HECKE: Modular Forms Calculator

by William Stein
January 2, 2004

What is Hecke?

Hecke is a C++ program that I wrote in 1999-2000 for computing with modular symbols and modular forms. It implements algorithms of Cremona, Hijikata, Merel, Mestre-Oesterlé, Shimura, and others. It is free software, released under the GPL, and available for Linux and OS X here:

`http://modular.fas.harvard.edu/Tables/hecke-cpp.html`

(There is also a Windows port, but I'm not sure how to get a copy.) Hecke is *not* "rock solid", so you should be skeptical of the results it gives and aware that the interface is primitive. Nonetheless, Hecke is capable of many computations which aren't currently available in any other free package.

Hecke grew out of work on my thesis which involves computing special values of L-functions, congruences, and verifying modularity of certain Galois representations. In a sense, Hecke is also the program I wish had existed when I was taking my first modular forms course and wanted to see lots of concrete examples of modular forms. Some of the tables computed using Hecke can be found at `http://modular.fas.harvard.edu/~was/Tables`.

I stopped new development of Hecke in 2000 and implemented many of the same algorithms (and more) in Magma, which is a closed source, non-free, but not for profit, computer algebra system developed at the University of Sydney. (All of the code I wrote for Magma is written in Magma and can be viewed by anyone.) Fortunately, several volunteers (Alex Brown, Ami Fischman, and Justin Walker) have kept Hecke up-to-date, in the sense that it will compile on modern computers.

Hecke is much different than the modular forms package I wrote in Magma. One major difference is that Magma is a far more complicated system, and using it to do anything nontrivial requires some basic familiarity with how Magma works, which takes some time to acquire. In contrast, Hecke (especially the "msymbols" mode, see below) does nothing besides computations with modular forms, and has menus, so you can immediately start computing Hecke operators and basis. However, when using Hecke you'll probably often import the results of computations to another system, such as PARI, Maple, or Mathematica, and work with them further there. Another key difference is that Hecke computations are persistent, in the sense that Hecke saves the result of *every* computation you do to your hard drive (in a folder called `data`), so next time you do the computation it is loaded from disk. In contrast, the Magma modular forms package as no built-in functionality for saving computations to disk (as of Jan. 2004).

What does HECKE do?

HECKE is an interactive calculator.

- **Modular forms and Hecke operators:** Computations on the spaces of modular forms $M_k(\Gamma_1(N), \varepsilon)$, for $k \geq 2$, over cyclotomic and finite fields. Functions include:
  - Computation of bases of newforms. Within computational limits, the level, weight, and character can be pretty much arbitrary, with the restriction that $k \geq 2$ be an integer. Furthermore, all eigenforms are computed, *not* just the ones with eigenvalues in $\mathbf{Q}$.
  - Computation of the "Birch and Swinnerton-Dyer" rational numbers $L(M_f, i)/\Omega_i$ where M_f is a complex torus attached to f and Ω_i is a certain volume.
  - For optimal quotients A_f of $J_0(N)$ associated to newforms:
    1. The modular degree and group structure of the canonical polarization obtained by pullback of the θ divisor.
    2. Intersection of $A_f^\vee$ and $A_g^\vee$.
    3. Order of image of $(0) - (\infty)$ in $A_f(\mathbf{Q})_{\mathrm{tor}}$.
    4. Multiple of order of $A_f(\mathbf{Q})_{\mathrm{tor}}$.
    5. Tamagawa numbers of quotients of $J_0(p)$, with p prime. (Note: Component group order at p for quotient of $J_0(N)$, when $\mathrm{ord}_p(N) = 1$, is available in MAGMA.)

    (Note: Similar computations for quotients of $J_1(N)$ are mostly implemented in MAGMA, though not well documented as of January 2004.)
  - Discriminants of Hecke algebras.
  - Numerical computation of special values and period lattices of forms of even weight $k \geq 2$, in many (but not all) cases. When f has rational Fourier coefficients, computation of the invariants of the associated elliptic curve over $\mathbf{R}$ (the implementation of this is not nearly as refined as in Cremona's elliptic curve package).
- **Formulas:** The classical formulas, such as the numbers of cusps on modular curves, dimensions of spaces of cusps forms, and computation of $\dim S_k(\Gamma_1(N), \varepsilon)$ for $k \geq 2$ and ε a Dirichlet character modulo N. (Note: Computation of $\dim S_k(\Gamma_1(N), \varepsilon)$ using the Hijikata trace formula is non-optimal; there is a much simpler formula of Cohen and Oesterle, which is what is implemented in MAGMA.)
- **Character groups of tori:** Action of Hecke operators T_ℓ, for $\ell = 2, 3, 5, 7$, on the character group associated to $J_0(p)$ (using the Mestre-Oesterlé graph method). The matrices attained in this way are very sparse.
- **Tables:** Functions for making tables.

