

Tate のアルゴリズム

Tate の方法 [Ta] によって, 楕円曲線を Weierstrass 方程式で与えた時, 前節の 小平-Néron の定理 に依る Kodaira 型を実際に求める. この節では, 次の様な記号を用いる.

$R : \mathfrak{p} = (\pi)$ を極大イデアルに持つ離散付値環, 商体を K , 剰余体 k は標数 p の完全体.

E/K : Weierstrass 方程式 $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ で与えられる楕円曲線.

\mathcal{C}/R : E/K の極小固有正則モデル.

\mathcal{E}/R : E/K の Néron モデル.

$\tilde{\mathcal{C}}/k$: \mathcal{C} の special ファイバー.

$\tilde{\mathcal{E}}/k$: \mathcal{E} の special ファイバー.

\mathcal{E}^0/R : \mathcal{E} の 恒等成分.

$\tilde{\mathcal{E}}^0/k$: 代数群 $\tilde{\mathcal{E}}/k$ の 恒等成分.

定義 1. E/K の Weierstrass 方程式 $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ ($a_i \in R$) が **極小 (minimal)** であるとは, E の R 係数 Weierstrass 方程式の中で, 判別式 Δ の付値が最小となるものを言う.

定理 2 (Tate のアルゴリズム). 楕円曲線 E/K が Weierstrass 方程式 $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ で与えられているとする. この時, 次に述べる手順により, \mathcal{C} の special ファイバー $\tilde{\mathcal{C}}$ の \bar{k} 上の Kodaira 型, \bar{k} 上定義された special ファイバーの既約成分の数 m , E/K の導手の指数 f , それから k 上定義された special ファイバー $\tilde{\mathcal{E}}$ 上重複度 1 の成分の数 $c = \#\tilde{\mathcal{E}}(k)/\tilde{\mathcal{E}}^0(k)$ を求める. また, 初めに与えられた Weierstrass 方程式は極小である必要はなく, このアルゴリズムにより極小 Weierstrass 方程式も得られる.

適当な変換により $a_i \in R$ としておく. **囲み線** は以後仮定する条件を表している.

Step 1.

$$b_2 = a_1^2 + 4a_2, \quad b_4 = 2a_4 + a_1a_3, \quad b_6 = a_3^2 + 4a_6, \quad b_8 = a_1^2a_6 + 4a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2, \\ c_4 = b_2^2 - 24b_4, \quad c_6 = b_2^3 + 36b_2b_4 - 216b_6, \quad \Delta = -b_2^2b_8 - 8b_4^3 - 27b_6^2 + 9b_2b_4b_6$$

を計算しておく. そこで $\pi \nmid \Delta$ ならば Type I_0 , $m = 1$, $f = 0$, $c = 1$, $\tilde{\mathcal{E}}/k$ は楕円曲線.

Step 2. $\pi \mid \Delta$ とする. 座標変換により $\pi \mid a_3, a_4, a_6$ とする. ここで, $\pi \nmid b_2$ ならば Type I_n ($n = \text{ord}(\Delta)$), $m = n$, $f = 1$ であり, k' を $T^2 + a_1T - a_2 = 0$ の k 上の分解体とする.

Step 2a. $k' = k$ ならば $c = \text{ord}(\Delta)$, $\tilde{\mathcal{E}}^0(k) \simeq k^\times$.

Step 2b. $k' \neq k$ ならば $c = \begin{cases} 1, & n: \text{奇数} \\ 2, & n: \text{偶数} \end{cases}$, $\tilde{\mathcal{E}}^0(k) \simeq \{\alpha \in k' \mid N_{k'/k}(\alpha) = 1\}$.

Step 3. $\pi \mid b_2$ とする. これ以降 $\tilde{\mathcal{E}}^0(k) \simeq k^+$ である. ここで $\pi^2 \nmid a_6$ ならば Type II, $m = 1$, $f = \text{ord}(\Delta)$, $c = 1$.

Step 4. $\pi^2 \mid a_6$ とする. これは同時に $\pi^2 \mid b_6, b_8$ を意味する. ここで $\pi^3 \nmid b_8$ ならば Type III, $m = 2$, $f = \text{ord}(\Delta) - 1$, $c = 2$.

