Hilbert modular forms for $\mathbb{Q}(\sqrt{5})$, $\mathbb{Q}(\sqrt{13})$ and $\mathbb{Q}(\sqrt{17})$

Description of the Maple$^{\mathrm{TM}}$ worksheets

Sebastian Mayer

August 2007

Abstract

This file together with zip file "hilbert-maple.zip" supplement my "Dissertation": Hilbert Modular Forms for the fields $\mathbb{Q}(\sqrt{5})$, $\mathbb{Q}(\sqrt{13})$ and $\mathbb{Q}(\sqrt{17})$, even if it can be read without this file and the corresponding worksheets. This text describes the Maple$^{\mathrm{TM}}$-files designed to calculate rings of Hilbert modular forms compressed in the zip-file "hilbert-maple.zip". They are especially designed for the calculation of the rings of extended Hilbert modular forms for $\mathbb{Q}(\sqrt{5})$, $\mathbb{Q}(\sqrt{13})$ and $\mathbb{Q}(\sqrt{17})$, but can be adapted to different cases.

Contents

1 General remarks

All calculations are made using Maple 7.0, Waterloo Maple Inc., 2001. These Maple^{TM}-worksheets come without any warranty. They are supposed to give all necessary algorithms for concrete calculations of rings of Hilbert modular forms, but are to be used at own risk. The calculations might be flawed, even if they work allright on my computer, and the algorithms are quite complex and use a lot of RAM, so the computer might well crash at various points during the calculations. It is advisable to save greater changes to the worksheet before executing the worksheet and to have no other unsaved data on the computer.

2 Structure

2.1 General description

The zip-file "`hilbert-maple.zip`" contains this description together with Maple^{TM}-worksheets and Maple^{TM}-data files connected to Hilbert modular forms and used for the calculations necessary in my "Dissertation": Hilbert Modular Forms for the fields $\mathbb{Q}(\sqrt{5})$, $\mathbb{Q}(\sqrt{13})$ and $\mathbb{Q}(\sqrt{17})$. When extracting the zip-file (under linux) in the directory "maple" in your home directory, it will create the following subdirectories (where ... stands for the path of your home-directory):

`.../maple/hilbert`	containing this file and the Maple^{TM}-worksheets
`.../maple/hilbert/data`	containing results and intermediate results of the computations
`.../maple/packages`	empty, to be filled by maple

The files extracted into these directories are of four different types. First there are the files containing the program to calculate a number of modular forms and describe the ring of Hilbert modular forms by a set of generators. These program-files a enumerated from 1 to 10 and have to be executed in the corresponding order. Some routines, most of which are quite simple or are frequently used, are gathered in a "package" contained in the file "00-Package-neu-Hilbert-Borcherdslift.mws" respectively in the file "00b-Procedures-of-Package.mws" (compare Section 2.2.2). It is mandatory to execute these "basic rountines" to be able to execute the other worksheets. Then there are some auxiliary files, for example to calculate all multiplier systems or to show the allready calculated Borcherds products. All of these can be found in the subdirectory .../maple/hilbert. At last there are the files containing the results of the calculations and a great number of intermediate results, which are given in the directory "/hilbert/data" and end with ".m". If you want to change the proposed path structure, just copy the files and change the paths as described in the next section.

2.2 The Basic Routines

There are two ways to use the basic routines. The more convenient one requires a short installation. If this does not work, you can alternatively either copy the necessary procedures into each worksheet you want to use or just execute the basic routines before executing the worksheet.

2.2.1 Installation

The installation adds the basic routines to the standard routines allready provided by ordinary maple. First copy (with linux) the files maple.lib, maple.ind and maple.hdb, which can be probabely found in the directory "/usr/local/maple7/lib/", into the directory ".../maple/packages". If you want to change this path or the path ".../maple/hilbert/data", you can do so by changing the paths given in the procedure "setpath()" in the package "00-Package-neu-Hilbert-Borcherdslift.mws".

Then open the file "00-Package-neu-Hilbert-Borcherdslift.mws" with Maple^{TM} and execute the worksheet (for example choose "Edit", "Execute", "Worksheet" from the menu in Maple^{TM}, at least in Version 7.0). There might occur errors like "During help initialisation - read failed", but all calculations will work fine all the same. This error will cause the help system of Maple not to work while the package is loaded. If you need Maple's help system, open a second (empty) Maple session without loading the package.

2.2.2 Alternatives

If you have problems initialising the package, just, instead of loading the package, evaluate the file "`00b-Procedures-of-Package.mws`" just before you execute a program file and delete the two commands "`restart;`" and "`with(HilbertBorcherdsLifte);`" in the opening of the program file. Here, you can also change the paths by changing them in the procedure "`setpath`". Alternatively you can replace the command "`with(HilbertBorcherdsLifte);`" by a copy of the file "`00b-Procedures-of-Package.mws`" and delete the command "`restart;`". Afterwards the file can be executed on its own.

