Hilbert Symbols

Probably Late

1 Hilbert Symbols over Number Fields

There are many motivations for studying Hilbert symbols over number fields. They give useful information about whether a quaternion algebra is a division ring or a matrix algebra. This information additionally allows us to compute maximal orders of quaternion algebras. [Voight] Away from quaternion algebras, the Hilbert symbol is seen to encode information as to whether the quadratic form $ax^2 + by^2$ represents 1 over a given field. [Voight] Finally, in elliptic curves the Hilbert symbol is used in the algorithm to compute the root number. [Sage Days 22 code]

Throughout this paper, F is a number field with ring of integers $\mathcal{O}_F$ and $B = \left(\frac{a,b}{F}\right)$ is a quaternion algebra over F with basis $1, i, j, ij$ where $i^2 = a, j^2 = b$, and $ij = -ji$. I will assume a working knowledge of quaternion algebras and basic algebraic number theory. For an introduction to quaternion algebras and background for this paper see John Voight, *The arithmetic of quaternion algebras*, book in preparation. `http://www.cems.uvm.edu/~voight/crmquat/book/quat-modforms-041310.pd`

1.1 Valuations.

Let v be a valuation of F. Then the field F_v has ring of integers R_v and let π_v be a uniformizer (denoted by π when v is obvious). Then we can define $B_v = B \otimes F_v$. Then B_v is a quaternion algebra over F_v.

Useful fact about local norms: If F is a number field with noncomplex valuation v, then F_v has a unique unramified quadratic extension K_v. This fact gives us the following:

Lemma 1. *Let v be a noncomplex place of F. Then there is a unique quaternion algebra B_v over F_v which is a division ring up to F_v-algebra isomorphism.*

As $\mathbb{C}$ is algebraically closed, there is no division quaternion algebra. Over $\mathbb{R}$ the unique division algebra is the Hamiltonians, $\mathbb{H} = \left(\frac{-1,-1}{\mathbb{R}}\right)$. Over $\mathbb{R}$, if $B = \left(\frac{a,b}{\mathbb{R}}\right)$ is not a division algebra, then $B \cong M_2(\mathbb{R})$.

If v is nonarchimedean, then F_v has K_v as it's unique unramified extension. Thus to create a division ring over F_v, $B_v \cong \left(\frac{K_v,\pi_v}{F_v}\right)$. Similarly, if B_v is not a division ring, then $B_v \cong M_2(F_v)$.

1.2 Hilbert Symbols

To encode the two possibilities, division ring or matrix algebra, we use the Hilbert symbol.

Definition 1. *Let K be a field and $a, b \in K$. Then the Hilbert symbol is defined to be*

$$(a,b)_K = \left\{ \begin{array}{ll} 1 & \text{when } B = \left(\frac{a,b}{K}\right) \text{ is split.} \\ -1 & \text{otherwise.} \end{array} \right\}$$

Notice that K can be a global field (i.e., $K = F$) or we could take K to be a local field, $K = F_v$. Notice that B is split if and only if B has a zero divisor. Additionally, we have the following theorem:

Theorem 1. *Let K be a field, $a, b \in K^\times$ and $B = \left(\frac{a,b}{K}\right)$. Further, let $L = K[i]$ where $i^2 = a$. Letting $N_{L/K}(L^\times)$ denote the norm from L/K on$L^\times$, we have that $(a,b)_K = 1$ if and only if $b \in N_{L/K}(L^\times)$.*

This theorem is very handy if we also recall that F_v has a unique unramified quadratic extension, K_v. In the case that B is ramified at v, we then have $B_v \cong \left(\frac{K_v,\pi_v}{F_v}\right)$. So if v divides 2 and if $B_v \cong \left(\frac{a,b}{F_v}\right)$ with $K_v = F_v[i]$, $i^2 = a$, then $(a,b)_v = 1$ if $\mathrm{ord}_v(b)$ even and $(a,b)_v = -1$ if $\mathrm{ord}_v(b)$ is odd.

