >>   p p --> gamma gamma gamma              >>   gamma gamma gamma production (NNLO)
```

Hands-on: MATRIX

Set up a process (first time)

```
$>> ppeex02
```

Agree with the terms to use Matrix

```
$>>y
```

```
$>>y
```

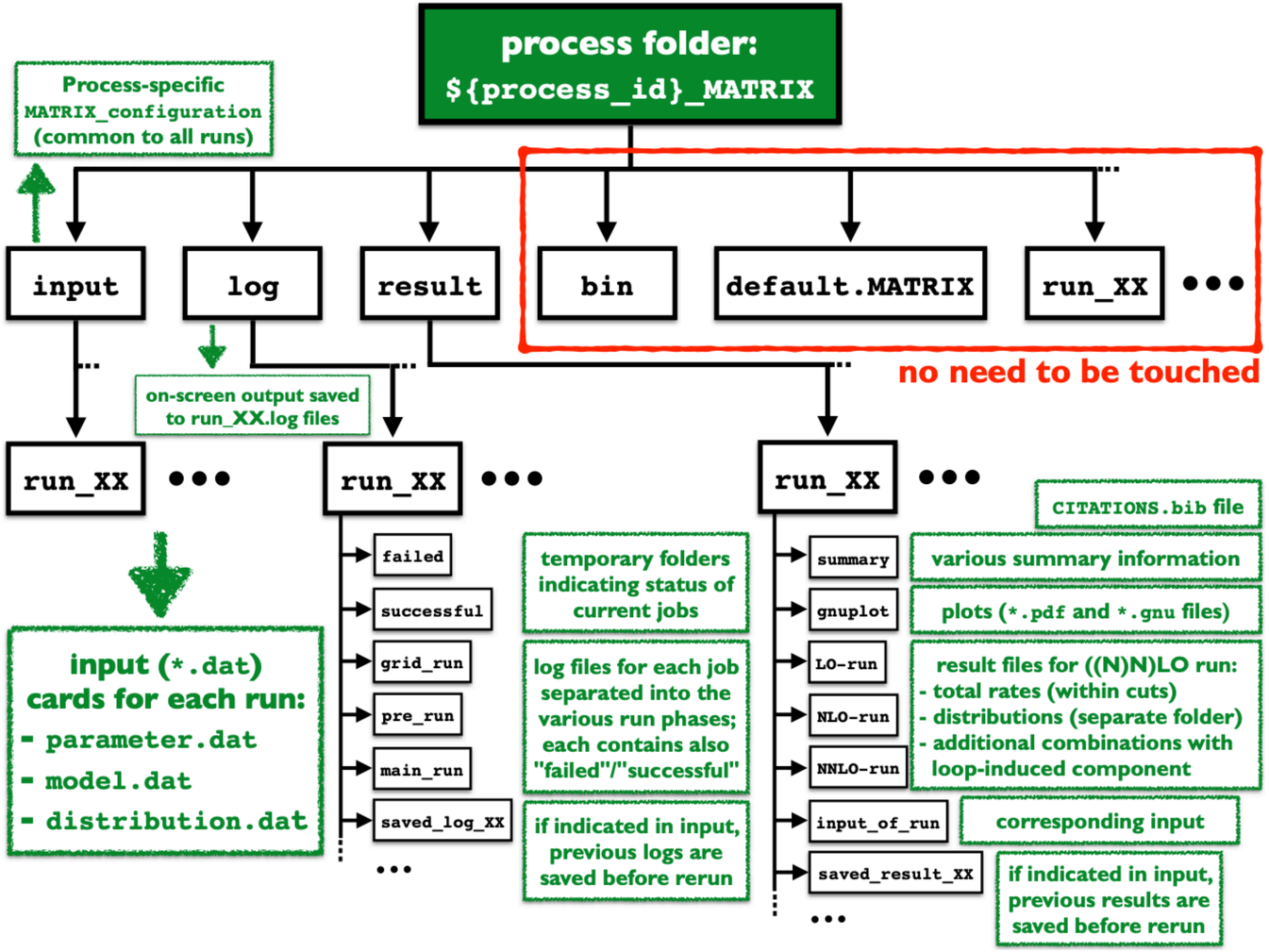
This will

- install all necessary dependences (OpenLoops2, CLN, GINAC)
- download and compile relevant tree-level and one-loop amplitudes
- compile the actual MATRIX process
- generate the main run folder

```
<<MATRIX-INFO>> Process folder successfully created.
