

Automatic extraction of one-loop Wilson coefficients in general BSM scenarios using MARTY-1.4

Grégoire Uhlich,^{a,*} Farvah Mahmoudi^{a,b} and Alexandre Arbey^{a,b}

^aUniversité de Lyon, Université Claude Bernard Lyon 1, CNRS/IN2P3,
Institut de Physique des 2 Infinis de Lyon, UMR 5822, F-69622, Villeurbanne, France

^bTheoretical Physics Department, CERN, CH-1211 Geneva 23, Switzerland

E-mail: marty.uhlrich@gmail.com

We present a fully automated procedure providing an easy way to perform, systematically, phenomenological analyses in flavor physics for general BSM scenarios. This procedure relies on MARTY-1.4, is model independent and requires as input only the Lagrangian of the theory. Once the Lagrangian has been defined, tree-level and one-loop Wilson coefficients can be calculated symbolically by MARTY, from which flavor observables can be computed numerically by available software programs. We focus in particular on $b \rightarrow s\gamma$ and the recently updated $b \rightarrow s\ell^+\ell^-$ observables which are in tension with the SM, and present a general procedure to extract the relevant one-loop coefficients, such as C_7 , C_7' , C_9 and C_{10} .

*** The European Physical Society Conference on High Energy Physics (EPS-HEP2021), ***

*** 26-30 July 2021 ***

*** Online conference, jointly organized by Universität Hamburg and the research center DESY ***

*Speaker

1. Introduction

In the following we present a sample program using MARTY-1.4 [1] to calculate, in general BSM scenarios, one-loop Wilson coefficients for flavor physics such as $C_7^{(\prime)}$ and $C_{9(10)}$. These coefficients which are relevant to describe the $b \rightarrow s$ transitions are defined from the corresponding effective Hamiltonian [2], namely

$$\mathcal{H}_{\text{eff}} \ni -\frac{4G_F}{\sqrt{2}} V_{tb} V_{ts}^* \left[\frac{e}{16\pi^2} m_b C_7^{(\prime)} (\bar{s} \sigma^{\mu\nu} P_{R(L)} b) F_{\mu\nu} + \frac{e^2}{16\pi^2} C_{9(10)} (\bar{s} \gamma^\mu P_L b) (\bar{\mu} \gamma_\mu (\gamma^5) \mu) \right], \quad (1)$$

with b , s and μ the SM fermionic fields and $F_{\mu\nu}$ the photon's field strength.

The calculation of these coefficients using MARTY is model-independent and is not designed specifically for $C_7^{(\prime)}$ or $C_{9(10)}$. As presented in the next section, the procedure is very general and can be used to derive other coefficients, at the tree level or for other types of effective operators. Once these values are calculated by MARTY symbolically, numerical values of flavor observables can be obtained using SuperIso [3] for example.

2. The program

For more information about MARTY (installation instructions, manuals, documentation or related publications) see <https://marty.in2p3.fr>.

Model definition This part is the only model-dependent part. It is possible to use built-in models in MARTY such as the SM or the MSSM, or define a custom Lagrangian. The program we present next only requires the model to have fermions named "b", "s" and "mu", and the photon "A".

```
PMSSM_Model model; // built-in pMSSM in MARTY
// SM_Model model; // built-in SM in MARTY
```

Settings for the calculation In order to extract the Wilson coefficients, MARTY needs the convention-dependent global factor defined in the effective Hamiltonian (can be identified in equation 1) and the order of the external fermions in the operator because it is not uniquely defined. For the factor, it is first necessary to create its symbolic expression (using SM parameters in `mt::sm_input::`), namely:

```
Expr factor = -pow_s(e_em, 4) * GetComplexConjugate(V_ts)
* V_tb / (32 * CSL_PI*CSL_PI * M_W*M_W * s2_theta_W);
```

Then, a `FeynOptions` object can be created to contain the factor and the fermion order:¹

¹The fermions are defined from the order given to the `computeWilsonCoefficients()` method. In the $b \rightarrow s\mu\mu$ process for example, the order (1, 0, 2, 3) applied to the particles ($b, \bar{s}, \bar{\mu}, \mu$) corresponds to an operator of the type $(\bar{s}b)(\bar{\mu}\mu)$.

```
FeynOptions options;
options.setWilsonOperatorCoefficient(factor);
options.setFermionOrder({1, 0, 2, 3});
```

