ノルム 1 トーラスとハッセノルム原理：GAP の使い方を中心に

山崎愛一 (京都大学)
Aiichi Yamasaki (Kyoto University)

1 はじめに

k を体，$\overline{k}$ を k の分離閉包，$\mathcal{G} = \mathrm{Gal}(\overline{k}/k)$ を k の絶対 Galois 群とする．このとき，k 上の代数的トーラス T と，$\mathcal{G}$ 加群としての T の指標加群 $X(T) = \mathrm{Hom}(T, \mathbb{G}_m)$ の間の双対性が知られている (小野 [Ono61])．特に，k が代数的数体で T がノルム 1 トーラスの場合は, T の弱近似定理の成否を表す量として $A(T)$, T のハッセノルム原理の成否を表す量として Shafarevich-Tate 群 Ш(T) があり，完全系列

$$0 \to A(T) \to H^1(k, \mathrm{Pic}\,\overline{X})^\vee \to Ш(T) \to 0$$

が知られている (Voskresenskii [Vos69])．ただし，$\mathrm{Pic}\,\overline{X}$ は T の滑らかなコンパクト化 X の Picard 群，$\vee$ は Pontryagin 双対を表す.

私は新潟大学の星明考さん，金井和貴さんとの共同研究で k が代数的数体で K/k が 15 次以下のとき，K/k のノルム 1 トーラス $T = R^{(1)}_{K/k}(\mathbb{G}_m)$ に対して Ш$(T) = 0$ が成り立つ必要十分条件を決定した ([HKY], [HKY2])．K/k が素数次ならば $H^1(k, \mathrm{Pic}\,\overline{X})^\vee = 0$ が成立することが言える (Colliot-Thélène, Sansuc [CTS87, Proposition 9.1] 参照)．K/k が Galois 拡大の場合は Tate による公式 (Tate [Tat67, 198 ページ]) が知られているが，非 Galois 拡大の場合は K/k の次数が 8 以上の合成数のときは完全には知られていなかった．それを，理論的にも技術的にも工夫されたアルゴリズムを用いることにより K/k が 15 次の場合まですべて解決した．この結果の応用例として玉河数 $\tau(T)$ を計算することができる (小野 [Ono65])．詳しい内容は論文を見ていただくことにして，この報告集では論文中で GAP を使っている部分の解説と GAP の実際の使用法をインストールの仕方を含めて紹介したい.

今回開発したアルゴリズムについては，数式処理ソフト GAP のプログラムを私のホームページ

`https://www.math.kyoto-u.ac.jp/~yamasaki/Algorithm/Norm1ToriHNP`

で公開しているので，誰でもダウンロードして計算することができる.

謝辞． 講演の機会を与えてくださった小林真一さん（九州大学)，加塩朋和さん（東京理科大学)，星明考さん（新潟大学）に感謝いたします.

本研究は JSPS 科研費 20K03511 の助成を受けたものです.

2 準備

この節ではよく知られた定義を復習し，関連した定理を述べる.

定義 1. k を代数的数体，T を k 上の代数的トーラスとする．T の k 有理点全体の集合を $T(k)$ と表す．k の

素点全体の集合を $V(k)$, $v \in V(k)$ での k の完備化を k_v で表す．$T(k)$ を $\prod_{v \in V_k} T(k_v)$ に対角的に埋め込むとき，$T(k)$ の閉包を $\overline{T(k)}$ で表す．

$$A(T) = \left(\prod_{v \in V_k} T(k_v) \right) \Big/ \overline{T(k)}$$

を T の弱近似の核と言う．$A(T) = 0$ のとき弱近似定理が成立すると言う．

定義 2. k を代数的数体，K を k の有限次拡大とする．K のイデール群を $\mathbb{A}_K^\times$ で表す．

$$(N_{K/k}(\mathbb{A}_K^\times) \cap k^\times)/N_{K/k}(K^\times) = 1$$

が成り立つとき，K/k はハッセノルム原理を満たすという．

定義 3. 代数的数体 k 上の代数的トーラス T に対して

$$Ш(T) = \mathrm{Ker} \left\{ H^1(k, T) \xrightarrow{\mathrm{res}} \bigoplus_{v \in V_k} H^1(k_v, T) \right\}$$

