Collaborative Research Center TRR 257

Particle Physics Phenomenology after the Higgs Discovery

Tracking Minima, Phase Transitions and Gravitational Waves with BSMPTv3

talk based on [arXiv:2404.19037]

Lisa Biermann[1] ✉

with:

Philipp Basler, Margarete Mühlleitner[1], Jonas Müller, Rui Santos[2, 3], João Viana[2]

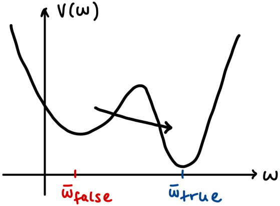
[1]Institute for Theoretical Physics (ITP)
Karlsruhe Institute of Technology (KIT)

[2]Centro de Física Teórica e Computacional
Faculdade de Ciências, Universidade de Lisboa

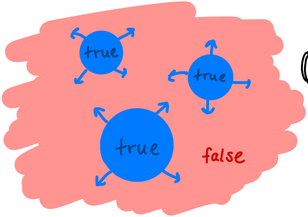
[3]ISEL - Instituto Superior de Engenharia de Lisboa
Instituto Politécnico de Lisboa

CATCH22+2

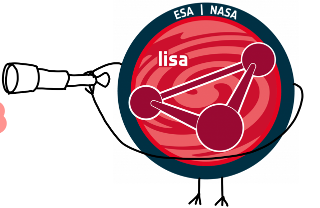
A Glimpse of the Early Universe through Phase Transitions



First-order phase transition (FOPT) in the early universe



True-vacuum bubble nucleation and expansion into hot plasma



Gravitational waves (GWs) in range of LISA (mHz)

- Models *beyond* the SM can undergo a first-order electroweak phase transition (FOEWPT) (between barrier-separated false vacuum and true vacuum)
  → Strong FOEWPT is necessary ingredient for electroweak baryogenesis
- FOPT leads to formation of true vacuum bubbles which expand into surrounding plasma
- Interactions with plasma (and collisions of bubbles) source GWs *within* sensitivity of LISA

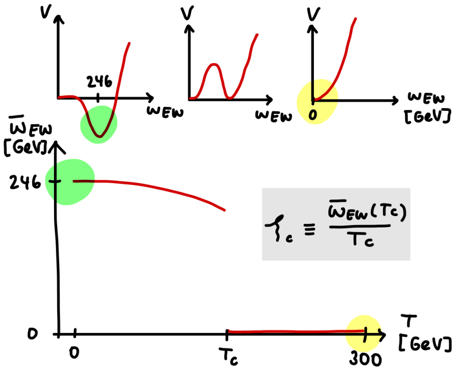
→ *High-interest*: multiple talks at CATCH22+2 about this topic

⇒ `BSMPTv3`: First public code that provides whole chain from particle physics model to GWs!

BSMPTv1/v2

[Basler, Mühlleitner, Müller, '18/'20]

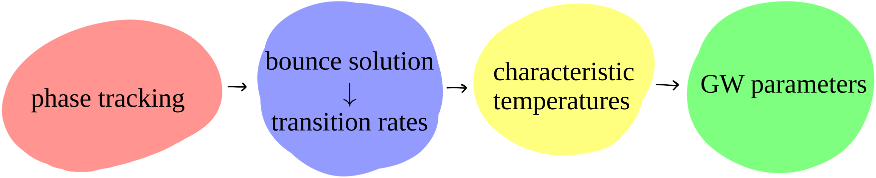
- Implementation of one-loop daisy-resummed effective potential at finite temperature
- *On-shell* renormalization scheme
- Critical temperature: via discontinuity in global EW minimum in $\{0, 300\}$ GeV requiring:
  - → EW symmetry restoration at 300 GeV
  - → EW VEV of 246 GeV at 0 GeV
- Calculation of strength $\xi_c \equiv \overline{\omega}_{\text{EW}}(T_c)/T_c$
- Loop-corrected zero-temperature effective trilinear Higgs self-couplings



- Baryon asymmetry calculation for the complex Two-Higgs Doublet Model (C2HDM)
- Can use input from `ScannerS` [Coimbra et al., '13; Mühlleitner et al., '20]: allowed parameter regions compatible w/ theor. and exp. constraints (using e.g. `HiggsTools` [Bahl et al., '22], `MicrOMEGAs` [Bélanger et al., '02-'23])
- Models already implemented: SM + singlet, SM + doublet (CP-conserving and CP-violating), SM + doublet + singlet
- Easy implementation of new models ☛details

Motivation of BSMPTv3

[Basler, LB, Mühlleitner, Müller, Santos, Viana, '24]