Step 5. $\pi^3 \mid b_8$ とする. すると $\pi^2 \mid a_4$. ここで $\pi^3 \nmid b_6$ ならば Type IV, $m = 3$, $f = \text{ord}(\Delta) - 2$. 記号

$$a_{i,j} := \pi^{-j}a_i \text{ を用いて } k' \text{ を } T^2 + a_{3,1}T - a_{6,2} = 0 \text{ の } k \text{ 上の分解体とすると, } c = \begin{cases} 3, & k'=k \\ 1, & k' \neq k \end{cases}.$$

Step 6. $\pi^3 \mid b_6$ とする. 適当な座標変換により $\pi \mid a_1, a_2, \pi^2 \mid a_3, a_4, \pi^3 \mid a_6$ とできる. こうして, 多項式 $P(T) = T^3 + a_{2,1}T^2 + a_{4,2}T + a_{6,3}$ を考える. P の判別式は $D = \pi^{-6}(-4a_2^3a_6 + a_2^2a_4^2 - 4a_4^3 - 27a_6^2 + 18a_2a_4a_6)$ である. ここで, $P(T)$ が 3 つの相異なる根を持つ (i.e., $\pi \nmid D$) ならば, Type I_0^* , $m = 5$, $f = \text{ord}(\Delta) - 4$, $c = 1 + \#\{\alpha \in k \mid P(\alpha) = 0\}$.

Step 7. $P(T)$ が単根と重根を持つならば, Type I_n^* , $m = n + 5$, $f = \text{ord}(\Delta) - 4 - n$. ここで c, n は次の手順により得られる.

Subprocedure. 変換により $T = 0$ が重根と成るようにする. すると $\pi^3 \mid a_4, \pi^4 \mid a_6, \pi^2 \nmid a_2$.

Step a. 多項式 $Y^2 + a_{3,2}Y - a_{6,4}$ が \bar{k} で異なる根を持つならば, k' をその分解体として

$n = 1, c = \begin{cases} 4, & k'=k \\ 2, & k' \neq k \end{cases}$. $Y^2 + a_{3,2}Y - a_{6,4}$ が \bar{k} で重根を持つならば, その根を変換により $Y = 0$

として $\pi^3 \mid a_3, \pi^5 \mid a_6$ である.

Step b. 多項式 $a_{2,1}X^2 + a_{4,3}X + a_{6,5}$ が \bar{k} で相異なる根を持てば, k' をその分解体として

$n = 2, c = \begin{cases} 4, & k'=k \\ 2, & k' \neq k \end{cases}$. 若し多項式 $a_{2,1}X^2 + a_{4,3}X + a_{6,5}$ が \bar{k} で重根を持つならば, それを

$X = 0$ として $\pi^4 \mid a_4, \pi^6 \mid a_6$ となる.

Step c. 多項式 $Y^2 + a_{3,3}Y - a_{6,6}$ が \bar{k} で異なる根を持つならば, k' をその分解体として $n = 3$

... とこれを繰り返す.

この手順の中の各 Step に依り $\text{ord}(a_3), \text{ord}(a_4), \text{ord}(a_6)$ が増加している. これは, 各 2 つの Step で b_4, b_6, b_8 の位数が上がることを表していて, それはつまり判別式 Δ を割る π の冪が上がっている事を示している. しかし Δ は上で使う様な変換により不変なのでこの subprocedure は停止する.

Step 8. $P(T)$ が \bar{k} で 3 重根を持つとする. 変換により $T = 0$ を根としてよい. つまり $\pi^2 \mid a_2, \pi^3 \mid a_4, \pi^4 \mid a_6$

そこで若し $Y^2 + a_{3,2}Y - a_{6,4}$ が \bar{k} で相異なる根を持てば, k' をその分解体として Type IV*, $m = 7, f =$

$\text{ord}(\Delta) - 6, c = \begin{cases} 3, & k'=k \\ 1, & k' \neq k \end{cases}$.

Step 9. $Y^2 + a_{3,2}Y - a_{6,4}$ が \bar{k} で重根を持つとする. 変換により $Y = 0$ が根であるとすると $\pi^3 \mid a_3, \pi^5 \mid a_6$.

若し $\pi^4 \nmid a_4$ ならば Type III*, $m = 8, f = \text{ord}(\Delta) - 7, c = 2$.

Step 10. $\pi^4 \mid a_4$ とする. $\pi^6 \nmid a_6$ ならば Type II*, $m = 9, f = \text{ord}(\Delta) - 8, c = 1$.