で Shafarevich-Tate 群 $Ш(T)$ を定義する．

定理 1 (小野 [Ono63, 70 ページ]). k を代数的数体，K を k の有限次拡大とする．このとき

$$Ш(R^{(1)}_{K/k}(\mathbb{G}_m)) \simeq (N_{K/k}(\mathbb{A}_K^\times) \cap k^\times)/N_{K/k}(K^\times)$$

が成り立つ．ただし $R^{(1)}_{K/k}(\mathbb{G}_m)$ は K/k のノルム 1 トーラスとする．特に，$Ш(R^{(1)}_{K/k}(\mathbb{G}_m)) = 0$ は K/k のハッセノルム原理が成り立つ必要十分条件である．

定義 4 (Drakokhrust，Platonov [PD85a, 350 ページ], [DP87, 300 ページ]). k を代数的数体，$L \supset K \supset k$ を有限次拡大の列で L は k 上 Galois 拡大とする．

$$\mathrm{Obs}(K/k) = (N_{K/k}(\mathbb{A}_K^\times) \cap k^\times)/N_{K/k}(K^\times)$$

を K/k のハッセノルム原理の total obstruction,

$$\mathrm{Obs}_1(L/K/k) = \left(N_{K/k}(\mathbb{A}_K^\times) \cap k^\times\right) / \left((N_{L/k}(\mathbb{A}_L^\times) \cap k^\times) N_{K/k}(K^\times)\right)$$

を有限次拡大の列 $L \supset K \supset k$ に対応する K/k のハッセノルム原理の first obstruction と呼ぶ．

$\mathrm{Obs}(K/k) = 1$ は K/k のハッセノルム原理が成り立つための必要十分条件になっている．また $\mathrm{Obs}_1(L/K/k) = \mathrm{Obs}(K/k)/(N_{L/k}(\mathbb{A}_L^\times) \cap k^\times)$ である．一般に $\mathrm{Obs}(K/k)$ の計算は難しいが，$\mathrm{Obs}_1(L/K/k)$ の方は次のように計算できることが知られている．

定理 2 (Drakokhrust, Platonov [PD85a, 350 ページ], [PD85b, 789–790 ページ], [DP87, Theorem 1]). k を代数的数体，$L \supset K \supset k$ を有限次拡大の列で L は k 上 Galois 拡大とする．$G = \mathrm{Gal}(L/k)$, $H = \mathrm{Gal}(L/K)$ とするとき，

$$\mathrm{Obs}_1(L/K/k) \simeq \mathrm{Ker}\, \psi_1 / \varphi_1(\mathrm{Ker}\, \psi_2)$$

が成り立つ．可換図式

$$\begin{array}{ccc} H/[H,H] & \xrightarrow{\psi_1} & G/[G,G] \\ \uparrow{\scriptstyle \varphi_1} & & \uparrow{\scriptstyle \varphi_2} \\ \bigoplus_{v\in V_k}\left(\bigoplus_{w|v} H_w/[H_w,H_w]\right) & \xrightarrow{\psi_2} & \bigoplus_{v\in V_k} G_v/[G_v,G_v] \end{array}$$

において，ψ_1, φ_1, φ_2 は埋め込み $H \subset G$, $H_w \subset H$, $G_v \subset G$ によりそれぞれ定義される．$h \in H_w = H \cap x^{-1}hx[G_v, G_v]$ $(x \in G)$ に対し

$$\psi_2(h[H_w, H_w]) = x^{-1}hx[G_v, G_v]$$

となるように ψ_2 を定める．

$v \in V_k$ に対して ψ_2 を $\bigoplus_{w|v} H_w/[H_w, H_w]$ に制限したものを ψ_2^v と表すことにする．また，ψ_2 を L/k の不分岐素点（および分岐素点）に制限したものをそれぞれ ψ_2^{nr} (および ψ_2^{r}) と表すことにする．