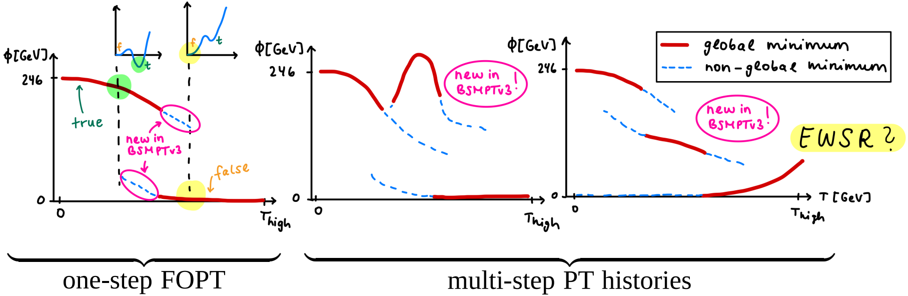


BSMPTv3 extends BSMPTv1/v2 by asking and answering the following questions:

- How does the temperature-dependent multi-dimensional minima landscape of the effective potential look like?
- Does a transition between the false and the true vacuum occur?
- And if: does it complete?
- What is the released energy and timescale of the transition?
- What is the GW peak frequency, peak amplitude, and signal-to-noise-ratio at LISA?

BSMPTv3 — phase tracking

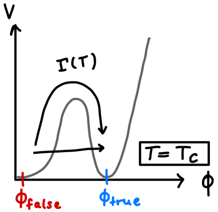
phase = temperature-dependent minimum



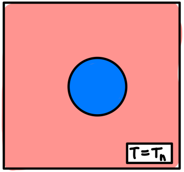
- *Local* minima-tracing (using numerical gradient/Hessian of effective potential) across user-defined temperature range
- Identification of overlap regions between false and true phase
- Identification of multi-step PT histories
- Additional features:
  - *Discrete symmetries*: identification and mapping to 'principal quadrant'
  - *Flat directions*: automatized mapping to lower-dimensional potential
  - *Electroweak symmetry-restoration check* (at high temperatures): derive EWSR behaviour from high-T-const. Hessian matrix

BSMPTv3 — transition rate and characteristic temperatures

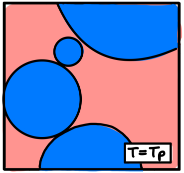
- Calculation of the high-temperature transition rate $\Gamma(T)$ between false and true phases
  $\rightarrow$ Find the *bounce solution*
- Derivation of characteristic temperature scales of PT:
  critical T_c, nucleation T_n, percolation T_p and completion temperature T_f



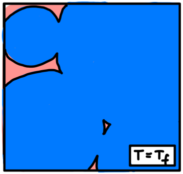
$T = T_c$:
false and true minimum are degenerate
discontinuity in VEVs of global minimum



$T = T_n$:
transition rate matches Hubble rate
$\Gamma(T_n) \equiv H^4(T_n)$



$T = T_p$:
percolation cluster formed, 71 % left in false vacuum
$(P_f(T_p) \equiv 0.71)$

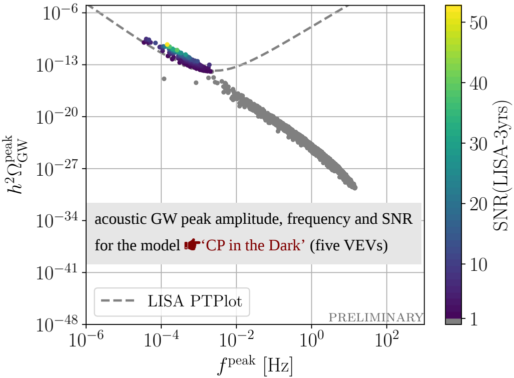


$T = T_f$:
1 % left in false vacuum
$(P_f(T_f) \equiv 0.01)$
$\rightarrow$ PT completed

optional user input: value of $P_f(T_p)$, $P_f(T_f)$

BSMPTv3 — gravitational waves

- Spherical symmetry breaking during bubble expansion through hot plasma generates *gravitational waves*!
- Implemented in BSMPTv3:
  - Sound waves
    [Giblin, Mertens '13/14; Hindmarsh et al., '14/15]
  - Magneto-hydrodynamic turbulence
    [Caprini, Durrer '06]
    [Kahniashvili, Kisslinger, Stevens '08/10]



calculated | calculated | optional user input

- GW spectrum determined by: released latent heat, inverse time scale, wall velocity

  → Peak frequency and peak amplitude calculated

- Signal-to-noise ratio at LISA [Caprini et al., '19]

$$\mathrm{SNR}(\mathcal{T}) = \sqrt{\mathcal{T}\int_{f_{\rm min}}^{f_{\rm max}} \mathrm{d}f \left[\frac{h^2\Omega_{\rm GW}(f)}{h^2\Omega_{\rm Sens}(f)}\right]^2}$$

$h^2\Omega_{\rm Sens}$	nominal LISA sensitivity
$\mathcal{T}$	exp. acquisition time
$f_{\rm min}$, $f_{\rm max}$	LISA sensitivity range

  - In BSMPT: $\mathrm{SNR}(3\,\text{years})$ calculated
  - For $\mathcal{Y}$ years: $\mathrm{SNR}(\mathcal{Y}) = \sqrt{\frac{\mathcal{Y}}{3}}\mathrm{SNR}(3\,\text{years})$

