How fast does the WAGO?

Yonsei-Konkuk-Sogang Mini-workshop

Jorinde van de Vis 10/02/2025

Configuration settings and convergence

Loading the configuration settings

- Defaults in src/WallGo/config.py
- Read in by the `manager`
- Individual configuration settings can be overwritten by e.g.
  `manager.config.configGrid.spatialGridSize = 20`
- You can also read in a custom configuration file, e.g.

```
639     def configureManager(self, inOutManager: "WallGo.WallGoManager") -> None:
640         """We load the configs from a file for this example."""
641         inOutManager.config.loadConfigFromFile(
642             pathlib.Path(self.exampleBaseDirectory / "singletStandardModelZ2Config.ini")
643         )
644         super().configureManager(inOutManager)
```

Configuration settings (these come from the default file)

Grid

```python
@dataclass
class ConfigGrid:
    """ Holds the config of the Grid3Scales class. """

    spatialGridSize: int = 40
    """ Number of grid points in the spatial direction (M in 2204.13120). """

    momentumGridSize: int = 11
    """
    Number of grid points in the momentum directions (N in 2204.13120).
    MUST BE ODD.
    """

    ratioPointsWall: float = 0.5
    """
    Fraction of points inside the wall defined by the interval
    [-wallThickness+wallCenter, wallThickness+wallCenter]. Should be a number between 0
    and 1.
    """

    smoothing: float = 0.1
    """ Smoothing factor of the mapping function (the larger the smoother). """
```

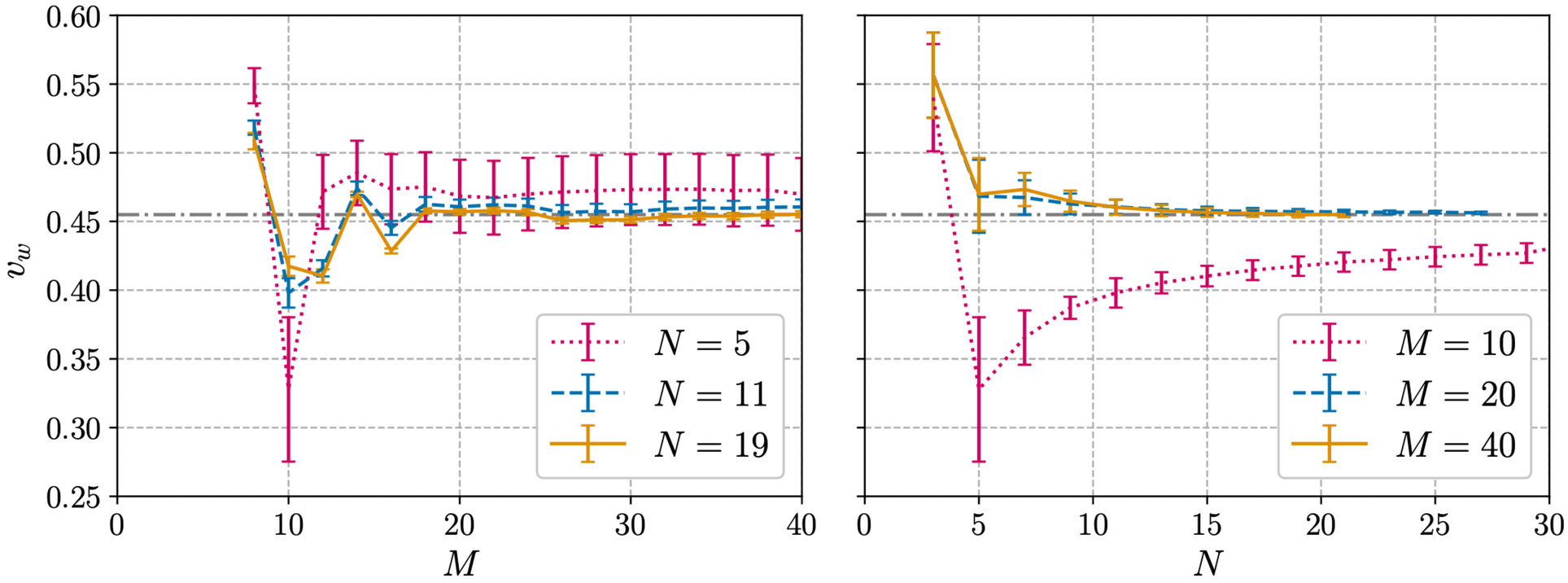
Configuration settings (these come from the default file)