定理 2 を使って，first obstruction の計算を GAP で計算できるようにした．ただし，first obstruction と total obstruction が一致するとは限らない．そこで，L の代わりに L を含む k の有限次 Galois 拡大 $\overline{L}$ で $\mathrm{Obs}(K/k) = \mathrm{Obs}_1(\overline{L}/K/k)$ が成り立つものを見つけたい．次の定理が知られている．

定理 3 (Drakokhrust, Platonov [DP87, Theorem 3, Corollary 1])**.** *k, $L \supset K \supset k$, G, H を定理 2 の通りとし，$H_i \leq G_i \leq G$ $(1 \leq i \leq m)$, $H_i \leq H \cap G_i$, $k_i = L^{G_i}$, $K_i = L^{H_i}$ とする．$\mathrm{Obs}(K_i/k_i) = 1$ がすべての $1 \leq i \leq m$ に対して成り立ち，*

$$\bigoplus_{i=1}^{m} \widehat{H}^{-3}(G_i, \mathbb{Z}) \xrightarrow{\text{cores}} \widehat{H}^{-3}(G, \mathbb{Z})$$

が全射ならば，$\mathrm{Obs}(K/k) = \mathrm{Obs}_1(L/K/k)$ が成り立つ．特に K/k の次数が平方因子を持たなければ $\mathrm{Obs}(K/k) = \mathrm{Obs}_1(L/K/k)$ が成り立つ．

定理 4 (Drakokhrust [Dra89, Theorem 1], $\widetilde{L}$ の存在は Opolka [Opo80, Satz 3])**.** *k, $L \supset K \supset k$, G, H を定理 2 の通りとし，$\widetilde{L} \supset L \supset k$ を Galois 拡大の列，$\widetilde{G} = \mathrm{Gal}(\widetilde{L}/k)$, $\widetilde{H} = \mathrm{Gal}(\widetilde{L}/K)$ とする．$\widetilde{G}$ は G の中心拡大で $1 \to A \to \widetilde{G} \to G \to 1$, $A \cap [\widetilde{G}, \widetilde{G}] \simeq M(G) = H^2(G, \mathbb{C}^\times)$; G の Schur multiplier，とする（これは inflation $M(G) \to M(\widetilde{G})$ が零写像になることと同値 (Beyl, Tappe [BT82, Proposition 2.13, 85 ページ])）．このとき $\mathrm{Obs}(K/k) = \mathrm{Obs}_1(\widetilde{L}/K/k)$ が成り立つ．特に $\widetilde{G}$ が G の Schur cover のとき（すなわち $A \simeq M(G)$ のとき），$\mathrm{Obs}(K/k) = \mathrm{Obs}_1(\widetilde{L}/K/k)$ が成り立つ．*

定理 4 より $\mathrm{Obs}(K/k) = \mathrm{Obs}_1(\widetilde{L}/K/k)$ が成立する $\widetilde{L}$ の存在が保証されている．しかし $\widetilde{L}/k$ の次数が高いと計算機上で $\mathrm{Obs}_1(\widetilde{L}/K/k)$ を計算するのが困難になる．その場合でも，定理 3 をみたす $\overline{L}/k$ のうち次数が低いものを GAP で探すための関数 SchurCoverG(G), MinimalStemExtensions(G) 等を作った．

3　GAP のインストール

以下では，Windows PC に GAP をインストールするものとします．

まず，PC のメインメモリが 8GB 以上あることを確認してください．「Windows+x」を押して WinX メニューを開いてから「Y」を押すと詳細情報が開くので，「実装 RAM」が実装されているメインメモリの容量です．メインメモリが 2GB や 4GB しかない場合は，増設して 8GB 以上（できれば 16GB 以上）にしてください．その際 DDR3, DDR4, DDR5 や DIMM, SO–DIMM 等規格の違いに気を付けてください．メモリは多ければ多いほど良いです．

次に GAP をインストールするストレージ（通常は C ドライブ）の空き容量が足りているか確認してください．「Windows+x」を押して WinX メニューを開いてから「Y」を押して詳細情報を開き，さらに「記憶域」をマウスで左クリックすると確認できます．空き容量が 20GB を切っているようであれば，ストレージを増設するか不要なファイルを消す等して空き容量を増やしてください．