Installation and Usage

- BSMPT is open source: `https://github.com/phbasler/BSMPT` (documentation)
- Questions, comments: `bsmpt@lists.kit.edu` and discussions

```
[lisa@pc: ~]$ pip3 install cmake conan
[lisa@pc: ~]$ git clone git@github.com:phbasler/BSMPT.git
[lisa@pc: ~]$ cd BSMPT
[lisa@pc: BSMPT]$ python3 Build.py
======== Input profiles ========
Profile host:
[settings]
arch=x86_64
build_type=Release
compiler=gcc
compiler.cppstd=gnu17
compiler.libcxx=libstdc++11
compiler.version=11
os=Linux
[...]
[lisa@pc: BSMPT]$ ls build/linux-x86_64-release/bin/
benchmarks  BSMPT  CalcCT  CalcGW  CalcTemps  GenericTests  MinimaTracer
NLOVEV  PotPlotter  standalone  Test  TripleHiggsCouplingsNLO  VEVEVO
[lisa@pc: BSMPT]$
```

get required packages

clone the repository

run installation script

available executables

Installation and Usage

- New executables of BSMPTv3:
  - MinimaTracer: tracing of minima as function of temperature
  - CalcTemps: calculation of characteristic temperatures for all found FOPTs
  - CalcGW: calculation of GW spectrum + SNR for all found FOPTs
  - PotPlotter: visualization of multi-dimensional potential contours

---

```
[lisa@pc: BSMPT/build/linux-x86_64-release]$ ./bin/CalcGW --help
CalcGW calculates the gravitational wave signal
it is called by

        ./bin/CalcGW model input output firstline lastline

or with arguments

        ./bin/CalcGW [arguments]

with the following arguments, ([*] are required arguments, others are optional):

argument                  default   description
--help                              shows this menu
--model=                            [*] model name
--input=                            [*] input file (in tsv format)
--output=                           [*] output file (in tsv format)
--firstline=                        [*] line number of first line in input file
                                        (expects line 1 to be a legend)
--lastline=                         [*] line number of last line in input file
--thigh=                  300       high temperature [GeV]
[...]
[lisa@pc: BSMPT/build/linux-x86_64-release]$
```

Status Quo: Available Public Codes

- `CosmoTransitions` [Wainwright, '11]: phase tracing, bounce solution, T_c, T_n^{approx}
- `Vevacious`, `VevaciousPlusPlus` [Camargo-Molina, O'Leary, Porod, Staub, '13]: finding minima
- `AnyBubble` [Masoumi, Olum, Wachter, '17]: bounce solution
- `EVADE` [Hollik, Weiglein, Wittbrodt, '18, + Ferreira, Mühlleitner, Santos '19]: finding minima, bounce solution
- `BubbleProfiler` [Athron, Balázs, Bardsley, Fowlie, Harries, White, '19]: bounce solution
- `PhaseTracer` [Athron, Balázs, Fowlie, Zhang, '20]: phase tracing, T_c
- `SimpleBounce` [Sato, '21]: bounce solution
- `FindBounce` [Guada, Nemevšek, Pintar, '20]: bounce solution
- `OptiBounce` [Bardsley, '22]: bounce solution

$\Rightarrow$ `BSMPTv3`: phase tracing, bounce solution, characteristic temperatures, GW parameters

Comparison along four points:

- User interface
- Runtime
- Results
- Complicated histories

User Interface

	CosmoTransitions	BSMPTv3
	python library	C++ package
models	user-defined	already implemented: SM, SM + singlet, SM + doublet (CP-conserving and CP-violating), SM + doublet + singlet; user-defined
usage	write own python code using routines, write model implementation	run executables with theor./exp. valid parameter points (generated via ScannerS)
renormalization	$\overline{\text{MS}}$-scheme	OS-scheme
input for model implementation	tree-level potential, full-field-dependent boson and fermion masses	scalar-coupling tensors, finite CTs for OS-scheme → automatized with SymPy and Maple model generation interface ☛details new stand-alone features*

* new stand-alone features of BSMPTv3: minima tracking, bounce solution, temperatures, GW spectrum directly for user-defined potential function (no model implementation needed!) ‘

Runtime

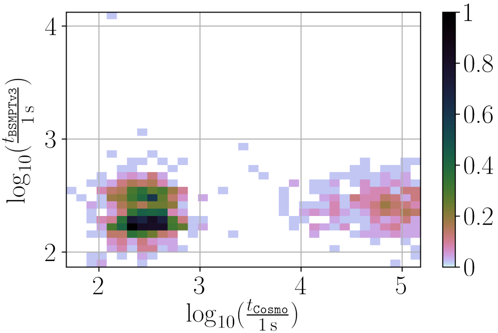
m_{H_a} [GeV]	m_{H_b} [GeV]	m_A [GeV]	$m_{H^\pm}$ [GeV]	c_{H_bVV}	$\tan\beta$	m_{12}^2 [GeV^2]
125.09	[30, 1500]	[30, 1500]	[150, 1500]	[−0.3, 0.3]	[0.8, 25]	[1×10^{-3}, 5×10^{5}]

Table 3: Scan ranges for the CP-conserving 2HDM type 1 in the input parameters used by ScannerS.