2.3 Calculations

When doing calculations, use the files in the right order given by the numbers in their names. The auxiliary files are somewhat different, see the following section for their meaning. Each file saves its results into a data-file in the subdirectory "data", automatically choosing an appropriate name (for example depending on p and the given precision), so you do not have to reevaluate all old calculations and it gets easier for maple to cope with the data. If you only want to load the results and do not want to make any new calculations, you can evaluate the corresponding file and set "`laden:=true`", then all results will be loaded (if existing) from the corresponding data file instead of calculating the results. This can also be used to skip single calculations by writing "`laden:=true`" in front and "`laden:=false`" after the command, which might be helpful to locate and mend errors, for example after an adaption of the program to some different number field.

Some of the more involved algorithms print intermediate results or some values of some counters, in order to reassure the user that they are still working and sometimes make it possible to estimate the remaining time. Some of the algorithms may even take about a day, but at least every hour some result should be printed, of course depending on the desired precision and on your computer.

2.4 The Precision of the Calculations

The algorithms are not always optimized for precision and might calculate terms of higher order than can be granted by the precision of the input. So the user needs to know the precision he can expect and to truncate the results to this precision. To do this, use the routines `qrestrict`(f,n)$= f(z) + o(q^n)$

and `restrict(`f`,[`a`,`b`])`$=\ f(\tau)+o(g^a)+o'(h^b)+\omega(h^{-b})$, where o is the Landau symbol defined by

$$f=o(g)\ \Leftrightarrow\ \lim_{\tau\to\infty}\left|\frac{g(\tau)}{f(\tau)}\right|=0$$

and by o' we denote

$$f=o'(g)\ \Leftrightarrow \lim_{\tau\to\infty}\left|\frac{f(\tau)}{g(\tau)}\right|=0.$$

In other words `restrict(`f`,[`a`,`b`])` calculates f with precision $g^a h^{\pm b}$.

2.5 Files

Most of the files are designed to be used for several values of $p\in\{5,13,17\}$. Just change the variable p to the desired case. If the algorithms depend on p (to some extend), then there are different files for one type of calculations. Of course, if you treat $p\notin 5,13,17$, you will have to occasionally change the Maple$^{\text{TM}}$-commands.

- `00-Package-neu-Hilbert-Borcherdslift.mws`

  Various procedures needed for the calculations.

- `00b-Procedures-of-Package.mws`

  All the procedures of the file 00-Package-neu-Hilbert-Borcherdslift.mws, not as a package, but just as procedures.

- `01-Sm-berechnen.mws`

  Calculate the sets S_m.

- `02-fm-berechnen.mws`

  Calculate the $f_m\in\mathcal{A}_k(p,\chi_p)$ with $f_m(z)=q^{-m}+O(1)$.

- `03a-RhoW-berechnen.mws`

  Calculate the $R(W,n)$ for $n<0$.

- `03b-RhoW+berechnen.mws`

  Calculate the $R(W,n)$ for $n>0$. This is only needed to simplify the `T[m]`.

- `04-Eisensteinreihen-berechnen.mws`

  Calculate Hilbert Eisenstein series. The second coefficient is calculated as indicated in the paper of Siegel. In file 8, this algorithm is simpified by just using this result, so file 8 is somewhat faster, not having to solve a basis problem in the space of elliptic modular forms. For an adaption to $p \notin \{5, 13, 17\}$ you will need to execute this file and change file 8 appropriately.

- `05-Psi-berechnen.mws`

  Calculate Borcherds products.

- `06-Thetareihen-berechnen.mws`

  Calculate θ-series.

- `06-Thetareihen-p=5.mws`

  θ-series in the special case $p = 5$, especially the ones introduced by Müller (cf. [Mü85]).

- `07-Psi_in_sk_gj_p=5.mws`,
  `07-Psi_p=13_in_bekanntem.mws`,
  `07-Psi_in_bekanntem_p=17.mws`

  Calculate the Fourier expansions of the restrictions of the Borcherds products and find a polynomial in η, E_4 and E_6 equal to the Borcherds product on the diagonal.

- `08-Hilbertsche_Poincare-Reihen_und_Eisensteinreihen.mws`

  Calculate Hilbert Eisenstein series.

- `09-Divisoren-bestimmen.mws`

  Simplify the divisors of the Borcherds products.

- `10-Reduktionsprozess.mws`,
  `10-p=17-Reduktionsprozess.mws`

  The reduction process to find the ring of extended Hilbert modular forms.

- `p=5-Multiplikatorsysteme.mws`

  Calculate all multiplier systems (for homogeneous weights) in case $p = 5$ (there is only the trivial one).