Grid

```python
@dataclass
class ConfigGrid:
    """ Holds the config of the Grid3Scales class. """

    spatialGridSize: int = 40
    """ Number of grid points in the spatial direction (M in 2204.13120). """

    momentumGridSize: int = 11
    """
    Number of grid points in the momentum directions (N in 2204.13120).
    MUST BE ODD.
    """

    ratioPointsWall: float = 0.5
    """
    Fraction of points inside the wall defined by the interval
    [-wallThickness+wallCenter, wallThickness+wallCenter]. Should be a number between 0
    and 1.
    """

    smoothing: float = 0.1
    """ Smoothing factor of the mapping function (the larger the smoother). """
```

Number of Chebyshev polynomials. More is better, but comes at a higher computational cost. Additional momentumGridSize points are more “expansive” than spatialGridSize.

Convergence in number of polynomials



Configuration settings (these come from the default file)

Grid

```python
@dataclass
class ConfigGrid:
    """ Holds the config of the Grid3Scales class. """

    spatialGridSize: int = 40
    """ Number of grid points in the spatial direction (M in 2204.13120). """

    momentumGridSize: int = 11
    """
    Number of grid points in the momentum directions (N in 2204.13120).
    MUST BE ODD.
    """

    ratioPointsWall: float = 0.5
    """
    Fraction of points inside the wall defined by the interval
    [-wallThickness+wallCenter, wallThickness+wallCenter]. Should be a number between 0
    and 1.
    """

    smoothing: float = 0.1
    """ Smoothing factor of the mapping function (the larger the smoother). """
```