- CP-conserving Two-Higgs Doublet Model (type 1) with **four** VEV directions

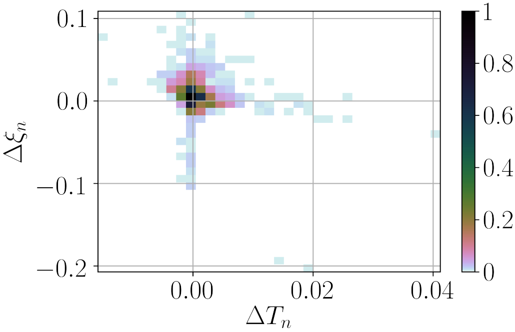
$$\Phi_1 = \frac{1}{\sqrt{2}}\begin{pmatrix}\rho_1 + i\eta_1 \\ \zeta_1 + \boldsymbol{\omega_1} + i\psi_1\end{pmatrix}, \quad \Phi_2 = \frac{1}{\sqrt{2}}\begin{pmatrix}\rho_2 + \boldsymbol{\omega}_{\mathrm{CB}} + i\eta_2 \\ \zeta_2 + \boldsymbol{\omega_2} + i(\psi_2 + \boldsymbol{\omega}_{\mathrm{CP}})\end{pmatrix}$$

- Broad parameter scan with `ScannerS`, `HiggsTools`
- Comparison between `BSMPTv3` and `CosmoTransitions` (for same-transitions subset)



- `BSMPTv3`: mean (median) runtime of 4.2 min (3.5 min)
- `CosmoTransitions`: mean (median) runtime of 41.5 min (5.6 min)
- `BSMPTv3` up to $\times10^3$ faster than `CosmoTransitions`

Results



$$\Delta T_i = \frac{\left(T_i^{\texttt{BSMPTv3}} - T_i^{\texttt{Cosmo}}\right)}{T_i^{\texttt{BSMPTv3}}}$$

$$\Delta \xi_i = \frac{\left(\xi_i^{\texttt{BSMPTv3}} - \xi_i^{\texttt{Cosmo}}\right)}{\xi_i^{\texttt{BSMPTv3}}}$$

with $\quad \xi_i = \frac{\sqrt{\sum_k \omega_k^2(T_i)}}{T_i}$

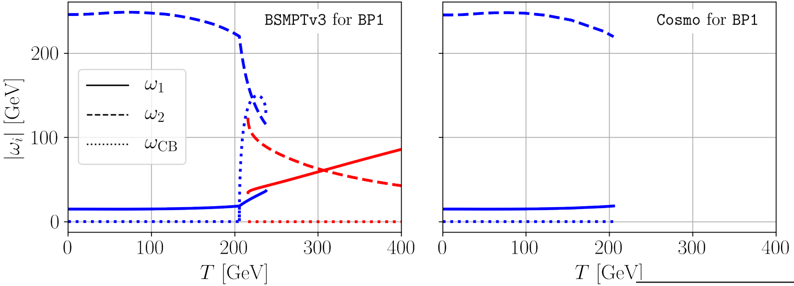
and $\quad \omega_k \in \{\omega_{\text{CB}}, \omega_1, \omega_2, \omega_{\text{CP}}\}$

- Mean (median) relative differences:
  - $\Delta T_c < 0.1\,\%$ (critical temperature)
  - $\Delta T_n < 1\,\%$ (nucleation temperature)
- Outliers in $\Delta\xi_n$ correlated w/ rapidly changing potential in small T interval

Complicated Histories

point from: [Aoki, LB, Borschensky, Ivanov, Mühlleitner, Shibuya, ’23]

BP1: $\text{type} = 1\,,\ \lambda_1 = 6.931\,,\ \lambda_2 = 2.631\,,\ \lambda_3 = 1.287\,,\ \lambda_4 = 4.772\,,\ \lambda_5 = 4.728\,,$
$m_{12}^2 = 1.893 \times 10^4\,\text{GeV}^2\,,\ \tan\beta = 16.578\,.$



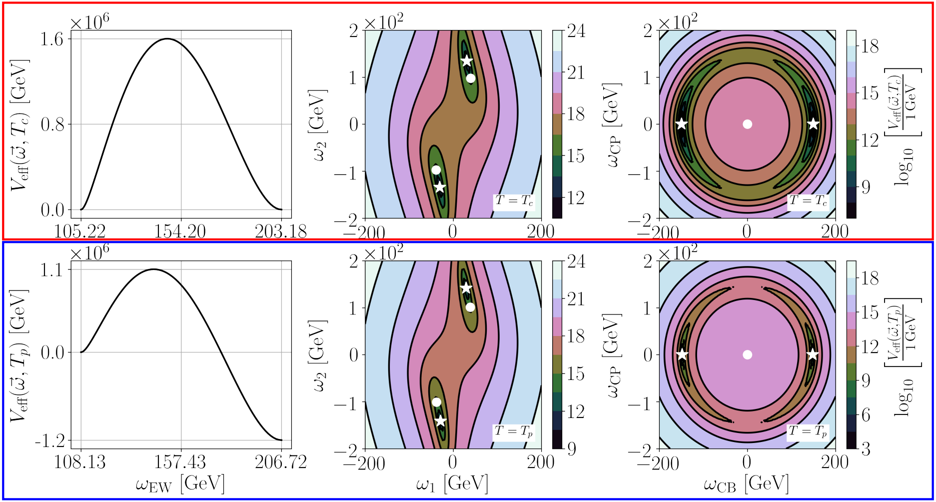
- 2HDM-point for which three field directions develop a non-zero VEV: $\{\omega_1, \omega_2, \omega_{CB}\}$ (intermediate charge-breaking (CB) phase!)
- First-order PT from neutral to CB phase
- Second-order PT to neutral, EW minimum
- BSMPTv3 tracks phases, calculates temperatures in $< 7\,\text{min}$
- CosmoTransitions fails to trace phases for $T > 206\,\text{GeV}$