- `p=13-Multiplikatorsysteme.mws`

  Calculate all multiplier systems (for homogeneuous weights) in case $p = 13$.

- `p=17-Multiplikatorsysteme.mws`

  Calculate all multiplier systems (for homogeneuous weights) in case $p = 17$.

- `show-Psi-p=13.mws`

  Show the Fourier expansions of the Borcherds products in case $p = 13$.

- `show-Psi-p=17.mws`

  Show the Fourier expansions of the Borcherds products in case $p = 17$.

- `suche-nach-f1-p=13.mws`

  Find f_1 in case $p = 13$.

- `suche-nach-f1-p=17.mws`

  Find f_1 and f_2 in case $p = 17$.

3 Some Algorithms

There are many algorithms realized in the Maple$^{\mathrm{TM}}$ worksheets, but three of them are most important. Those are the following, of which the first and the second are needed to get sensible outputs, so they are the most important algorithms to get the calculated data and the last one is the cue algorithm in the calculation of the Borcherds products. A description of the algorithm can be found in the section "A Basis for the Plus Space" of the main text.

- restrict(f,$[a,b]$)

  The Fourier expansion of f is given with precision $g^a h^{\pm b}$.

- qrestrict(f,n)

  Truncate Fourier expansion of $f : \mathbb{H} \to \mathbb{C}$ such that qrestrict(f, n) is given of order $o(q^n)$.

- plusspace(n,d)

  The Fourier expansions of the $f_1, \ldots, f_n$ are calculated with precision d.

4 Variables

4.1 Mathematical Variables

Variable	Maple™	explanation
p	`p`	Hilbert modular forms for $\mathbb{Q}(\sqrt{p})$ are investigated
$j^{(p)}$	`jp`	
$H^{(q)} = \frac{\left(\eta^{(p)}\right)^p}{\eta}$	`etaquotientH`	
$H^{(1)} = \frac{\eta^p}{\eta^{(p)}}$	`etaquotientG`	
$\tilde{H}$	no variable	
$= e^{\pi i(\tau_1+\tau_2)}$, $\tau \in \mathbb{H}^2$	`g`	unassigned, used for Fourier expansion
$= e^{\pi i(\tau_1-\tau_2)/\sqrt{p}}$, $\tau \in \mathbb{H}^2$	`h`	unassigned, used for Fourier expansion
$= e^{2\pi iz}$, $z \in \mathbb{H}$	`q`	unassigend, used for Fourier expansions
G_4^*	`G04`	normalized elliptic Eisenstein series
G_6^*	`G06`	normalized elliptic Eisenstein series
Δ^*	`Delta0`	normalized Dirac Δ function
j	`j`	absolute invariant
$z \mapsto j(pz)$	`jp`	
G_2, H_2	`G2, H2`	Eisenstein series of Nebentypus
	`FrakturE2`	Eisenstein series of Haupttypus and of weight 2
E_2^+	`E12`	
J	`J`	holomorphic modular form of weight -2 for $\mathrm{SL}(2,\mathbb{Z})$, $J = q^{-1} + O(1)$
η	`eta`	Dedeking η-function
$q^{-1/24}\eta$	`etaprod`	
$z \mapsto \eta(pz)$	`etap`	
$z \mapsto q^{-p/24}\eta(pz)$	`etapprod`	
	`Loesungen[m]`	A set of representatives of the subset of $\mathfrak{o}/\sqrt{p}$ of elements with with norm $-m$ with respect to multiplication with ε_0^2
$R(W, -m)$	`R[m]`	like `Loesungen[m]`, but normalized

4.2 Non-mathematical Variables

`Variable`	Description
`gen`	Precision up to which polynomials are expanded ($O(q^{\mathrm{gen}+1})$).
`mmax, fmax`	Highest index of calculated divisors, Borcherds products etc.
`inhalt`, `inhalt`$\mathcal{X}$	Both variables contain information on the content of the "`.m`"-file they are saved in. The data type is array of pairs of strings. Both variables `inhalt` and `inhalt`$\mathcal{X}$ have the same value and $\mathcal{X}$ takes values according to the number of the file it was generated with. So if you load several files at once, you find the correct description of the content corresponding to file number $\mathcal{X}$ in the array `inhalt`$\mathcal{X}$, since the variable `inhalt` allways contains the informations about the file read last. `inhalt[1]` contains the information which file it is from and what kind of data is saved. The following entries give the names of the saved variables and explain their meaning. So you can read all information of such a file by the command `print(inhalt);`
`laden`	Boolean variable, if `laden=true`, then all results are loaded instead of doing the calculations again, if `laden=false`, then all calculations are redone. `laden=true` is usefull for reading data which was calculated before and for changes in the middle of a worksheet, in this case one writes `laden:=true;` in the beginning of the document and `laden:=false;` just in front of the calculation one wants to redo.