Dependence on Grid configuration settings

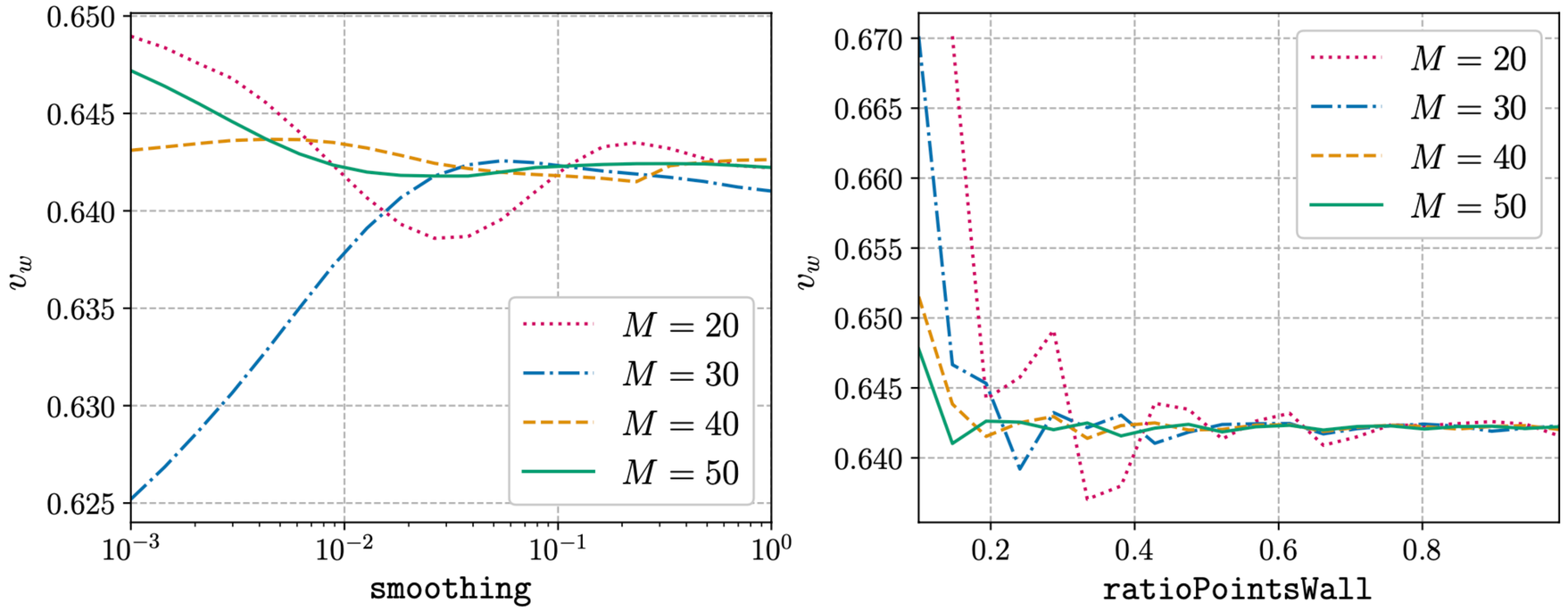


Figure 5: Dependence of v_w on different values of `smoothing` (left) and `ratioPointsWall` (right) in the xSM. We used `ratioPointsWall` $= 0.5$ in the left plot, `smoothing` $= 0.1$ in the right plot, and `meanFreePathScale` $= 50/T_\mathrm{n}$ for both.

Dependence on Grid configuration settings

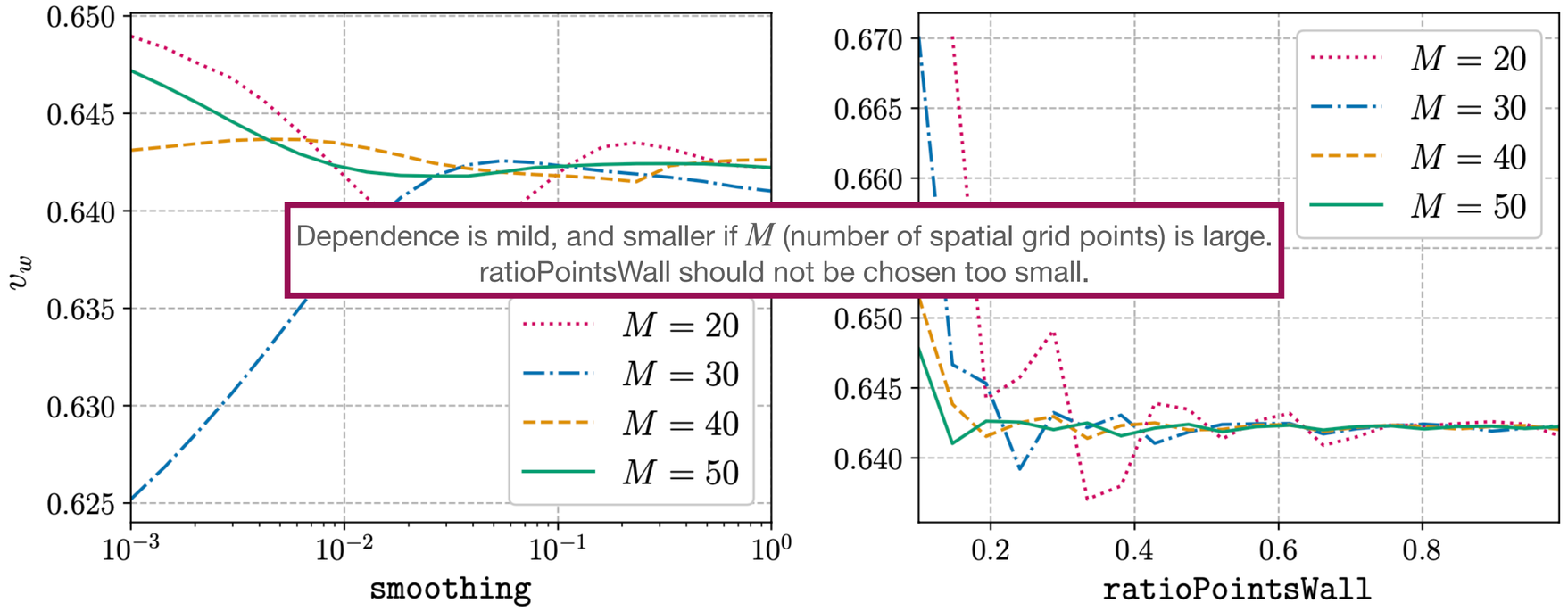


Figure 5: Dependence of v_w on different values of `smoothing` (left) and `ratioPointsWall` (right) in the xSM. We used `ratioPointsWall` $= 0.5$ in the left plot, `smoothing` $= 0.1$ in the right plot, and `meanFreePathScale` $= 50/T_\mathrm{n}$ for both.