	BP1
$\text{phases}_{\texttt{BSMPT}}$	0: {216, 400}
	1: {0, 237}
$\text{pairs}_{\texttt{BSMPT}}$	0: $[0 \to 1]$ {216, 237}
$t_{\texttt{MinimaTracer}}$	41.47 s
T_c	226.3
T_n	{222.9, 222.9}
T_p	222.6
T_f	222.6
$t_{\texttt{CalcTemps}}$	6.87 min
history	$0 - (0) \to 1$
$\text{phases}_{\texttt{Cosmo}}$	{0, 206}
$T_{\text{crit},\texttt{Cosmo}}$	–
$T^{\text{approx}}_{\text{nucl},\texttt{Cosmo}}$	–
$t_{\texttt{Cosmo}}$	3.95 s

Complicated Histories - `BP1` visualized with `PotPlotter`

$$\omega_{\text{EW}} = \sqrt{\sum_{i=1,2,\text{CB},\text{CP}} \omega_i^2}$$

critical temperature T_c



percolation temperature T_p

slice from *false* to *true* vacuum

ω_1-ω_2-contour

ω_{CB}-ω_{CP}-contour

Why BSMPTv3?

→ The first public (open-source) code that implements the full chain from particle physics model to gravitational waves

→ Optimized phase tracking over any temperature interval

→ Numerical derivation of bounce solution for any number of field dimensions

→ Besides critical and nucleation, calculation of percolation and completion temperatures

→ Able to treat multi-step PTs, discrete symmetries, flat directions, check for EWSR, report of transition history

→ Calculation of PT parameters and peak frequency/amplitude for (acoustic and turbulence) GW spectrum

→ Computation of signal-to-noise-ratio at LISA

→ For all implemented models (CxSM, R2HDM, C2HDM, N2HDM, CP in the Dark) and *beyond*: (stand-alone features [new in `v3`] + model implementation interface [unchanged from `v1/v2`])

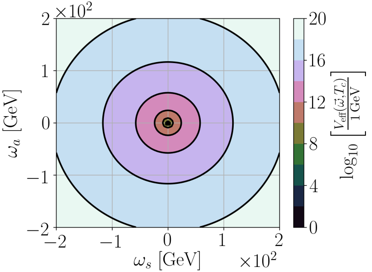
→ Embedded in the existing `BSMPT` code (triple Higgs couplings, EWBG calculation for C2HDM, can use `ScannerS` input)

→ On average faster than `CosmoTransitions` (with overall agreement) and can deal (better) with higher dimensional potentials/complicated PT histories

Thanks!

☛ https://github.com/phbasler/BSMPT
ℹ https://arxiv.org/abs/2404.19037
☺ https://github.com/phbasler/BSMPT/discussions
✉ mailto:bsmpt@lists.kit.edu

Phase Tracking with Discrete Symmetries and Flat Directions in BSMPTv3



BP3: $v = 246.22\,\text{GeV}\,,\ v_s = 0\,\text{GeV}\,,\ v_a = 0\,\text{GeV}\,,\ m^2 = -15\,650\,\text{GeV}^2\,,$
$b_2 = -8859\,\text{GeV}^2\,,\ \lambda = 0.52\,,\ \delta_2 = 0.55\,,\ d_2 = 0.5\,,$
$a_1 = 0\,\text{GeV}^3\,,\ b_1 = 0\,\text{GeV}^2\,.$

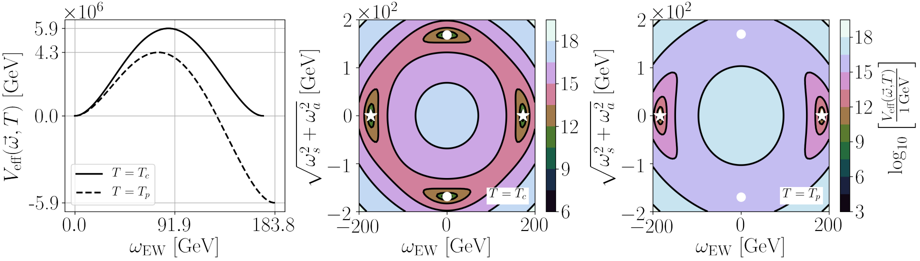
$$V = \frac{m^2}{2}\Phi^\dagger\Phi + \frac{\lambda}{4}\left(\Phi^\dagger\Phi\right)^2 + \frac{\delta_2}{2}\Phi^\dagger\Phi|\mathbb{S}|^2 + \frac{b_2}{2}|\mathbb{S}|^2 + \frac{d_2}{4}|\mathbb{S}|^4 + \left(\frac{b_1}{4}\mathbb{S}^2 + a_1\mathbb{S} + c.c.\right)$$