Guided tour

In this guided tour, you will see how to use HECKE to compute the action of Hecke operators, bases of eigenforms, and obtain information about special values of L-functions. The output given below is a small subset of the actual output, which is much more verbose.

Starting HECKE

To start HECKE, type `hecke` at the command line. You will see something like this:

```
# hecke
HECKE Version 0.4, Copyright (C) 1999-2003, William A. Stein
HECKE comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions; read the included COPYING file for details.


          HECKE:  Modular Forms Calculator   Version 0.4 (Sep. 24, 2003)

                                William Stein
        Ported to Mac OS X (Justin C. Walker) and Windows (Alex F. Brown)
Send bug reports and suggestions to was@math.harvard.edu.
Type ? for help.

HECKE>
```

Typing `?` gives a list of commands and modes which include:

```
HECKE> ?

  about:     About
  calc:      Motive calculator
  exsymbols: Extended modular symbols mode
  formulas:  Formula calculator
  graphs:    Monodromy pairing calculator
  msymbols:  Modular symbols calculator
  tables:    Table making routines
  quit:      Quit
```

Modular forms and Hecke operators calculator

Type `msymbols` to start the modular forms and Hecke operators calculator. You will be asked for several pieces of information, which define the space on which to work. In this mode you can only compute with one space of modular forms at a time (see the `calc` mode for computations that relate different spaces). Answer as follows:

```
level N = 389
character chi = 0
weight k = 2
```

After a brief computation the calculator interface will print some information about $M_2(\Gamma_0(389))$ and await your command.

```
-----------------------------------------------------------------
Current space:   M_2(Gamma_0(389); Q)^+, dim=33
Hecke action on: V=M_2, dim=33
-----------------------------------------------------------------
M_2(389) ?
```

The `Q)^+` means that we are actually computing with the +1 quotient of modular symbols, so in general some Eisenstein series could be missing, but all cusp forms will be present. Use the `exsymbols` mode for more precise control.

Typing `?` gives a list of help topics.

```
1: computing OPERATORS
2: setting current SPACE
3: cutting out SUBSPACES
4: computing BASIS
5: CONVERSIONS between representations
6: arithmetic INVARIANTS of torus A_V
7: INVARIANTS of Hecke algebra
8: OPTIONS
```

To get an idea of what $M_2(\Gamma_0(389))$ looks like, we compute the characteristic polynomials of several Hecke operators T_n. We do this by typing `t`, then entering a positive integer n.

```
? t
Tn: Enter values of n, then q when done.
2
f2=(x-3)*(x + 2)*(x^2 -2)*(x^3 -4*x -2)*
   (x^20 -3*x^19 -29*x^18 + 91*x^17 + 338*x^16 -1130*x^15
    -2023*x^14 + 7432*x^13 + 6558*x^12 -28021*x^11 -10909*x^10
    + 61267*x^9 + 6954*x^8 -74752*x^7 + 1407*x^6 + 46330*x^5
    -1087*x^4 -12558*x^3 -942*x^2 + 960*x + 148)*
   (x^6 + 3*x^5 -2*x^4 -8*x^3 + 2*x^2 + 4*x -1);
q
```

Let's compute the action of a few Hecke operators on the factor of degree two. Type `subeigenpoly`, then select the appropriate factor:

```
M_2(389) ? subeigenpoly
[...]
n = 2                              <---- you type this
Choose one of the following factors.
```

```
        1: x+2
        2: x-3
        3: x^2-2
        4: x^3-4*x-2
        5: x^20-3*x^19-29*x^18+91*x^17+338*x^16-1130*x^15-2023*x^14+
        7432*x^13+6558*x^12-28021*x^11-10909*x^10+61267*x^9+6954*x^8-
        74752*x^7+1407*x^6+46330*x^5-1087*x^4-12558*x^3-942*x^2+960*x+148
        6: x^6+3*x^5-2*x^4-8*x^3+2*x^2+4*x-1
        7: ALL factors
   Select a factor: 3               <---- you type this
```