Configuration settings (these come from the default file)

Equation of motion

```python
@dataclass
class ConfigEOM:
    """ Holds the config of the EOM class. """

    errTol: float = 1e-3
    """ The absolute error tolerance for the wall velocity result. """

    pressRelErrTol: float = 0.1
    """ Relative error tolerance for the pressure. """

    maxIterations: int = 20
    """ Maximum number of iterations for the convergence of the pressure. """

    conserveEnergyMomentum: bool = True
    r"""
    Flag to enforce conservation of energy and momentum. Normally, this should be set to
    True, but it can help with numerical stability to set it to False. If True, there is
    an ambiguity in the separation between :math:`f_{eq}` and :math:`\delta f` when the
    out-of-equilibrium particles form a closed  system (or nearly closed). This can lead
    to a divergence of the iterative loop. In the end, it is better to set this to False
    if most of the degrees of freedom are treated as out-of-equilibrium particle. If
    most of the dofs are in the background fluid, setting it to True will give better
    results.
    """

    wallThicknessBounds: list[float] = field(default_factory=lambda: [0.1, 100.0])
    """ Lower and upper bounds on wall thickness (in units of 1/Tnucl). """

    wallOffsetBounds: list[float] = field(default_factory=lambda: [-10.0, 10.0])
    """ Lower and upper bounds on wall offset. """

    ## The following parameters are only used for detonation solutions ##
    vwMaxDeton: float = 0.99
    """ Maximal Velocity at which the solver will look to find a detonation solution """

    nbrPointsMinDeton: int = 5
    """ Minimal number of points probed to bracket the detonation roots. """

    nbrPointsMaxDeton: int = 20
    """ Maximal number of points probed to bracket the detonation roots. """

    overshootProbDeton: float = 0.05
    """
    Desired probability of overshooting a root. Must be between 0 and 1. A smaller value
    will lead to more pressure evaluations (and thus a longer time), but is less likely
    to miss a root.
    """
```

Configuration settings (these come from the default file)

Hydrodynamics

```
78  @dataclass
79  class ConfigHydrodynamics:
80      """ Holds the config of the Hydrodynamics class. """
81
82      tmin: float = 0.01
83      """ Minimum temperature that is probed in Hydrodynamics (in units of Tnucl). """
84
85      tmax: float = 10.0
86      """ Maximum temperature that is probed in Hydrodynamics (in units of Tnucl). """
87
88      relativeTol: float = 1e-6
89      """ Relative tolerance used in Hydrodynamics. """
90
91      absoluteTol: float = 1e-10
92      """ Absolute tolerance used in Hydrodynamics. """
93
```

Configuration settings (these come from the default file)

Thermodynamics

```python
@dataclass
class ConfigThermodynamics:
    """ Holds the config of the Hydrodynamics class. """

    tmin: float = 0.8
    """
    Minimum temperature used in the phase tracing (in units of the estimate for the
    minimum temperature obtained in the template model).
    """

    tmax: float = 1.2
    """
    Maximum temperature used in the phase tracing (in units of the estimate for the
    maximum temperature obtained in the template model).
    """

    phaseTracerTol: float = 1e-6
    """
    Desired accuracy of the phase tracer and the resulting FreeEnergy interpolation.
    """

    phaseTracerFirstStep: float | None = None
    r"""
    Starting step for phaseTrace. If a float, this gives the starting step
    size in units of the maximum step size :py:data:`dT`. If :py:data:`None` then
    uses the initial step size algorithm of :py:mod:`scipy.integrate.solve_ivp`.
    """
```

Choosing these numbers closer to one reduces time spent in the initialization phase. The risk of choosing it too close is that the free energy is not known in the relevant temperature range.