$$\Phi = \frac{1}{\sqrt{2}}\begin{pmatrix} G^+ \\ \omega_{\text{EW}} + h + iG^0 \end{pmatrix}\,,$$

$$\mathbb{S} = \frac{1}{\sqrt{2}}\left(s + \omega_s + i\left(a + \omega_a\right)\right)\,,$$

$$V \propto (\omega_s^2 + \omega_a^2)^2 \equiv \omega_s^2$$

$$V(\omega_{\text{EW}}, \omega_s) = V(-\omega_{\text{EW}}, \omega_s) = V(\omega_{\text{EW}}, -\omega_s)$$



Wall velocity in BSMPTv3

see [Athron et al., '23] for a review and references within

By default $v_w = 0.95$, or set to user input or one of the following estimates:

- Estimate by [Lewicki et al., '22] (assuming steady-state ($\dot{v}_b = 0$) and local thermal equilibrium):

$$v_b \simeq \begin{cases} \sqrt{\frac{\Delta V}{\alpha \rho_\gamma}} & \text{if} \quad \sqrt{\frac{\Delta V}{\alpha \rho_r(T_*)}} < v_{\text{CJ}} \\ 1 & \text{if} \quad \sqrt{\frac{\Delta V}{\alpha \rho_r(T_*)}} > v_{\text{CJ}} \end{cases}$$

$$\rho_r(T_*) = \frac{\pi^2}{30} g^*(T_*) T_*^4$$

rel. matter density

$$\Psi = \frac{\omega_t}{\omega_f} \quad \text{enthalpy ratio}$$

$$a = 0.2233 \quad \text{num. fit result}$$

$$b = 1.704 \quad \text{num. fit result}$$

$$p = -3.433 \quad \text{num. fit result}$$

$$c_s = \frac{1}{\sqrt{3}} \quad \text{sound speed}$$

- Estimate by [Laurent et al., '23] (assuming local thermal equilibrium):

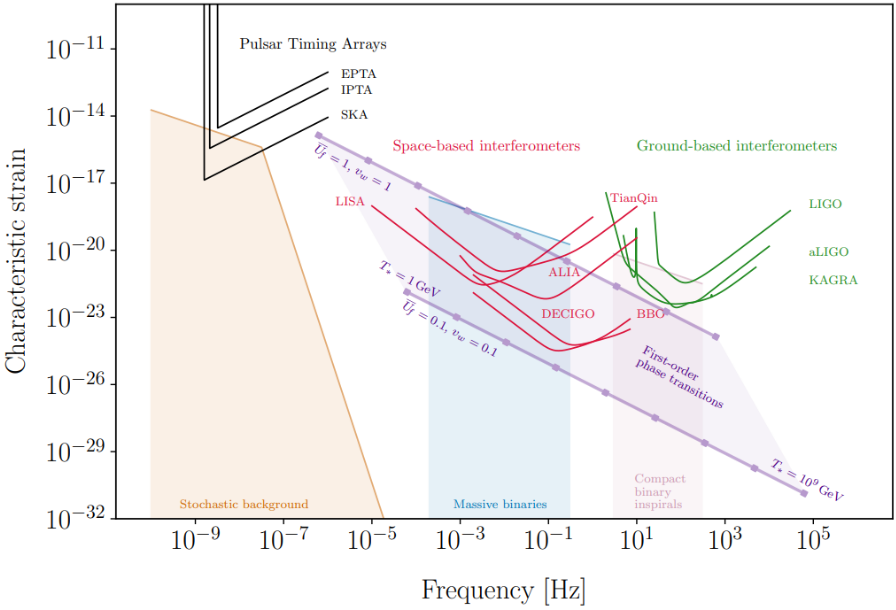
$$v_b = \left(\left| \frac{3\alpha + \Psi - 1}{2\left(2 - 3\Psi + \Psi^3\right)} \right|^{\frac{p}{2}} + \left| v_{\text{CJ}} \left(1 - a \frac{(1-\Psi)^b}{\alpha} \right) \right|^{\frac{p}{2}} \right)^{\frac{1}{p}}$$

with Chapman-Jouguet velocity $v_{\text{CJ}} = \frac{1}{1+\alpha}\left(c_s + \sqrt{\alpha^2 + \frac{2}{3}\alpha} \right)$

- Estimates of v_b in *local thermal equilibrium* serve as **upper bound** as v_b gets reduced by non-equilibrium effects!