More options can be set to customize the calculation, for example:

```
options.setTopology(Topology::Triangle); // Only triangles (3-point functions)
options.addFilters(
  filter::disableParticle("A"), // No photon
  filter::forceParticle("t") // Only top quark contributions
);
```

Extracting $C_{9(10)}$ Once the options are set, the calculation of the $b \rightarrow s\mu\mu$ decay can be performed by MARTY simply using:

```
auto wilsons = model.computeWilsonCoefficients(
  OneLoop,
  {Incoming("b"), Outgoing("s"), Outgoing("mu"), Outgoing(AntiPart("mu"))},
  options // The options as last parameter
);
```

In order to extract the coefficients in the amplitude, MARTY now has to identify operators in the result. These must be provided by the user but a simple interface exists to create any fermion currents,² in this case $d = 6$ operators:

```
auto O9 = dimension6Operator(model, wilsons, DiracCoupling::VL, DiracCoupling::V);
auto O10 = dimension6Operator(model, wilsons, DiracCoupling::VL, DiracCoupling::A);
```

Using the fermion order defined in the options, it is possible to give the Dirac couplings corresponding to the two currents (Scalar (S), Pseudoscalar (P), Left (L), Right (R), Vector (V), Axial (A), Vector Left (VL), Vector Right (VR), Tensor (T), Tensor Axial (TA), Tensor Left (TL) or Tensor Right (TR)). Once O_9 and O_{10} have been defined, the coefficients can be extracted:

```
Expr C9 = getWilsonCoefficient(wilsons, O9);
Expr C10 = getWilsonCoefficient(wilsons, O10);
```

²It is also possible to define general effective operators in MARTY based on a technique similar to the one used for the creation of Lagrangian terms.

Finally, MARTY can generate the library containing the numerical functions evaluating C_9 , C_{10} and the tree-level spectrum generator for the model:

```
Library lib("wilson_pmssm"); // Library called "wilson_pmssm"
lib.generateSpectrum(model); // Provides the spectrum generator for the model
lib.addFunction("C9_pmssm", C9);
lib.addFunction("C10_pmssm", C10);
lib.build(); // Creates and builds the library automatically
```

Extracting $C_7^{(\prime)}$ For the extraction of $C_7^{(\prime)}$ in the $b \rightarrow s\gamma$ decay, the factor must be modified and the calculation is different:³

```
auto wilsons = model.computeWilsonCoefficients(
    OneLoop,
    {Incoming("b"), Outgoing("s"), Outgoing("A")},
    options // The options as last parameter
);
auto O7 = chromoMagneticOperator(model, wilsons, DiracCoupling::R); // P_R
auto O7p = chromoMagneticOperator(model, wilsons, DiracCoupling::L); // P_L
Expr C7 = getWilsonCoefficient(wilsons, O7);
Expr C7p = getWilsonCoefficient(wilsons, O7p);
```

3. Conclusion

We presented a general procedure to extract C_7 , C_7' , C_9 and C_{10} in any BSM model using MARTY-1.4. Such a procedure is very useful to obtain, in a fully automated way, the loop-level predictions of flavor observables in general BSM scenarios that is a very long and difficult task by hand. Furthermore, MARTY is not limited to these Wilson coefficients and can also be used to extract coefficient of general $d \leq 6$ operators. Considering also the capability of MARTY to calculate squared amplitudes at tree-level or one-loop for general BSM, this software program allows us to automate theoretical predictions at the loop level for general BSM scenarios.

References

- [1] G. Uhlich, F. Mahmoudi, and A. Arbey, *Computer Physics Communications* **264**, 107928 (2021), arXiv:2011.02478.
- [2] A. J. Buras, Weak Hamiltonian, CP violation and rare decays, in *Les Houches Summer School in Theoretical Physics, Session 68*, 1998, arXiv:hep-ph/9806471.
- [3] F. Mahmoudi, *Computer Physics Communications* **178**, 745 (2008), arXiv:0710.2067.

³The function is called `chromoMagneticOperator()` because it works for general external bosons, including those with a non-trivial gauge representation such as the gluon for which an algebra generator has to be supplemented in the operator.