Can be tweaked if the thermodynamics functions seem ill-behaved

Configuration settings (these come from the default file)

Boltzmann

```python
@dataclass
class ConfigBoltzmannSolver:
    """ Holds the config of the BoltzmannSolver class. """

    basisM: str = 'Cardinal'
    """ The position polynomial basis type, either 'Cardinal' or 'Chebyshev'. """

    basisN: str = 'Chebyshev'
    """ The momentum polynomial basis type, either 'Cardinal' or 'Chebyshev'. """

    collisionMultiplier: float = 1.0
    """
    Factor multiplying the collision term in the Boltzmann equation. Can be used for
    testing or for studying the solution's sensibility to the collision integrals. Don't
    forget to adjust meanFreePathScale accordingly if this is different from 1
    (meanFreePathScale should scale like 1/collisionMultiplier).
    WARNING: THIS CHANGES THE COLLISION TERMS WRT TO THEIR PHYSICAL VALUE.
    """
```

Useful for debugging purposes/testing how strong the dependence on the collision terms is.
Should be set to 1 if you want to make a physical prediction!
If set to a large value, it will return the LTE result.

Model-dependent settings

`temperatureVariationScale` and `fieldValueVariationScale`

- Inside `ExampleInputPoint`

```
700                 WallGo.VeffDerivativeSettings(
701                     temperatureVariationScale=10.0, fieldValueVariationScale=[10.0, 10.0]
702                 ),
```

- Quantities that help to estimate ΔT and $\Delta \phi$ for numerical derivatives
- A reasonable estimate is $T_{\text{crit}} - T_{\text{nucl}}$ and the vacuum expectation value

Model-dependent settings

`temperatureVariationScale` and `fieldValueVariationScale`

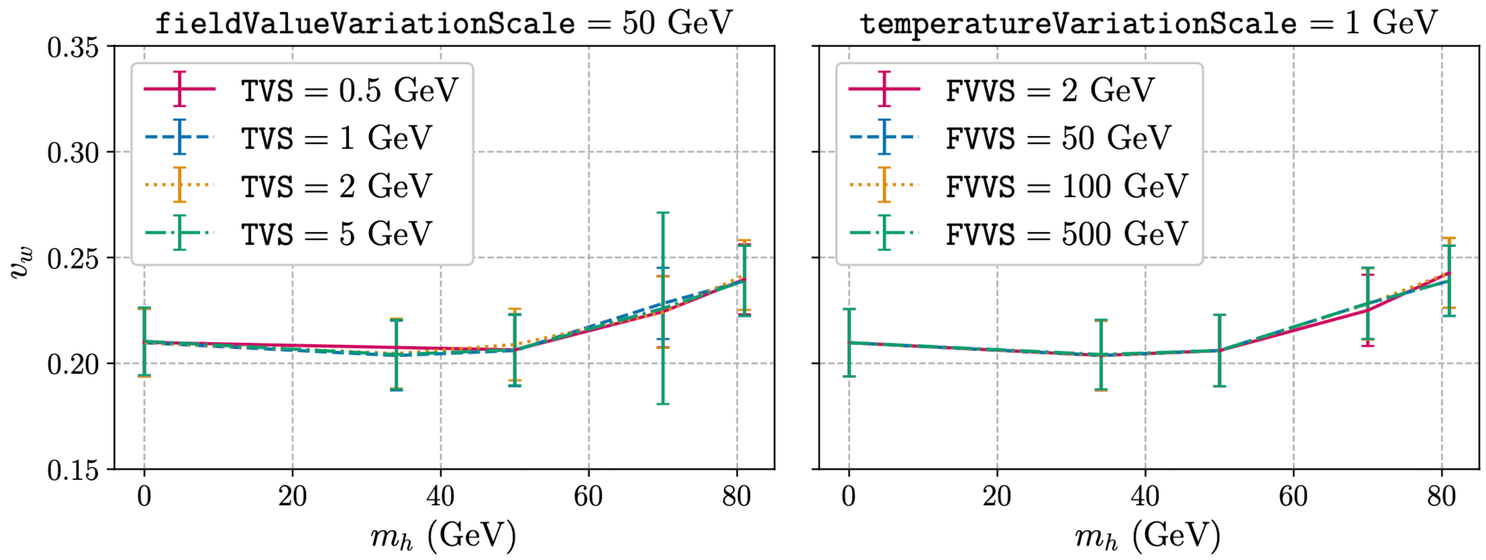


Figure 6: Dependence of v_w on different values of `temperatureVariationScale` (`TVS`, left) and `fieldValueVariationScale` (`FVVS`, right) for the Standard Model with a light Higgs (cf. section 6.2).