メインメモリとストレージを混同しないようにしましょう．

GAP をインストールするためにはまず，WSL2 をインストールして Linux 環境を構築する必要があります．ここでは Linux 環境は Ubuntu 20.04 をインストールするものとします．具体的な操作方法は

```
https://docs.microsoft.com/ja-jp/windows/wsl/install
```

を参照してください．管理者権限で PowerShell を開くには「Windows+x」を押して WinX メニューを開いてから「A」を押してください．

Ubuntu を起動するためには「Windows+x」を押して WinX メニューを開いてから「I」を押して PowerShell を開いた後で，「ubuntu」と入力して「Enter」キーを押してください．それで Ubuntu が起動しなければ「ubuntu2004」と入力してください．それでも Ubuntu が起動しなければ，WSL2 か Ubuntu のインストールがうまくいっていないものと考えられます．

WSL2 と Ubuntu のインストールが終わったら Ubuntu から

```
sudo apt update
```

と入力して更新ファイルを確認します．パスワードの入力を求められたらログインパスワードを入力してください．更新ファイルが見つかれば

```
sudo apt upgrade
```

と入力して更新してください．もし

```
Do youwant to continue? [Y/n]
```

と表示されたら「Y」キーと「Enter」キーを押して続行してください．

GAP のダウンロードページ

```
https://www.gap-system.org/Releases/
```

から GAP をダウンロードしてください．ダウンロードするファイルは GAP のバージョンにより gap–4.9.3.tar.gz, gap–4.11.1.tar.gz 等となります．以後 gap–4.X.Y.tar.gz と書くことにします．次にダウンロードしたファイルを Ubuntu で展開する必要があります．まず，Ubuntu から順に

```
cd /mnt/c
mkdir tmp
```

```
cd tmp
```

と入力してください．これで C ドライブに tmp フォルダが出来たはずです．次に「Windows+x」を押して WinX メニューを開いてから「E」を押してファイルエクスプローラーを開いてください．ダウンロードフォルダを開いて，先ほどダウンロードした gap–4.X.Y.tar.gz があることを確認してください．もう一回「Windows+x」「E」を押してファイルエクスプローラーを開いて，今度は C:¥tmp フォルダを開いてください．マウスでドラッグ＆ドロップでダウンロードフォルダから C:¥tmp フォルダへ gap–4.X.Y.tar.gz をコピーしてください．Ubuntu から

```
ls
```

と入力してファイルが正しくコピーできたことを確認してください．次に Ubuntu から

```
sudo apt-get install build-essential
```

と入力して開発環境をインストールしてください．さらに

```
sudo apt-get install autoconf libtool libgmp-dev libreadline-dev zlib1g-dev
```

と入力して GAP のインストールに必要なものをインストールしてください．必要に応じて polymake, asymptote, imagemagick, singular, pari-gp 等をインストールしても良いでしょう．例えば polymake をインストールするには

```
sudo apt install polymake
```

と入力します．

GAP のインストール準備が出来たのでいよいよ GAP をインストールします．Ubuntu から

```
cd
tar zxvf /mnt/c/tmp/gap-4.X.Y.tar.gz
```

と入力してダウンロードしたファイルを自分のホームフォルダに展開します．次に，順に

```
cd gap-4.X.Y
./configure
make
```

と入力して GAP をコンパイルします．4.X.Y のところは GAP のバージョンごとに適当に読み替えてください．

```
cd pkg
../bin/BuildPackages.sh
```

と入力してパッケージをインストールしてください．最後に

```
sudo ln -s ~/gap-4.X.Y/bin/gap.sh /usr/local/bin/gap-4.X.Y
```

と入力してください．これで，

```
gap-4.X.Y
```

と入力するだけで GAP 4.X.Y が起動するはずです．

将来 GAP 新しいバージョンが登場しても従来のバージョンを消す必要はありません．複数のバージョンをインストールしておくと，必要に応じて使い分けることができます．

4 Norm1ToriHNP のインストール

私のホームページ

```
https://www.math.kyoto-u.ac.jp/~yamasaki/Algorithm/Norm1ToriHNP/
```