When the `M_2(389) ?` prompt appears, type `opmatrix` to turn on matrix display and `opcharpoly` to turn off computation of characteristic polynomials. Now you can compute matrices which represent the Hecke operators on this dimension two space:

```
M_2(389) ? opmatrix
matrix display on
M_2(389) ? opcharpoly
charpoly display off
M_2(389) ? t
2
T2=[2,1;-2,-2];
3
T3=[0,1;-2,-4];
6
T6=[-2,-2;4,6];
q
M_2(389) ? opmatrix
matrix display off
M_2(389) ? opcharpoly
charpoly display on
```

Let A denote the corresponding two-dimensional optimal quotient of $J_0(389)$. To compute the BSD value $L(A,1)/\Omega_A$, type `torusbsd`. HECKE outputs 0 along with the first few terms of the q-expansion of f and the discriminant of the ring $\mathbf{Z}[\ldots,a_n,\ldots]$:

```
\\ L(A_f1,1)/Omega_f = 0
\\ disc(Z[f]) = 2^3
a1 = Mod(a,a^2-2);
f1 = q + (a1)*q^2 + (a1-2)*q^3 + -1*q^5 + (-2*a1+2)*q^6 +
         (-2*a1-1)*q^7 + O(q^8);
```

That $L(A_f,1)=0$ is consistent with the sign of the functional equation for $L(A_f,s)$. The sign in the functional equation for the L-function is minus the sign of the Atkin-Lehner involution W_{389}. To compute this involution, type `actatkin` and then enter `389` for p. HECKE compute that $W_{389}=I$ on A, so the sign in the functional equation is -1 and $L(A,1)$ is forced to vanish.

```
M_2(389) ? actatkin
Compute W_q where p = 389
[...]
charpoly(W389) = (x -1)^2
```

Where it says "Compute W_q where $p =$", the subscript q is the power of the prime p that exactly divides the level.

To obtain the q-expansion of a normalized eigenform in our two-dimensional space, type `basisnew` then `n=7`. The result is

```
s1=t^2-2;  s=Mod(t,t^2-2);
f1 = q + (s)*q^2 + (s-2)*q^3 + -1*q^5 +
        (-2*s+2)*q^6 + (-2*s-1)*q^7 + O(q^8);
```

which means that a normalized newform is

$$f_1 = q + \sqrt{2}q^2 + (\sqrt{2}-2)q^3 - q^5 + (-2\sqrt{2}+2)q^6 + (-2\sqrt{2}-1)q^7 + \cdots$$

The space $S_2(\Gamma_0(389))$ is of particular interest because $p = 389$ is the only prime < 50000 such that p divides the discriminant of the Hecke algebra associated to $S_2(\Gamma_0(p))$. To compute the discriminant of the Hecke algebra $\mathbf{T}$ on $S_2(\Gamma_0(389))$, first switch back to the full cuspidal subspace using the `spacecusp` command.

```
M_2(389) ? ?2

spacecusp       spaceeisenstein     spaceintegral
spacefull       spaceload           spacemkz
spacenew        spacesave

M_2(389) ? spacecusp
-----------------------------------------------------------------
Current space:   M_2(Gamma_0(389); Q)^+, dim=33
Hecke action on: V=S_2, dim=32
-----------------------------------------------------------------
```

Next type `heckedisc` and wait a minute while Hecke operators T_n, for $n \leq 65$ are computed by Hecke in order to compute the discriminant.

```
M_2(389) ? heckedisc
... (wait) ...
disc(T|V) = 1512755021268492313729721471006297487222442965725740
            0066431700067942762387865600000
[It seems to wait forever in the current version, perhaps
 trying to factor the discriminant...]
```

We find that the discriminant of the $\mathbf{Z}$-module generated by $T_1, \ldots, T_{65}$ is

$$5924565544861062256019564094047982931042610200956162134098575360000000$$

$$= 2^{53} \cdot 3^4 \cdot 5^6 \cdot 31^2 \cdot 37 \cdot 97^2 \cdot 389 \cdot 3881 \cdot 215517113148241 \cdot 477439237737571441$$

Nontrivial character and weight

Next, compute a basis of eigenforms for $S_4(\Gamma_0(13), \varepsilon)$ where $\varepsilon : (\mathbf{Z}/13\mathbf{Z}) \to \mathbf{C}^*$ is a character whose image has order 3. Type `x` to quit computing on $M_2(389)$, type `msymbols` again and enter `N = 13`, `chi = 3`, and `k = 4`. In a second, the status display will appear:

```
-----------------------------------------------------------------
Current space:   M_4(Gamma_0(13), eps=[3]; Q[a]/(x^2 + -1 * x + 1))^+, dim=5
Hecke action on: V=M_4, dim=5
-----------------------------------------------------------------
M_4(13) ?
```