Model-dependent settings

`temperatureVariationScale` and `fieldValueVariationScale`

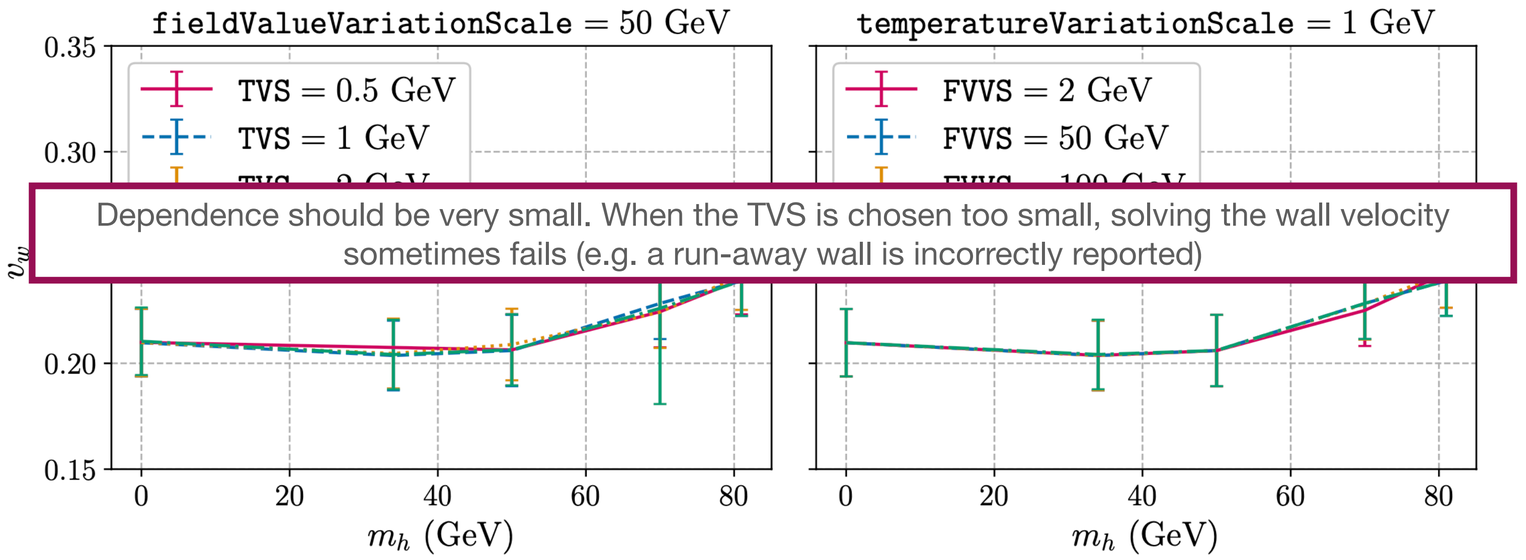


Figure 6: Dependence of v_w on different values of `temperatureVariationScale` (`TVS`, left) and `fieldValueVariationScale` (`FVVS`, right) for the Standard Model with a light Higgs (cf. section 6.2).