から Norm1ToriHNP.zip をダウンロードしてください．GAP のインストールのときと同様にして C:¥tmp フォルダへ Norm1ToriHNP.zip をコピーしてください．Ubuntu から順に

```
mkdir Norm1ToriHNP
cd Norm1ToriHNP
unzip /mnt/c/tmp/Norm1ToriHNP.zip
```

と入力して Norm1ToriHNP.zip を展開してください．もし unzip がインストールされていなくて展開できないときは

```
sudo apt install unaip
```

で unzip をインストールしてください．

各関数の詳しい説明については

```
https://www.math.kyoto-u.ac.jp/~yamasaki/Algorithm/Norm1ToriHNP/HNP.html
```

を見てください．

5 GAP および Norm1ToriHNP の使い方

Linux 上で

```
cd ~/Norm1ToriHNP
gap-4.X.Y
```

と入力して GAP を立ち上げます．次に

```
Read("HNP.gap");
```

と入力して Norm1ToriHNP を読み込みます．このとき画面上には

```
gap> Read("HNP.gap");
```

と表示されるはずです．

K/k が $V_4 \simeq C_2 \times C_2$ Galois 拡大のときを例に使い方を見ていくことにしましょう．まず，S_4 の可移部分群 V_4 を G とおきます．S_2 から S_{37} までといくつかのより次数の高い対称群とについては可移部分群の共役類がすべて分類されていて，S_{30} までは GAP で呼び出せるようになっています．V_4 は S_4 の 2 番目の可移部分群で

```
TransitiveGroup(4,2);
```

で呼び出せます．GAP で変数 `G` に V_4 を代入するのは

```
G:=TransitiveGroup(4,2);
```

と入力すればできます．画面上では

```
gap> G:=TransitiveGroup(4,2);
E(4) = 2[x]2
```

と表示されるはずです．

以下では，GAP の入出力を画面に表示される形式で表すことにします．先頭に

```
gap>
```

がある行は GAP に入力している行，ない行は GAP からの出力行です．入力行では `gap>`よりも右が入力です．

例えば GAP の関数 TransitiveGroup の使い方を調べたいときは

```
gap> ? TransitiveGroup
```

と入力すれば説明が表示されます．

GAP はオープンソースなのでコードを見ることができます．例えば

```
gap> Print(TransitiveGroup);
```

とすれば TransitiveGroup のコードが表示されます．

また GAP では入力を補完する方法が 2 種類あります．まず「Tab」キーを使った入力補完について説明します．例えば `TransitiveGroup` と入力したいときに最初の 6 文字だけ入力して

```
gap> Transi
```

となっているときに，「Tab」を押すと自動的にあと 3 文字だけ入力されて

```
gap> Transitiv
```

となります．これは `Transi` で始まる GAP の関数はすべて，続く 3 文字が `tiv` になっているので「Tab」を押すと自動的に入力してくれるのです．もう一回「Tab」を押すと

```
gap> Transitiv
TransitiveClosureBinaryRelation  TransitiveIdentification
TransitiveGroup                  Transitivity
```

```
TransitiveGroupsAvailable
gap> Transitiv
```

となり補完候補を教えてくれます．出力結果を参考にしながら残りを入力するかまた何文字か入力して再度「Tab」を押せば `TransitiveGroup` が入力できます．この例における具体的な補完候補は GAP のバージョンによって異なるかもしれません．

GAP で入力補完をするもう一つの方法はカーソルキーを使う方法です．例えば

```
gap> Transi
```

となっているときに，「↑」を押すと，入力の履歴の中から `Transi` で始まる行が順にさかのぼって表示されます．「↑」「↓」を使って目的の行を選ぶことができます．「Enter」を押せば選んだ行をそのまま使うことができます．選んだ行を編集することもできます．その際，「←」「→」を使って一文字ずつカーソルを移動させる以外に，「Ctrl+←」,「Ctrl+→」で直接行頭, 行末に飛ぶこともできます．