In the case that F is understood and we are computing the Hilbert symbol locally, we use the following notation: $(a,b)_v := (a,b)_{F_v}$. If v is a complex place, then $B_v = B \otimes \mathbb{C}$ must be split. This is because $\mathbb{C}$ is algebraically closed and thus has no field extensions. Thus for the rest of the paper, when I refer to a place of F, I will mean either a real place or a finite place.
Theorem 2.
Lemma 2. *We have the following equalities:*

1. $(a,b)_K = (b,a)_K = (-ab,b)_K$
2. *For any* $u,t \in K^\times$, $(a,b)_K = (at^2, bu^2)_K$.

These equalities hold as the quaternion algebras in each case are isomorphic.

2 Algorithms and Implementations

The Hilbert symbol is currently implemented in both Magma and Pari. In Magma, the Hilbert symbol was implemented by John Voight using his algorithm from *Identifying the Matrix Ring*. I will outline this algorithm below. Pari uses a similar algorithm. Both algorithms are divided into two cases, odd places and even places.

Definition 2. *We say that v is an odd place if v is archimedean or if v is an odd prime (lies over an odd prime of $\mathbf{Z}$.) Otherwise we say that v is even. In this case v lies over 2.*

The main difference between the Magma and Pari implementations is when computing $(a,b)_v$ and v is an even place.

2.1 Voight's Algorithm

As mentioned above, this algorithm has two cases, odd places and even places. The case where v is an odd place can be simplified to computing what Voight calls the square symbol:

Definition 3. *Take $a \in F$ and v an odd place then the square symbol is defined as follows:*

$$\left\{\frac{a}{v}\right\} = \left\{ \begin{array}{cl} 1 & \textit{if } a \in F_v^{\times 2} \\ -1 & \textit{if } a \notin F_v^{\times 2} \textit{ and } \mathrm{ord}_v(a) \textit{ is even} \\ 0 & \textit{if } a \notin F_v^{\times 2} \textit{ and } \mathrm{ord}_v(a) \textit{ is odd} \end{array} \right\}.$$

With the square symbol, the odd case relies on the following theorem from [Voight]:

Theorem 3. *Let v be an odd place of F and let $a,b \in F_v^\times$. Then $(a,b)_v = 1$ if and only if*

$$\left\{\frac{a}{v}\right\} = 1 \textit{ or } \left\{\frac{b}{v}\right\} = 1 \textit{ or } \left\{\frac{-ab}{v}\right\} = 1$$

$$\textit{or if } \left\{\frac{a}{v}\right\} = \left\{\frac{b}{v}\right\} = -1.$$

Thus by computing $\left\{\frac{a}{v}\right\}$, $\left\{\frac{b}{v}\right\}$, and possibley $\left\{\frac{-ab}{v}\right\} = 1$ we can compute $(a,b)_v$.

Computing the square symbol is straight forward. If v is complex, then $\left\{\frac{a}{v}\right\}$ is trivial. If v is real, $\left\{\frac{a}{v}\right\}$ is 1 or 0 if $a > 0$ or $a < 0$ respectively. If v is nonarchimedean, we can do a little more work and reduce this to Legendre symbol. Suppose $\mathrm{ord}_v(a) = e$. If e is odd then $\left\{\frac{a}{v}\right\} = 0$. If e is even then we define $a_0 = a\pi_v^{-e/2}$ and now $\left\{\frac{a}{v}\right\} = \left(\frac{a_0}{v}\right)$, so we've reduced the case of computing the Legendre symbol.

Now for the even case. Let v be an even place, which will be denoted by the prime $\mathfrak{p}$, and $B_\mathfrak{p} = \left(\frac{a,b}{F_\mathfrak{p}}\right)$ Throughout the even case it is useful to remember that the Hilbert symbol computes whether $B_\mathfrak{p}$ is ramified or split. We know that $F_\mathfrak{p}$ has a unique unramified quadratic extension $K_\mathfrak{p}$. We also know that in the split case $B_\mathfrak{p} = M_2(F_\mathfrak{p})$ thus has a zero divisor. So our goal in the even case is to either:

- find $K_\mathfrak{p} = F_\mathfrak{p}[i']$ for some $i' \in B_\mathfrak{p}$ with $(i')^2 = a'$ and compute $\mathrm{ord}_\mathfrak{p}(b')$
- or to find a zero divisor.