Model-dependent settings

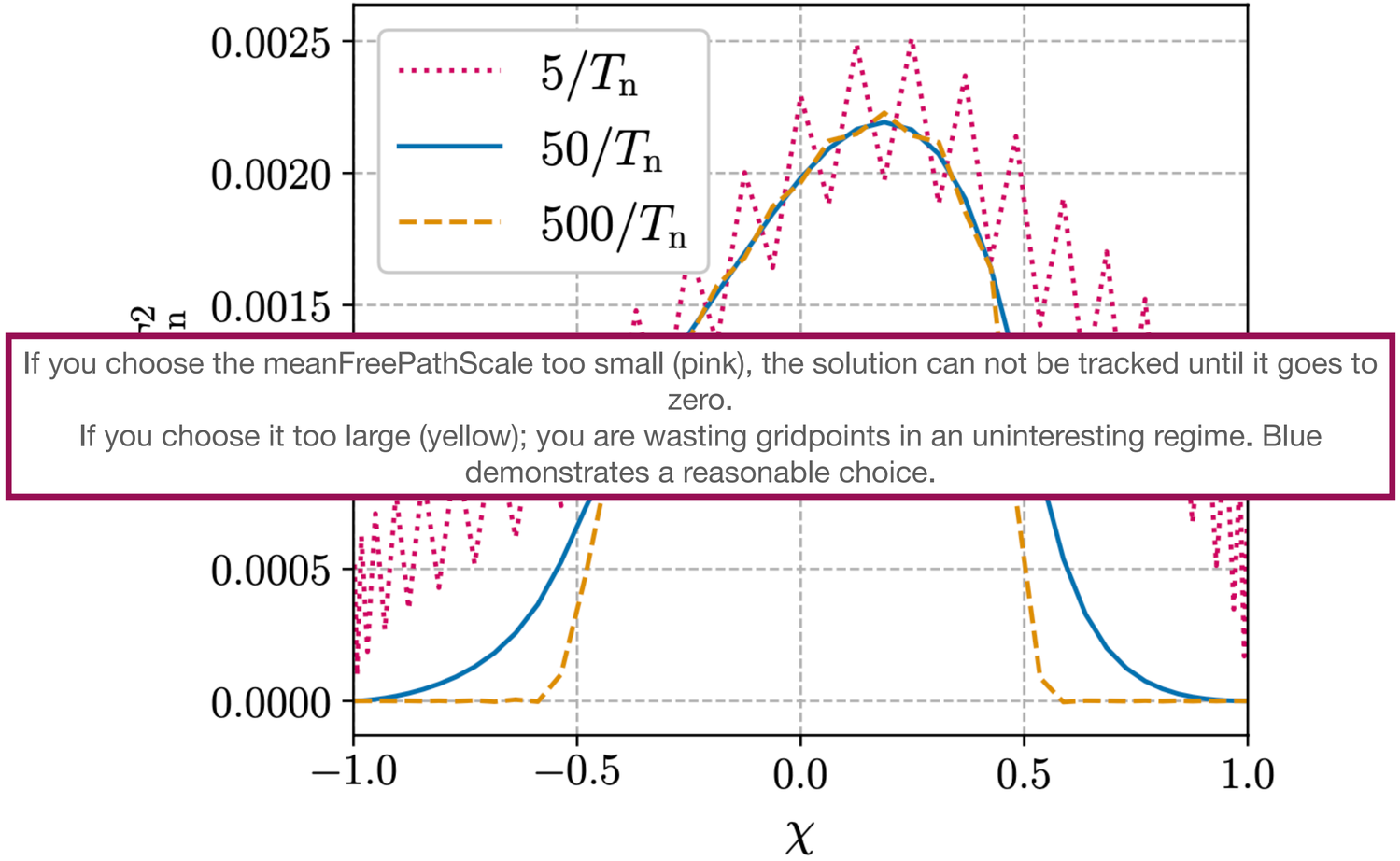
`mean`

- Inside `ExampleInputPoint`

```
555                    WallGo.WallSolverSettings(
556                        # we actually do both cases in the common example
557                        bIncludeOffEquilibrium=True,
558                        meanFreePathScale=100.0, # In units of 1/Tnucl
559                        wallThicknessGuess=20.0, # In units of 1/Tnucl
560                    ),
```

- MeanFreePathScale: estimates the decay length of the out-of-equilibrium particles. This example is for the case of strong interactions.

meanFreePathScale



If you choose the meanFreePathScale too small (pink), the solution can not be tracked until it goes to zero.
If you choose it too large (yellow); you are wasting gridpoints in an uninteresting regime. Blue demonstrates a reasonable choice.

Quesitons?