Step 11. $\pi^6 \mid a_6$ とすると, 初めに与えられた Weierstrass 方程式は極小ではない. そこで変換 $(x, y) = (\pi^2 x', \pi^3 y')$ に依り判別式 $\Delta' = \pi^{-12} \Delta$ を持つ Weierstrass 方程式

$$y'^2 + a_{1,1}x'y' + a_{3,3}y'^2 = x'^3 + a_{2,2}x'^2 + a_{4,4}x' + a_{6,7}$$

が得られて, この方程式に対して Step 1 からやり直す.

証明. 概略を述べる. 与えられた Weierstrass 方程式で定義される閉部分スキームを $\mathcal{W} \subset \mathbb{P}_R^2$ とする. このスキームは, E/K を generic ファイバーに持ち, special ファイバー $\tilde{\mathcal{W}}$ は, 元の楕円曲線 E の π による還元 \tilde{E} に対応していた事に注意する. また, \mathcal{W}^0 で \mathcal{W} の滑らかな最大の部分スキームとすると, 定義から $\mathcal{W}^0 \simeq \mathcal{E}^0$ である.

Step 1 の証明. $\pi \nmid \Delta$ に依り $\tilde{\mathcal{W}} = \tilde{E}$ は楕円曲線. この場合, \mathcal{W} は E/K の Néron モデルとなっていて, $\mathcal{E} = \mathcal{W} = \mathcal{E}$. これは, \mathcal{E} が Type I₀ である事を示している. \square

Step 2 の証明. $\pi \mid \Delta$ とすると, $\tilde{\mathcal{W}} = \tilde{E}$ は特異点を持つ. そこでその特異点を原点に移すと, $\pi \mid a_3, a_4, a_6$

である. 今 $\pi \nmid b_2$ と仮定すると, $\tilde{E} = \tilde{\mathcal{W}}$ は乗法的な還元を持つ. これは, $\tilde{\mathcal{W}}$ が結節点 (node) を持つ事と同値であった. そこで, この特異点をブローアップすることで特異点を解消する. こうして得られた曲面が極小固有正則モデル \mathcal{E} と一致し, その special ファイバーは Type I_n ($n = \text{ord}(\Delta)$) となる. 後は乗法的還元の内, 分裂型 (split) であるもの, つまり特異点での傾きが k に入るならば Step 2a の場合となり, そうでないなら Step 2b となる. これを 2 次方程式 $T^2 + a_1T - a_2$ の根で判定している. \square

Step 3 の証明. $\pi \mid b_2$ とする. $b_2 = a_1^2 + 4a_2$ は, 2 次形式 $y^2 + a_1xy - a_2x^2$ の判別式にあたることから,

これは \bar{k} で重根を持つことが分かる. この根を原点に移す事で $\pi \mid a_1, a_2$ として良い. すると $\mathcal{W}^0(k) \simeq k^+$.

今 $\pi^2 \nmid a_6$ とする. これより $\tilde{\mathcal{W}}$ の原点に対応する点 $m = (\pi, x, y)$ は \mathcal{W} の非特異点である. 故に \mathcal{W} は正

則で $\mathcal{C} = \mathcal{W}$, $\mathcal{E} = \mathcal{W}^0 = \mathcal{C}^0$ となる. 従て special ファイバー $\tilde{\mathcal{C}} = \tilde{\mathcal{W}}$ は $y^2 + \tilde{a}_1 xy = x^3 + \tilde{a}_2 x^2$ で定義される曲線である. $\pi \mid b_2$ 依り, この曲線はカスプを持っているので, Type II となることがわかる. \square

Step 4 の証明. $\boxed{\pi^2 \mid a_6}$ とする. \mathcal{W} の特異点 $\pi = x = y = 0$ でブローアップすると,

$$\begin{aligned}\mathcal{W}_1 &: y_1^2 + a_1 x_1 y_1 + a_{3,1} y_1 = \pi x_1^3 + a_2 x_1^2 + a_{4,1} x_1 + a_{6,2}, \\ \mathcal{W}' &: 1 + a_1 x' + a_{3,1} x' = x'^3 y' + a_2 x'^2 + a_{4,1} \pi' x' + a_{6,2} \pi'^2, \quad \pi' y' = \pi, \\ \mathcal{W}'' &: y''^2 + a_1 y'' + a_{3,1} \pi'' y'' = x'' + a_2 + a_{4,1} \pi'' + a_{6,2} \pi''^2, \quad \pi'' x'' = \pi.\end{aligned}$$