(Note: The quadratic polynomial in the quotient should be in terms of a instead of x, but I never implemented this.) Type `basisnew`, then `n = 3` to get the first 3 terms of the q-expansions of a basis of newforms.

```
M_4(13) ? basisnew
max n=3
```

(Note: Only one representative from each Galois conjugacy class of newforms is provided.) The output is

```
f1 = q + (-4*a)*q^2 + (-2*a)*q^3 + O(q^4);
s2=t^2+(-5*a)*t+(2*a-2); s=Mod(t,t^2+(-5*a)*t+(2*a-2));
f2 = q + (s)*q^2 + ((-3)*s+(5*a))*q^3 + O(q^4);
```

This means that there are two conjugacy classes of normalized eigenforms, with representatives f_1 and f_2. The first is $f_1 = q - 4aq^2 - 2aq^3 + \cdots$, where a is a primitive cube root of 1, and the second is $f_2 = q + sq^2 + (-3s + 5a)q^3 + \cdots$ where s is a root of $t^2 - 5at + 2a - 2 = 0$.

To work in fields of characteristic other than 0, use the extended mode by typing `exsymbols` instead of `msymbols` at the `HECKE>` prompt.

Motives associated to modular forms

The `msymbols` mode is useful for computing basis of eigenforms and the action of Hecke operators on spaces of modular forms. It is less useful for computing specific information about the structure of $J_0(N)$. For that, use the `calc` mode. Type `x` to get to the `HECKE>` prompt, then type `calc`. When asked if you want to work in the fast $+1$ quotient, typing `n`. (If you type `yes`, many computations will be an order of magnitude faster, but are likely to be wrong by a power of 2.)

A space (or the corresponding abelian variety) is specified in the `calc` mode as follows:

[level]k[weight][isogeny class].[command][(arguments)]

where the level is an integer, the weight is an integer ≥ 2, the isogeny classes are A, B, C, ..., Z, AA, BB, CC, and the commands will be outlined below. Omitting the weight part of the command is the same as specifying $k = 2$.

First enter the calc mode, then type `125` to obtain a list of optimal quotients of $J_0(125)$.

```
M_4(13) ? q
Leaving modular symbols calculator.
HECKE> ?

  about:     About
  calc:      Motive calculator
  exsymbols: Extended modular symbols mode
  formulas:  Formula calculator
  graphs:    Monodromy pairing calculator
  msymbols:  Modular symbols calculator
  tables:    Table making routines
  quit:      Quit

HECKE> calc
Welcome to the motive calculator.
WORK IN FAST MODE (+1 quotient, certain powers of 2 *wrong*)?   [y] n
Type ? for help.

? 125

 MOTIVECOMMAND = Summarize level.
 ****** SUMMARIZE LEVEL.
125k2  dim      W
A        2      +
B        2      -
C        4      -
```

The first time you type 125, a huge amount of log information will be printed. Simply type 125 again to list the above without the logging information. The second time you compute level 125, it takes no extra time, since everything is stored in the `data` directory. Even if you quit HECKE and restart, it will still list 125 quickly.

This means that $J_0(125) \sim A \times B \times C$ where A, B, C are abelian varieties of dimensions 2, 2, and 4. We can compute $L(A,1)/\Omega_A$, $L(B,1)/\Omega_B$ and $L(C,1)/\Omega_C$:

```
? 125A.bsdratio
0
? 125B.bsdratio
2^2/5
? 125C.bsdratio
1/5
```

You may also type `125a.bsdratio`, as the isogeny code is not case sensitive. The signs in the `W` column above give the signs of the Atkin-Lehner involution W_{125}.

What about the torsion? Type `125.torsionbound(13)` to get an upper bound on the torsion subgroup of $J_0(125)$. Then type `125.cusporder` to compute the order of $(0) - (\infty) \in J_0(125)(\mathbf{Q})$.

```
? 125.torsionbound(13)
5^2
? 125.cusporder
5^2
```

We're lucky because the divisor and multiple match up, so we conclude that $J(\mathbf{Q}) \approx \mathbf{Z}^2 \oplus (\mathbf{Z}/25\mathbf{Z})$. Next type `125A.intersection(B)` to obtain the structure of the finite abelian group $A' \cap B' \subset J$, where A', B' are the abelian varieties dual to A and B. The answer `[2,2,2,2]` indicates that the intersection is $(\mathbf{Z}/2\mathbf{Z})^4$. This implies that the corresponding newforms satisfy a congruence in characteristic 2. To exit `calc` mode, type `\q`.

Unfortunately, I have not written a tutorial for using the extended modular symbols mode or making tables.