<<MATRIX-INFO>> Process generation finished, to go to the run directory type:
                cd /disk/data11/ttp/lbuono/codes/MATRIX_v2.1.0/run/ppeex02_MATRIX
<<MATRIX-INFO>> and start run by typing:
                ./bin/run_process
```

Hands-on: MATRIX

Structure of run directory



Hands-on: MATRIX

Run the process

In the run directory

```
$ ./bin/run_process
```

Interactive mode

1. choose a name of the run (run_XXX)
2. set up main parameters of the run, parameters of the model, (single and, possibly, double) distributions
3. run all stages or run particular stage

Modification of the parameters is performed through the selected default editor

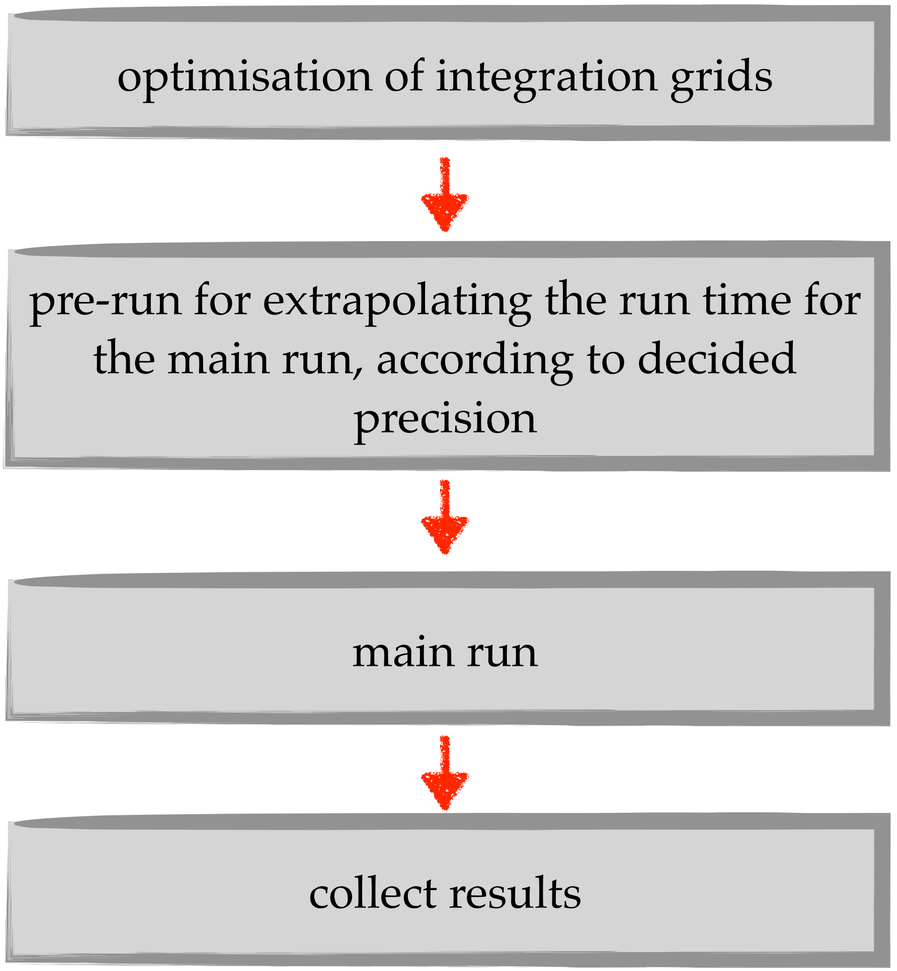
Alternatively

1. choose a name of the run (run_XXX)
2. set up run folder
3. changes parameters “offline”
4. rerun ./bin/run_process with the same run name, run all stages or run particular stage