但し, $a_{i,j} = \pi^{-j} a_i$. それぞれの special ファイバーを見てやれば \mathcal{W}'' の special ファイバーがブローアップの全ての成分と special ファイバーの特異点を持つことがわかる. 自然な写像 $R[y'', \pi''] \hookrightarrow R[x'', y'', \pi'']$ に依り \mathcal{W}'' は, $y''^2 \pi'' + \pi a_{1,1} y'' \pi'' + a_{3,1} y'' \pi''^2 = \pi + \pi a_{2,1} \pi'' + a_{4,1} \pi''^2 + a_{6,2} \pi''^3$ で定義される $\mathbb{A}_R^2 = \text{Spec } R[y'', \pi'']$ の部分スキームと同型. 更に, これの \mathbb{P}_R^2 での Zariski 閉包を取ると $Y^2 Z + \pi a_{1,1} X Y Z + a_{3,1} Y Z^2 = \pi X^3 + \pi a_{2,1} X^2 Z + a_{4,1} X Z^2 + a_{6,2} Z^3$ で定義されるスキーム $\mathcal{V} \subset \mathbb{P}_R^2 = \text{Proj } R[X, Y, Z]$ となる. \mathcal{V} の special ファイバーを見ると, これは $Z = 0$ と 2 次曲線

$$Y^2 + \tilde{a}_{3,1} Y Z - \tilde{a}_{6,2} Z^2 = \tilde{a}_{4,1} X Z \quad (*)$$

から成り, これらは $Y = Z = 0$ に於いて重複度 2 で交わる. そこで $\pi^3 \nmid b_8$ ならば b_8 の定義と今までの仮定から $\pi^2 \nmid a_4$ となり, これは上の 2 次曲線 (*) が \mathcal{W} の非特異点であることを意味しているので, Type III, $m = 2$, $c = 1$, $f = \text{ord}(\Delta) - 1$. \square

Step 5 の証明. $\boxed{\pi^3 \mid b_8}$ とする. $\pi^2 \mid a_4$ と (*) に依り, \mathcal{V} の special ファイバーは, $Z = 0$ と

$$Y^2 + \tilde{a}_{3,1} Y Z - \tilde{a}_{6,2} Z^2 = 0 \quad (**)$$

となる. そこで $\pi^3 \nmid b_6$ とするならば, $\tilde{b}_{6,2} = \tilde{a}_{3,1} + 4\tilde{a}_{6,2}$ が (*) の判別式であることから, (**) が \bar{k} 上で相異なる根を持つことが分かり, \mathcal{V} の special ファイバーが 1 点で横断的に交わる 3 つの非特異曲線から成る. 故に Type IV であって, $Y^2 + \tilde{a}_{3,1} Y Z - \tilde{a}_{6,2} Z^2$ の分解体を k' とすれば, $k = k'$ か, $k \neq k'$ かに依り c は異なる. \square

$\boxed{\pi^3 \mid b_6}$ と仮定する. これは上の曲線 (**) が k で重根を持つことを意味する. そこでこの根を $Y = 0$ として $\boxed{\pi \mid a_1, a_2, \pi^2 \mid a_3, a_4, \pi^3 \mid a_6}$ と考えてよい. この時 \mathcal{V} の special ファイバーは, 重複度 2 の $Y = 0$ と重複度 1 の $Z = 0$ から成る. そこでこの直線 $\pi = Y = 0$ でブローアップして得られたスキームの special ファイバーを見ると,

$$\tilde{\mathcal{V}} : Y^2 Z = 0, \quad \tilde{\mathcal{V}}_0 : x_1^3 + \tilde{a}_{2,1} x_1^2 + \tilde{a}_{4,2} x_1 + \tilde{a}_{6,3}.$$

となる. そこで多項式 $P(T) = T^3 + \tilde{a}_{2,1} T^2 + \tilde{a}_{4,2} T + \tilde{a}_{6,3}$ の \bar{k} での相異なる根の数で場合分けを行う.