H として，G の一点固定群を取ります．

```
gap> H:=Stabilizer(G,1);
Group(())
```

定理 2 の $\mathrm{Obs}_1(L/K/k) \simeq \mathrm{Ker}\,\psi_1/\varphi_1(\mathrm{Ker}\,\psi_2)$ における分子 $=\mathrm{Ker}\,\psi_1$ を計算する GAP の関数は FirstObstructionN(G,H) です．ただし H が G の一点固定群のときは省略することができます．より詳しい説明は

https://www.math.kyoto-u.ac.jp/~yamasaki/Algorithm/Norm1ToriHNP/HNP.html

を見てください．

```
gap> FirstObstructionN(G).ker;
[ [  ], [ [  ], [  ] ] ]
```

一番左の要素を見ることで $\mathrm{Ker}\,\psi_1 = 0$ となることが分かります．

G の Schur Cover $\widetilde{G}$ を取って `tG` とおきます．また $\widetilde{G}$ における H の逆像 $\widetilde{H}$ を `tH` とおきます．

```
gap> ScG:=SchurCoverG(G);
rec( SchurCover := Group([ (2,3), (1,2)(3,4) ]),
  epi := [ (2,3), (1,2)(3,4) ] -> [ (1,2)(3,4), (1,4)(2,3) ], Tid := [ 4, 3 ] )
gap> tG:=ScG.SchurCover;
Group([ (2,3), (1,2)(3,4) ])
gap> tH:=PreImage(ScG.epi,H);
Group([ (1,4)(2,3) ])
```

G, H の代わりに $\widetilde{G}$, $\widetilde{H}$ に対して分子 $=\mathrm{Ker}\,\psi_1$，分母の不分岐成分 $=\varphi_1(\mathrm{Ker}\,\psi_2^{\mathrm{nr}})$ を計算します．

```
gap> FirstObstructionN(tG,tH).ker;
[ [ 2 ], [ [ 2 ], [ [ 1 ] ] ] ]
```

```
gap> FirstObstructionDnr(tG,tH).Dnr;
[ [  ], [ [ 2 ], [  ] ] ]
```

一番左の要素を見ることにより分子 $\simeq \mathbb{Z}/2\mathbb{Z}$, 分母の不分岐成分 $\simeq 0$ となることが分かります.

$\widetilde{L}/k$ の分岐素点 v に対し，G_v は G の部分群，G_v の Schur cover による逆像 $\widetilde{G}_v$ は $\widetilde{G}$ の部分群となります．$\widetilde{G}$ の各部分群ごとに $\widetilde{G}_v$ がそれになったときに分母の v 成分 $= \varphi_1(\mathrm{Ker}\,\psi_2^v)$ を計算します.

```
gap> tGs:=AllSubgroups(tG);
[ Group(()), Group([ (2,3) ]), Group([ (1,4) ]), Group([ (1,4)(2,3) ]),
  Group([ (1,2)(3,4) ]), Group([ (1,3)(2,4) ]), Group([ (1,4), (2,3) ]),
  Group([ (1,2)(3,4), (1,4)(2,3) ]), Group([ (1,3,4,2), (1,4)(2,3) ]),
  Group([ (1,4), (2,3), (1,2)(3,4) ]) ]
gap> List(tGs,x->FirstObstructionDr(tG,x,tH).Dr);
[ [ [  ], [ [ 2 ], [  ] ] ],
  [ [  ], [ [ 2 ], [  ] ] ],
  [ [  ], [ [ 2 ], [  ] ] ],
  [ [  ], [ [ 2 ], [  ] ] ],
  [ [  ], [ [ 2 ], [  ] ] ],
  [ [  ], [ [ 2 ], [  ] ] ],
  [ [  ], [ [ 2 ], [  ] ] ],
  [ [  ], [ [ 2 ], [  ] ] ],
  [ [  ], [ [ 2 ], [  ] ] ],
  [ [ 2 ], [ [ 2 ], [ [ 1 ] ] ] ] ]
gap> List(tGs,x->StructureDescription(Image(ScG.epi,x)));
[ "1", "C2", "C2", "1", "C2", "C2", "C2", "C2", "C2", "C2 x C2" ]
```

一番左の要素を見ると最後以外は [ ] なので分母の v 成分は 0 と同型が分かります．最後は $\widetilde{G}_v = \widetilde{G}$ に対応していて分母の v 成分は $\mathbb{Z}/2\mathbb{Z}$ と同型が分かります．これより $G_v = G$ となる v が存在することが K/k がハッセノルム原理成立の必要十分条件となり，知られている結果と一致します ([HKY, Section 1] 参照).