Advice: in cluster mode, use a **terminal multiplexer** (as `screen` or `tmux`)

Hands-on: MATRIX

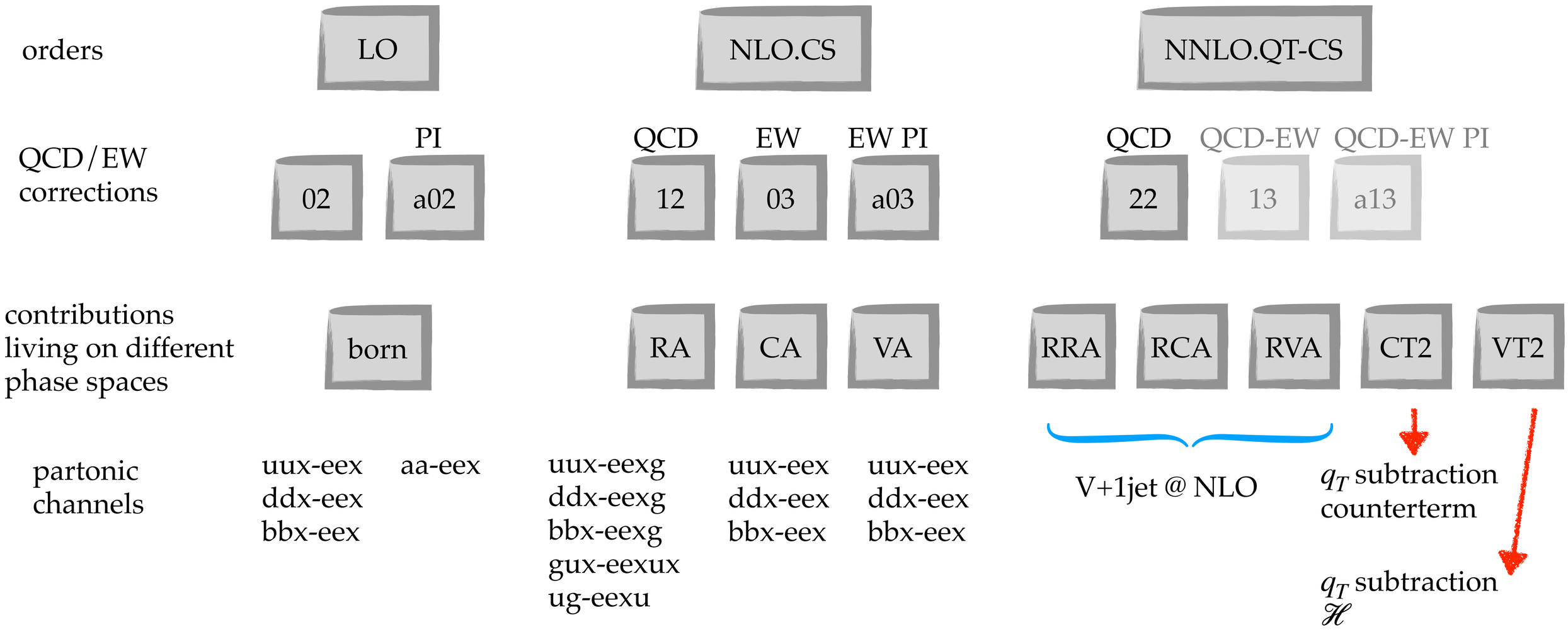
Run stages



automatically determines the **required number of integration points** for each contribution and prepare all the folders to start their parallel runs

Hands-on: MATRIX

Layers of parallelisation for DY example (no MPI)



Hands-on: MATRIX

Systematics due to the use of q_T-subtraction (see Sec7. of 1711.06631)

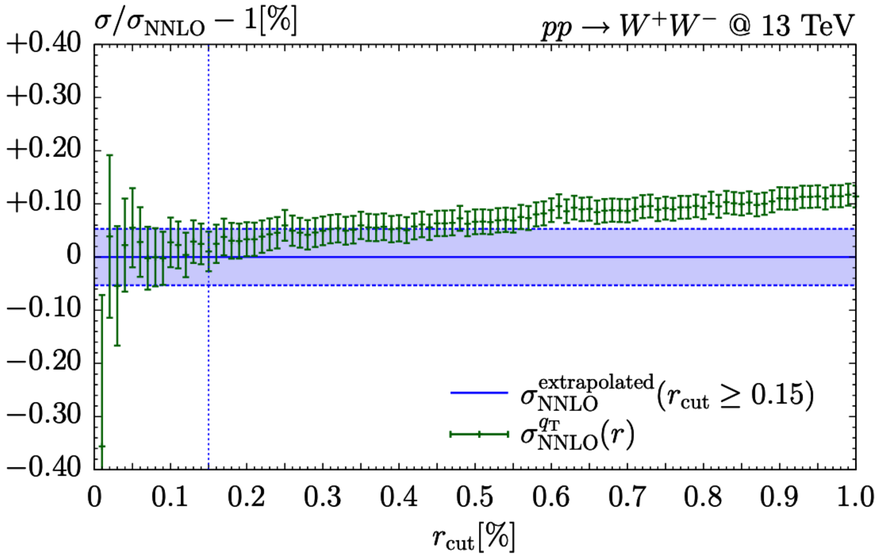
Only the NNLO corrections is computed using the q_T subtraction method

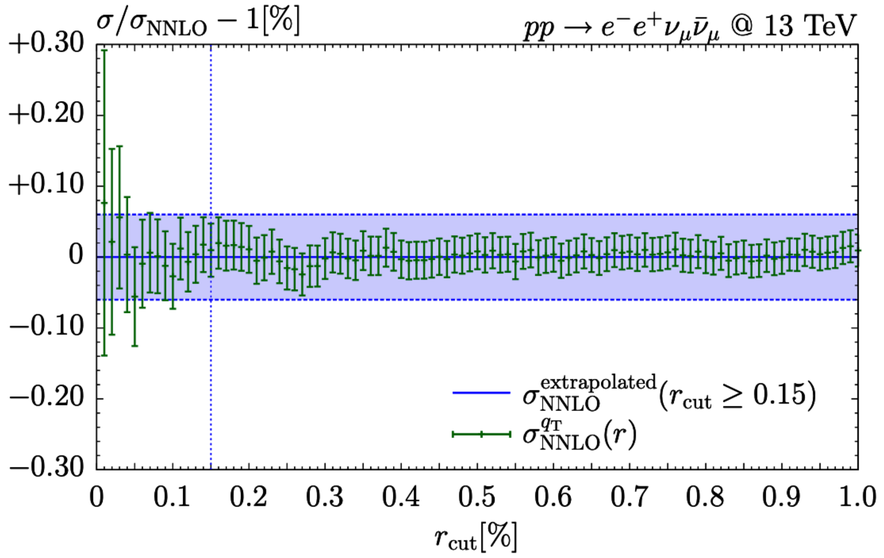
For a process of the kind $p + p \to F + X$, a minimun cut on the transverse momentum of the triggered final state F divided by its mass is imposed $q_T/M_F > r_{\text{cut}}^{\text{min}}$ (<1%)

The cumulative distribution in the range $[r_{\text{cut}}^{\text{min}}, 1\%]$ is accumulated **during the same run** wist steps of 0.01%

From these data, an **extrapolation procedure** is used to estimate the limits $r_{\text{cut}} \to 0$ and a relative uncertainty

The extrapolation procedure is based on a quadratic χ^2 fit and can also be applied to differential distributions on bin-by-bin basis





Hands-on: MATRIX

Systematics due to the use of q_T-subtraction (see Sec7. of 1711.06631)

Only the NNLO corrections is computed using the q_T subtraction method

For a process of the kind $p + p \to F + X$, a minimun cut on the transverse momentum of the triggered final state F divided by its mass is imposed $q_T/M_F > r_{\text{cut}}^{\text{min}}$ (<1%)

Large power corrections for specific cases as processes involving photons, due to photon isolation requirements (Frixione isolation)

In these cases, it is better to use a smaller value for r_{cut} (in input card `parameter`, this is set by `switch_qT_accuracy`)