Step 6 の証明. $P(T)$ が 3 つの相異なる根を持つならば, $\tilde{\mathcal{V}}_0$ は 3 つの相異なる直線から成る. 故に Type I_0^* となり, c は P の k での根の数による. \square

Step 7 の証明. $P(T)$ が単根と重根を持つとする. 変換によって $T = 0$ を重根とする. これは $\pi^2 \nmid a_2$, $\pi^3 \mid a_4$, $\pi^4 \mid a_6$ を意味する. \mathcal{V}_0 の special ファイバーは $(x_1 + \tilde{a}_{2,1}) x_1^2 = 0$ なので, $\pi = x_1 = 0$ でブローアップして \mathcal{V}_1 を得る. ここまでの操作で得られた全てのスキームの special ファイバーは, 重複度 1 の直線 $Z = 0$, $x_1 + \tilde{a}_{2,1} = 0$ と重複度 2 の $Y = 0$, $x_1 = 0$, それに, \mathcal{V}_1 の special ファイバー $\tilde{\mathcal{V}}_1$ から成る. $\tilde{\mathcal{V}}_1$ は $y_2^2 + \tilde{a}_{3,2} y_2 - \tilde{a}_{6,4} = 0$ であり, この曲線が 2 つの既約な直線から成る, つまり 2 次方程式 $Y^2 + a_{3,2} Y - a_{6,4}$ が \bar{k} で相異なる根を持てば, Type が定まり, 重根を持てば, 更にブローアップを繰り返す. アルゴリズムのところにも書いてある様に, この操作は必ず有限回で停止する. \square

Step 8 の証明. $P(T)$ が \bar{k} で 3 重根を持つとする. その根を $T = 0$ とすると $\pi^2 \mid a_2, \pi^3 \mid a_4, \pi^4 \mid a_6$. スキーム \mathcal{Y}_0 を $\pi = x_1 = y_2 = 0$ でブローアップすれば,

$$\pi + \pi a_{1,1}x' + a_{3,2}\pi' = x'^3y' + \pi a_{2,2}x'^2 + a_{4,3}x'\pi' + a_{6,5}\pi'^2, \quad \pi = \pi'y'$$

与えられるスキーム $\mathcal{U}' \subset \mathbb{P}_R^3 = \text{Spec } R[\pi', x', y']$ が得られる. この special ファイバーの既約成分と重複度を計算する事で, $\pi^4 \nmid a_4$ であることから Type IV* が得られる. \square

Step 9 の証明. $\pi^3 \mid a_3$ 依り, スキーム \mathcal{U}' は $\pi' = x' = y' = 0$ で特異点を持つ. またブローアップをすることで,

$$\mathcal{U}'' : y''\pi'' + a_{1,1}x''y''\pi'' + a_{3,3}x''y''\pi''^2 = x''^2y'' + a_{2,2}x''^2y''\pi'' + a_{4,3}\pi'' + a_{6,5}\pi''^2, \quad \pi = x''^2y''\pi''$$

得られる. Step 8 と同様に既約成分と重複度を求める事で, $\pi^4 \nmid a_4$ であることから Type III* が得られる. \square

Step 10 の証明. $\pi^4 \mid a_4$ とすると, \mathcal{U}'' は $\pi'' = x'' = y'' = 0$ で特異点を持つ. この点でブローアップをすることで,

$$\mathcal{U}''' : \pi''' + a_{1,1}x'''y'''\pi''' + a_{3,3}x'''y'''^2\pi'''^2 = x'''^2y''' + a_{2,2}x'''^2y'''^2\pi''' + a_{4,4}x'''^2y'''^3\pi'''^2 + a_{6,5}\pi'''^2, \quad \pi = x'''^2y'''^4\pi'''$$

得られる. Step 9 と同様に既約成分と重複度を求める事で, $\pi^6 \nmid a_6$ であることから Type III* が得られる. \square

Step 11 の証明. $\pi^6 \mid a_6$ とする. これまでの仮定を見ると, $\pi \mid a_1, \pi^2 \mid a_2, \pi^3 \mid a_3, \pi^4 \mid a_4, \pi^6 \mid a_6$ である. \mathcal{U}''' で $\pi''' = 1/\pi y_2^2, x''' = x_2^2/\pi y_2^3, y''' = \pi y_2^2/x_3$ とすれば,

$$y_3^2 + a_{1,1}x_2y_3 + a_{3,3}y_3 = x_2^3 + a_{2,2}x_2^2 + a_{4,4}x_2 + a_{6,6}$$

