

# MHS EXAMPLES

JULIAN ROSEN

This document lists some example computations for the Python 2.7 package `mhs`. The software accompanies the papers [2] and [3], and makes use of computations from the MZV data mine [1]. All of the examples in this document can be found in the iPython notebook `mhs_examples.ipynb`. The most recent version can always be found at

<https://sites.google.com/site/julianrosen/mhs>.

## 1. COMPUTATIONS

First, we import the `mhs` package. The default weight is 8.

```
>>> from mhs import *  
Imported data through weight 9
```

The package uses a basis for the space of multiple harmonic sums. Let's see the basis.

```
>>> mhs_basis()  
[(), (2, 1), (4, 1), (4, 1, 1), (6, 1), (5, 2, 1), (6, 1, 1)]
```

In other words, modulo  $p^9$ , every weighted multiple harmonic sum can be written as a rational linear combination of

$$1 = h_p(\emptyset), h_p(2, 1), h_p(4, 1), h_p(4, 1, 1), h_p(6, 1), h_p(5, 2, 1), h_p(6, 1, 1).$$

The computer can compute expansions in terms of this basis.

```
>>> a = Hp(1, 2, 3); a.disp()  
 $H_{p-1}(1, 2, 3)$ 
```

```
>>> a.mhs()
```

---

*Date:* February 22, 2019.

$$-2H_{p-1}(4, 1, 1) - \frac{11}{28}pH_{p-1}(6, 1) + p^2 \left( -\frac{27}{7}H_{p-1}(5, 2, 1) - 9H_{p-1}(6, 1, 1) \right) + O(p^3)$$

The computer is telling us that the congruence

$$H_{p-1}(1, 2, 3) \equiv -2H_{p-1}(4, 1, 1) - \frac{11}{28}pH_{p-1}(6, 1) + p^2 \left( -\frac{27}{7}H_{p-1}(5, 2, 1) - 9H_{p-1}(6, 1, 1) \right) \pmod{p^3}$$

holds for all sufficiently large  $p$ , and indeed the computer has found a *proof* of this congruence.

We can also find expansions in terms of  $p$ -adic multiple zeta values. For example, binomial coefficients.

```
>>> b = binp(5,2); b.display()
```

$$\binom{5p}{2p}$$

```
>>> b.mzv()
```

$$10 - 300p^3\zeta_p(3) - 5700p^5\zeta_p(5) + 4500p^6\zeta_p(3)^2 - 108300p^7\zeta_p(7) + 171000p^8\zeta_p(3)\zeta_p(5) + O(p^9)$$

This is a lot of output. If we only want an expansion modulo e.g.  $p^6$ :

```
>>> b.mzv(err=6)
```

$$10 - 300p^3\zeta_p(3) - 5700p^5\zeta_p(5) + O(p^6)$$

Another quantity we can compute:

```
>>> c = curp(2,4); c.display()
```

$$\sum_{\substack{n_1+n_2+n_3+n_4=p^2 \\ p \nmid n_1n_2n_3n_4}} \frac{1}{n_1n_2n_3n_4}$$

```
>>> c.mhs()
```

$$-\frac{24}{5}p^2H_{p-1}(4, 1) + \frac{28}{15}p^3H_{p-1}(4, 1, 1) + O(p^4)$$

If we want higher precision, we should import data of higher weight.

```
>>> import_data(9)
```

Imported data through weight 9

```
>>> c = curp(2,4); c.mhs()
```

$$-\frac{24}{5}p^2H_{p-1}(4,1)+\frac{28}{15}p^3H_{p-1}(4,1,1)+p^4\left(-\frac{24}{5}H_{p-1}(4,1)+\frac{4421}{420}H_{p-1}(6,1)\right)+O(p^5)$$

The software can also handle various other nested sums. For example, we can compute something like

$$\sum_{n=1}^{p-1} \frac{(-1)^n}{n+2} \binom{p+n+1}{n-1} \binom{p}{n}.$$

```
>>> import_data(8)
Imported data through weight 8
>>> a = (BIN(1,-1)*BINN(0,0)*nn(-2)).sum(1,0).e_p()
>>> a.mhs()
```

$$\begin{aligned} & \frac{1}{2} - \frac{5}{4}p + \frac{15}{8}p^2 - \frac{41}{16}p^3 + p^4 \left( \frac{99}{32} - \frac{1}{3}H_{p-1}(2,1) \right) - \frac{221}{64}p^5 \\ & + p^6 \left( \frac{471}{128} + \frac{5}{12}H_{p-1}(2,1) - \frac{7}{30}H_{p-1}(4,1) \right) + O(p^7) \end{aligned}$$

```
>>> a.mzv()
```

$$\begin{aligned} & \frac{1}{2} - \frac{5}{4}p + \frac{15}{8}p^2 - \frac{41}{16}p^3 + p^4 \left( \frac{99}{32} - \zeta_p(3) \right) - \frac{221}{64}p^5 \\ & + p^6 \left( \frac{471}{128} + \frac{5}{4}\zeta_p(3) - 3\zeta_p(5) \right) + O(p^7) \end{aligned}$$

The software uses MathJax to render the output in a Jupyter notebook. If you would prefer to get non-TeX output, this can be done.

```
>>> a.mzv(tex=False)
'1/2 - 5/4p + 15/8p^2 - 41/16p^3 + p^4(99/32 - zeta_p(3)) - 221/64p^5
+ p^6(471/128 + 5/4zeta_p(3) - 3zeta_p(5)) + 0(p^7)'
```

## REFERENCES

- [1] Johannes Blumlein, DJ Broadhurst, and Jos AM Vermaseren. The multiple zeta value data mine. *Computer Physics Communications*, 181(3):582{625, 2010.
- [2] Julian Rosen. The MHS algebra and supercongruences. 2016, arxiv:1608.06864.
- [3] Julian Rosen. The completed finite period map and Galois theory of supercongruences. *International Mathematics Research Notices*, 2018.