Algorithm for even places: Let $B = \left(\frac{a,b}{F}\right)$, $a, b \in F^\times$, $\mathfrak{p}$ be an even prime of F, and $e = \mathrm{ord}_\mathfrak{p}(2)$. This algorithm returns $(a,b)_\mathfrak{p}$.

1. Multiply a and b by squares in $F^\times$ so that $a, b \in \mathcal{O}_F$.
2. Compute $y, z, w \in \mathcal{O}_F$ so that $1 - ay^2 - bz^2 + abw^2 \equiv 0 (\mathrm{mod}\ \mathfrak{p}^{2e})$. Take $i' = \frac{1+yi+zi+wij}{2}$ and let $p(t) = t^2 - \mathrm{trd}(i')t + \mathrm{nrd}(i')$ be the minimal polynomial of i' in $\mathcal{O}_F$. Notice that $\mathrm{nrd}(i') = 1 - y^2 - z^2 - w^2 \equiv 0 (\mathrm{mod}\ \mathfrak{p}^{2e})$, so we've constructed a probable zero divisor in $F_\mathfrak{p}$.
3. If p has a solution mod v then by Hensel's lemma we can lift this to a root in $\mathcal{O}_{F,\mathfrak{p}}$ and we've found a zero divisor, i'. Thus return 1.
4. Otherwise, we can change basis by taking $j' = (zb)i - (ya)j$ and $b' = (j')^2$ (so that $i'j' = -j'i'$). As p has no roots in $F_\mathfrak{p}$, by adjoining the root i' of p to $F_\mathfrak{p}$ we get the unique unramified quadratic extenion $K_\mathfrak{p} = F_\mathfrak{p}(i')$. Thus if $\mathrm{ord}_\mathfrak{p}(b')$ is even, return 1 and otherwise, return -1.

To use this algorithm we must be able to compute y, z, w as above. Up to this point, Sage has all the machinery to compute Hilbert symbols natively. To compute the y, z, w in an intelligent manner (i.e., not just looping through all choices), Voight uses a Hensel-type lift which requires working in residue rings, $\mathcal{O}_F/\mathfrak{p}^n$ for some integer n of size up to $2e$. Sage does not yet have general residue rings implemented. We start with a, b mulitplied by elements in $F^{\times 2}$ so that a, b are square free. Thus we have the following cases for their valuations:

1. $\mathrm{ord}_\mathfrak{p}(a) = 0$ and $\mathrm{ord}_\mathfrak{p}(b) = 1$
2. $\mathrm{ord}_\mathfrak{p}(a) = \mathrm{ord}_\mathfrak{p}(b) = 0$

Notice that if $\mathrm{ord}_{\mathfrak{p}}(a) = \mathrm{ord}_{\mathfrak{p}}(b) = 1$, then $-ab$ is not square free, so we can reduce to one of the previous cases by possibly replacing a or b with $-ab$.

In the following algorithms, when we write $\sqrt{u}$, we mean that for $u \in (\mathcal{O}_F/\mathfrak{p}^{2e})^\times$ take any lift of $\sqrt{u} \in (\mathcal{O}_F/\mathfrak{p})^\times$ to $\mathcal{O}_F/\mathfrak{p}^{2e}$.

Case 1: $\mathrm{ord}_{\mathfrak{p}}(a) = 0$ and $\mathrm{ord}_{\mathfrak{p}}(b) = 1$

This algorithm outputs $y, z \in \mathcal{O}_F/\mathfrak{p}^{2e}$ such that

$$1 - ay^2 - bz^2 \equiv 0 (\mathrm{mod}\ p^{2e}).$$

1. Initialize $y = 1/\sqrt{a}$ and $z = 0$.
2. Define $N := 1 - ay^2 - bz^2 \in \mathcal{O}_F/4\mathcal{O}_F$ and let $t := \mathrm{ord}_{\mathfrak{p}}(N)$. If $t \geq 2e$, go to step 3. Otherwise, if t is even, replace y with

   $$y = y + \sqrt{\frac{N}{a\pi^t}}\pi^{t/2}$$

   and if t is odd, replace z with

   $$z = z + \sqrt{\frac{N}{b\pi^{t-1}}}\pi^{\lfloor t/2 \rfloor}$$

   Return to step 2.
3. Return y, z.