なる Weierstrass 方程式が得られ, この判別式 Δ' は, $\Delta' = \pi^{-12}\Delta$ となる. \square

Step 11 の証明にある様に, この手順を繰り返すと極小 Weierstrass 方程式が得られ必ず Kodaira 型が定まる. \square

系 3. \tilde{E}/k を π に依る E の還元, $\tilde{E}_{\text{ns}}(k)$ で $\tilde{E}(k)$ の非特異点全体, $E_0(K) = \{P \in E(K) \mid \tilde{P} \in \tilde{E}_{\text{ns}}(k)\}$, $E_1(K) = \{P \in E(K) \mid \tilde{P} = \tilde{O}\}$ とすると, $E(K) = \mathcal{E}(R)$, $E_0(K) = \mathcal{E}^0(R)$ となり, 特に, $E(K)/E_0(K) \simeq \mathcal{E}(R)/\mathcal{E}^0(R)$.

Tate のアルゴリズムを実際にプログラムで書いてみよう. Tate のアルゴリズムの条件で一番簡単な $R = \mathbb{Z}_p$ の場合に, 次の様なプログラムを考える.

入力 : Weierstrass 方程式の係数 $a_1, a_2, a_3, a_4, a_6 \in \mathbb{Z}$, 素数 p ,

出力 : Kodaira 型 K_p , 導手の中の p の指数 f_p , 局所連結成分の数 c_p .

ここでは Pari/GP を用いたが, Pari 固有の関数は殆ど使っていないので他のプログラム言語に移植する事は容易であろう. 先ず, 次の様なサブルーチンを準備する.

compute_invariants() : 与えられた a_1, a_2, a_3, a_4, a_6 から $b_2, b_4, b_6, b_8, c_4, c_6, \Delta$ を計算する.

valuation(a,p) : a の p での付値. つまり $\text{ord}_p(a)$.

reducible(a,p) : a が p で割り切れるなら TRUE, 割り切れないなら FALSE を返す.

transcoord(r,s,t,u) : 与えられた r, s, t, u で変換

$$x = u^2x' + r, \quad y = u^3y' + u^2sx' + t$$

を行い, ついでに $b_2, b_4, b_6, b_8, c_4, c_6, \Delta$ を計算しなおす.

quadroots(a,b,c,p) : 二次合同式 $ax^2 + bx + c \equiv 0 \pmod{p}$ が解を持つとき TRUE を, 持たないときは FALSE を返す.

nrootscubic(b,c,d,p) : 合同式 $x^3 + bx^2 + cx + d \equiv 0 \pmod{p}$ の解の個数を返す.