LISA sensitivity vs. FOPT

Fig. taken from [Athron et al., ’23]

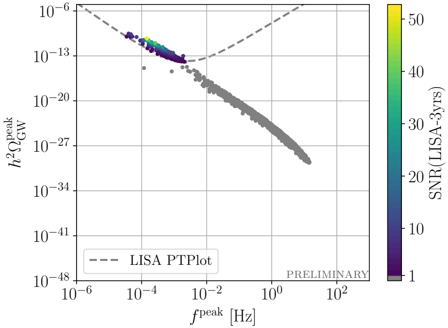


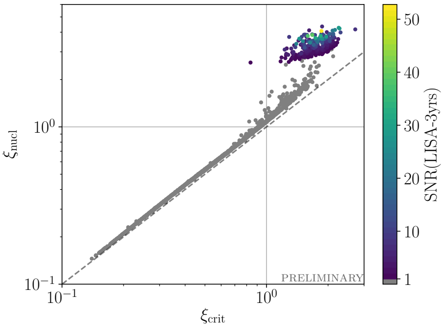
Results for ‘CP in the Dark’ with CalcGW

[Azevedo, Ferreira, Mühlleitner, Patel, Santos, Wittbrodt, ’18]
[LB, Mühlleitner, Müller, ’22/’23]
[LB, Mühlleitner, Santos, Viana, to appear]

- N2HDM-like extended scalar sector, *one* discrete $\mathbb{Z}_2$ symmetry: $\Phi_1 \to +\Phi_1\,, \quad \Phi_2 \to -\Phi_2\,, \quad \Phi_S \to -\Phi_S$

$$V^{(0)} = m_{11}^2|\Phi_1|^2 + m_{22}^2|\Phi_2|^2 + \frac{m_S^2}{2}\Phi_S^2 + \left(A\Phi_1^\dagger\Phi_2\Phi_S + h.c.\right) + \frac{\lambda_1}{2}|\Phi_1|^4 + \frac{\lambda_2}{2}|\Phi_2|^4$$
$$+ \lambda_3|\Phi_1|^2|\Phi_2|^2 + \lambda_4|\Phi_1^\dagger\Phi_2|^2 + \frac{\lambda_5}{2}\left[\left(\Phi_1^\dagger\Phi_2\right)^2 + h.c.\right] + \frac{\lambda_6}{4}\Phi_S^4 + \frac{\lambda_7}{2}|\Phi_1|^2\Phi_S^2 + \frac{\lambda_8}{2}|\Phi_2|^2\Phi_S^2$$





- find SNR(LISA-3yrs) > 10 in agreement with theor. and exp. constraints
- points with SNR(LISA-3yrs) > 10 have $\xi_n > \xi_c \gtrsim 1$ (condition for strong-FOPT)

CP-conserving Two-Higgs Doublet Model

$$V_{\text{tree}} = m_{11}^2 \Phi_1^\dagger \Phi_1 + m_{22}^2 \Phi_2^\dagger \Phi_2 - \left[m_{12}^2 \Phi_1^\dagger \Phi_2 + \text{h.c.} \right] + \frac{1}{2}\lambda_1 (\Phi_1^\dagger \Phi_1)^2 + \frac{1}{2}\lambda_2 (\Phi_2^\dagger \Phi_2)^2 + \lambda_3 (\Phi_1^\dagger \Phi_1)(\Phi_2^\dagger \Phi_2) + \lambda_4 (\Phi_1^\dagger \Phi_2)(\Phi_2^\dagger \Phi_1) + \left[\frac{1}{2}\lambda_5 (\Phi_1^\dagger \Phi_2)^2 + \text{h.c.} \right] .$$

$$\Phi_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} \rho_1 + i\,\eta_1 \\ \zeta_1 + \omega_1 + i\,\psi_1 \end{pmatrix} , \quad \Phi_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} \rho_2 + \omega_{\text{CB}} + i\,\eta_2 \\ \zeta_2 + \omega_2 + i\,(\psi_2 + \omega_{\text{CP}}) \end{pmatrix}$$

$$\{\omega_{\text{CB}},\, \omega_1,\, \omega_2,\, \omega_{\text{CP}}\}|_{T=0} = \{0, v_1, v_2, 0\}\,, \text{ with}$$
$$\omega_{\text{EW}}|_{T=0} \equiv \left. \sqrt{\omega_1^2 + \omega_2^2 + \omega_{\text{CB}}^2 + \omega_{\text{CP}}^2} \right|_{T=0} = \sqrt{v_1^2 + v_2^2} \equiv v = 246 \text{ GeV}$$

Model Implementation with BSMPT — Maple

- model-worksheet: `BSMPT/tools/ModelGeneration/Maple/CreateModel.mw`
- *plug-and-play*:

```
> restart; with(LinearAlgebra) : with(CodeGeneration) : with(VectorCalculus) :
> CodeGeneration:-LanguageDefinition:-Define("MyC", extend = "C", SetLanguageAttribute("Name_IsValid" = true));
> interface(rtablesize = 12) :
> interface(warnlevel = 0) :
```

Higgs potential

In this section the scalar potential is defined. As an example the N2HDM is shown