Proof: See Voight.

Case 2: $\mathrm{ord}_{\mathfrak{p}}(a) = \mathrm{ord}_{\mathfrak{p}}(b) = 0$

This algorithm outputs $y, z, w \in \mathcal{O}_F/\mathfrak{p}^{2e}$ such that

$$1 - ay^2 - bz^2 + abw^2 \equiv 0 (\mathrm{mod}\ p^{2e}).$$

1. If $a, b \in (\mathcal{O}_f/\mathfrak{p}^e)^{\times 2}$ find a_0 and b_0 such that

   $$(a_0)^2 a \equiv 1 (\mathrm{mod}\ \mathfrak{p}^e) \text{ and } (b_0)^2 b \equiv 1 (\mathrm{mod}\ \mathfrak{p}^e).$$

   Return $y = a_0, z = b_0, w = a_0 b_0$.
2. Swap a, b so that $a \notin (\mathcal{O}_F/\mathfrak{p}^e)^\times$. Take t to be the largest integer such that $a \in (\mathcal{O}_F/\mathfrak{p}^t)^{\times 2}$ but $a \notin (\mathcal{O}_F/\mathfrak{p}^{t+1})^{\times 2}$. Now lift, meaning, find a_0 and a_t in $\mathcal{O}_F$ so that $a = a_0^2 + \pi^t a_t$. We have now reduced to Case 1. Input $a, -\pi a_t/b$ into Case 1 to get y_1, z_1. Return

   $$y = \frac{1}{a_0}, z = \frac{\pi^{\lfloor t/2 \rfloor}}{a_0 z_1}, w = \frac{y_1 \pi^{\lfloor t/2 \rfloor}}{a_0 z_1}.$$

Proof: See Voight.

So the only problem with implementing this algorithm in Sage is lifting from $(\mathcal{O}_F/\mathfrak{p})^\times$ to $\mathcal{O}_F/\mathfrak{p}^{2e}$.

2.2 Pari's Imlementation

For the case where v is an odd place, Pari's implementation seems to be the same as Voights. For the even place case Pari calls a function called

```
nf_hyperell_locally_soluble
```

which:

```
/* = 1 if equation y^2 = z^deg(T) * T(x/z) has a pr-adic rational solution
 * (possibly (1,y,0) = oo), 0 otherwise.
 * coeffs of T are algebraic integers in nf */
```

and this and the full source code can be found at:

http://pari.math.u-bordeaux.fr/cgi-bin/viewcvs.cgi/trunk/src/basemath/buch4.c?view=markup&root=pari&pathrev=12778

3 Code/patch in sage

The trac ticket for this project is number 9334. To wrap Pari's Hilbert symbol in Sage the following code works, but is slow:

```
def pari_hs(K,a,b,P):
    nK = gp(K)
    na = gp(a)
    nb = gp(b)
    hnfP = nK.idealhnf(gp(P))
    mP = gp.idealfactor(nK,hnfP)
    np = mP[1,1]
    return nK.nfhilbert(na,nb,np)
```

and to compute the Hilbert symbol in Magma the analogous code is:

```
>P<x>:=PolynomialRing(IntegerRing());
>f:=x^5-23;
>K<a>:=NumberField(f);
>b:=-a+5;
>g:=-7*a^4+13*a^3-13*a^2-2*a+50;
>OK:=RingOfIntegers(K);
>Q:=ideal<OK|g>;
>HilbertSymbol(a,b,Q);
>1
```

4 References

[Sage Days 22 code] http://wiki.sagemath.org/days22/dokchitser?action=AttachFile&do=view&target=root_number.sage

[Voight] John Voight, *Identifying the Matrix Ring*, submitted. http://www.cems.uvm.edu/~voight/articles/quatalgs-040110.pdf