GAP の計算結果をログファイルに残したい場合は, 例えば

```
gap> LogTo("gap.log");
```

とすれば良いです．ログファイルへの書き込みをやめるときは

```
gap> LogTo();
```

とします.

GAP を終了するには

```
gap> quit;
```

とします．エラーで GAP が計算途中で止まったときにエラーから抜けるときも

```
gap> quit;
```

とします.

ログファイルを Windows から開くために C:¥tmp フォルダにコピーするには Linux から

```
cp gap.log /mnt/c/tmp
```

等と入力してください.

参考文献

[BT82] F. R. Beyl, J. Tappe, *Group extensions, representations, and the Schur multiplicator*, Lecture Notes in Mathematics, 958. Springer-Verlag, Berlin-New York, 1982.

[CTS87] J.-L. Colliot-Thélène, J.-J. Sansuc, *Principal homogeneous spaces under flasque tori: Applications*, J. Algebra **106** (1987) 148–205.

[Dra89] Yu. A. Drakokhrust, *On the complete obstruction to the Hasse principle*, (Russian) Dokl. Akad. Nauk BSSR **30** (1986) 5–8; translation in Amer. Math. Soc. Transl. (2) **143** (1989) 29–34.

[DP87] Yu. A. Drakokhrust, V. P. Platonov, *The Hasse norm principle for algebraic number fields*, (Russian) Izv. Akad. Nauk SSSR Ser. Mat. **50** (1986) 946–968; translation in Math. USSR-Izv. **29** (1987) 299–322.

[HKY] A. Hoshi, K. Kanai, A. Yamsaki, *Norm one tori and Hasse norm principle*, arXiv:1910.01469, to appear in Math. Comp.

[HKY2] A. Hoshi, K. Kanai, A. Yamsaki, *Norm one tori and Hasse norm principle, II: degree 12 case*, arXiv:2003.08253.

[Kap87] G. Karpilovsky, *The Schur multiplier*, London Mathematical Society Monographs. New Series, 2. The Clarendon Press, Oxford University Press, New York, 1987.

[Ono61] T. Ono, *Arithmetic of algebraic tori*, Ann. of Math. (2) **74** (1961) 101–139.

[Ono63] T. Ono, *On the Tamagawa number of algebraic tori*, Ann. of Math. (2) **78** (1963) 47–73.

[Ono65] T. Ono, *On the relative theory of Tamagawa numbers*, Ann. of Math. (2) **82** (1965) 88–111.

[Opo80] H. Opolka, *Zur Auflösung zahlentheoretischer Knoten*, Math. Z. **173** (1980) 95–103.

[PD85a] V. P. Platonov, Yu. A. Drakokhrust, *On the Hasse principle for algebraic number fields*, (Russian) Dokl. Akad. Nauk SSSR **281** (1985) 793–797; translation in Soviet Math. Dokl. **31** (1985) 349–353.

[PD85b] V. P. Platonov, Yu. A. Drakokhrust, *The Hasse norm principle for primary extensions of algebraic number fields*, (Russian) Dokl. Akad. Nauk SSSR **285** (1985) 812–815; translation in Soviet Math. Dokl. **32** (1985) 789–792.

[Rob96] D. J. S. Robinson, *A course in the theory of groups*, Second edition. Graduate Texts in Mathematics, 80. Springer-Verlag, New York, 1996.

[Tat67] J. Tate, *Global class field theory*, Algebraic Number Theory, Proceedings of an instructional conference organized by the London Mathematical Society (a NATO Advanced Study Institute) with the support of the International Mathematical Union, Edited by J. W. S. Cassels and A. Fröhlich, 162–203, Academic Press, London; Thompson Book Co., Inc., Washington, D.C. 1967.

[Vos69] V. E. Voskresenskii, *The birational equivalence of linear algebraic groups*, (Russian) Dokl. Akad.

Nauk SSSR **188** (1969) 978–981; erratum, ibid. 191 1969 nos., 1, 2, 3, vii; translation in Soviet Math. Dokl. **10** (1969) 1212–1215.