Define higgs fields

```
> higgsbase := [rho1, rho2, eta1, eta2, psi1, psi2, zeta1, zeta2, zetaS];
```

$$higgsbase := [\rho1, \rho2, \eta1, \eta2, \psi1, \psi2, \zeta1, \zeta2, zetaS]$$

Assign vevs at T=0

```
> higgsvev := [0, 0, 0, 0, 0, 0, v1, v2, vs];
```

$$higgsvev := [0, 0, 0, 0, 0, 0, v1, v2, vs]$$

Assign vevs at T != 0

```
> higgsvevFiniteTemp := [0, wcb, 0, 0, 0, wcp, w1, w2, ws];
```

$$higgsvevFiniteTemp := [0, wcb, 0, 0, 0, wcp, w1, w2, ws]$$

Replacement list for the vevs

```
> VEVRep := {seq(higgsbase[i] = higgsvev[i], i = 1 ..nops(higgsbase))};
```

$$VEVRep := \{\eta1 = 0, \eta2 = 0, \psi1 = 0, \psi2 = 0, \rho1 = 0, \rho2 = 0, \zeta1 = v1, \zeta2 = v2, zetaS = vs\}$$

Replacement list set fields zero

Replacement list set fields zero

```
> RepHiggsZero := {seq(higgsbase[i] = 0, i = 1 ..nops(higgsbase))};
```

$$RepHiggsZero := \{\eta1 = 0, \eta2 = 0, \psi1 = 0, \psi2 = 0, \rho1 = 0, \rho2 = 0, \zeta1 = 0, \zeta2 = 0, zetaS = 0\}$$

Define number of Higgses

```
> nHiggs := nops(higgsbase);
```

$$nHiggs := 9$$

Define parameters of the potential

```
> par := [m11Sq, m22Sq, m12Sq, L1, L2, L3, L4, L5, msSq, L6, L7, L8];
```

$$par := [m11Sq, m22Sq, m12Sq, L1, L2, L3, L4, L5, msSq, L6, L7, L8]$$

Define Higgs doublet

...

Model Implementation with BSMPT — python

- SymPy toolkit in: BSMPT/tools/ModelGeneration/sympy/
- Need to write MODEL.py (provided for reference: SM.py and G2HDM.py (generic 2HDM))
- Excerpt from SM.py:

```python
[...]
# parameters
msq, la = symbols('msq lambda', real=True)
params=[msq,la]
# fields
rho,eta,zeta,psi = symbols('rho eta zeta psi', real=True)
# VHiggs
phi = Matrix([[rho+I*eta], [zeta+I*psi]]) * 1/sqrt(2)
phiSq = simplify((Dagger(phi)*phi)[0])
VHiggs = msq/2 * phiSq + la/factorial(4) * phiSq**2
# VGauge
W1, W2, W3, B0 = symbols('W1 W2 W3 B0',real=True)
Dmu = -I*Cg/2 * (sigma1*W1 + sigma2 * W2 + sigma3*W3) -I*Cgs/2 * sigma0 * B0
VGauge = simplify(Dagger(Dmu*phi)*(Dmu*phi))[0,0]
[...]
# Generate the model
toyModel = ModelGenerator.ModelGenerator(params,dparams,CTTadpoles,Higgsfields,VHiggs,\
                                         zeroTempVEV,finiteTempVEV)
toyModel.setGauge([W1,W2,W3,B0],VGauge)
toyModel.setLepton(LepBase, VFLep)
toyModel.setQuark(QuarkBase, VQuark)
```

- Get scalar-coupling tensors and finite counterterms:

```
# display tensors
[lisa@pc: ~]$ python3 MODEL.py --show tensors
# show finite counterterms
[lisa@pc: ~]$ python3 MODEL.py --show ct
```

Stand-alone Features of BSMPTv3

- Exemplary shown here: `BSMPT/standalone/CalculateAction.cpp`
- Calculation of Euclidean action for user-defined potential and initial guess path
- Calculation using analytical derivatives possible, if gradient of potential is provided

---

```
// Define the potential
std::function<double(std::vector<double>)> V = [&](std::vector<double> x)
{
  double c  = 5;
  double fx = 0;
  double fy = 80;

  double r1 = x[0] * x[0] + c * x[1] * x[1];
  double r2 = c * pow(x[0] - 1, 2) + pow(x[1] - 1, 2);
  double r3 = fx * (0.25 * pow(x[0], 4) - pow(x[0], 3) / 3.);
  r3 += fy * (0.25 * pow(x[1], 4) - pow(x[1], 3) / 3.);

  return (r1 * r2 + r3);
};

// Define the false and true vacuum
std::vector<double> FalseVacuum = {0, 0};
std::vector<double> TrueVacuum  = {1, 1};

// Your best guess for the path
std::vector<std::vector<double>> path = {TrueVacuum, FalseVacuum};

 // Calculate the action
BounceActionInt bc(path, TrueVacuum, FalseVacuum, V, 0, 6);
bc.CalculateAction();

std::cout << "Action calculated using numerical derivatives is " << bc.Action
          << "\n";
```