Tate のアルゴリズム (Pari/GP)

```
001: /**
002:  * @param a1,a2,a3,a4,a6 : coefficients of Weierstrass eq.
003:  * @param p : closed point
004:  * @return [Kp,fp,cp,[a1,a2,a3,a4,a6]]
005:  *   Kp : Kodaira Type (String)
006:  *   fp : exponent of conductor in p
007:  *   cp : local index
008:  *   [a1,a2,a3,a4,a6] : Weierstrass coeff.
009:  **/
010: local_red(a1,a2,a3,a4,a6, p) =
011: {
012:   local(TRUE,FALSE,Kp,fp,cp,b2,b4,b6,b8,c4,c6,disc);
013:   TRUE = 1;
014:   FALSE = 0;
015:   Kp = ""; \\ Kodaiara Type
016:   fp = -1; \\ exp. of conductor in p
017:   cp = -1; \\ local index
018:
019:   while(TRUE,
020:     /* --- STEP 1 --- */
021:     compute_invariants();
022:     n = valuation(disc,p);
023:     \\ test for type I_0
024:     if(n==0, Kp = "I0"; fp = 0; cp = 1; break);
025:
026:     /* --- STEP 2 --- */
027:     \\ Change coordinates so that p | a3,a4,a6
028:     if(p==2,
029:       if(reducible(b2, p) == TRUE,
030:         r = a4 % p; t = r*(1+a2+a4)+a6 % p;
031:         ,
032:         r = a3 % p; t = r + a4 % p;
033:       );
034:     );
035:     if(p==3,
036:       if(reducible(b2, p) == TRUE, r = -b6 % p; , r = -b2*b4 % p );
037:       t = -inv(2,p) * (a1*r+a3);
038:     );
039:     if(p>3,
040:       if(reducible(c4, p) == TRUE, r = -inv(12,p) * b2; , r = -12*c4 * (c6 + b2*c4); );
041:       t = -inv(2,p) * (a1*r+a3);
042:       r = r % p; t = t % p;
043:     );
044:     transcoord(r, 0, t, 1);
045:
046:     \\ test for type I_n
047:     if(reducible(b2,p) == FALSE,
048:       if(quadroots(1,a1,-a2,p) == TRUE,
049:         cp = n
```

```

050:     ,
051:     if(reducible(n, 2) == TRUE, cp = 2, cp = 1);
052:     );
053:     Kp = Str("I" n); fp = 1; break;
054: );
055:
056: /* --- STEP 3 (test for type II) --- */
057: if(reducible(a6,p^2) == FALSE, Kp = "II"; fp = n; cp = 1; break );
058:
059: /* --- STEP 4 (test for type III) --- */
060: if(reducible(b8,p^3) == FALSE, Kp = "III"; fp = n-1; cp = 2; break );
061:
062: /* --- STEP 5 (test for type IV) --- */
063: if(reducible(b6,p^3) == FALSE,
064:     if(quadroots(1, a3/p, -a6/p^2, p) == TRUE, cp = 3, cp=1);
065:     Kp = "IV"; fp = n-2; break;
066: );
067:
068: /* --- STEP 6 --- */
069: \\ Change coordinates so that p|a1,a2; p^2|a3,a4; p^3|a6
070: if(p==2,
071:     s = a2 % 2; t = 2*(a6/4 %2);
072:     ,
073:     s = -a1*inv(2,p); t = -a3*inv(2,p);
074: );
075: transcoord(0, s, t, 1);
076:
077: \\ Set up auxiliary cubic T^3 + bT^2 +cT+d
078: b = a2/p; c = a4/p^2; d = a6/p^3;
079: w = 27*d^2-b^2*c^2+4*b^3*d-18*b*c*d+4*c^3; \\ discriminant
080: x = 3*c-b^2;
081:
082: \\ Test for distinct roots : type I_0^*
083: if(reducible(w,p) == FALSE, Kp = "I*0"; fp = n-4; cp = 1 + nrootscubic(b,c,d,p); break);
084:
085: /* --- STEP 7 --- */
086: \\ test for double root : Type I_m^*
087: if(reducible(x,p) == FALSE,
088:     \\ change coordinates so that the double root is T = 0
089:     if(p==2, r = c);
090:     if(p==3, r = b*c);
091:     if(p>3, r = (b*c-9*d)*inv(2*x,p));
092:     r = p*(r%p);
093:     transcoord(r,0,0,1);
094:
095:     /* --- Subprocedure --- */
096:     \\ Make a3,a4,a6 repeatedly more divisible by p
097:     m = 1; mx = p^2; my = p^2; cp = 0;
098:     while(cp==0,
099:         xa2 = a2/p; xa3 = a3/my; xa4 = a4/(p*mx); xa6 = a6/(mx*my); \\ Y^2 a_{3,2} Y - a_{6,4}
100:         /* --- Step a --- */
101:         if(reducible(xa3^2+4*xa6, p) == FALSE,
102:             if(quadroots(1,xa3,-xa6,p) == TRUE, cp = 4, cp = 2);
103:         ,
104:         /* --- Step b --- */
105:         if(p==2, t = my*xa6, t = my*((-xa3*inv(2,p)) %p) );
106:         transcoord(0,0,t,1);
107:         my = my*p; m = m+1;
108:         xa2 = a2/p; xa3 = a3/my; xa4 = a4/(p*mx); xa6 = a6/(mx*my);

```

```

109:         if(reducible(xa4^2-4*xa2*xa6, p) == FALSE,
110:             if(quadroots(xa2,xa4,xa6,p) == TRUE, cp = 4, cp = 2);
111:         ,
112:         if(p==2, r = mx*((xa6*xa2) % 2), r = mx*((-xa4*inv(2*xa2,p)) %p));
113:         transcoord(r,0,0,1);
114:         mx = mx*p; m = m+1;
115:     );
116: );
117: ); \\ end while
118: fp = n-m-4; Kp = Str("I*" m); break;
119: );
120:
121: /* --- STEP 8 --- (triple root case) */
122: \\ change coordinates so that the triple root is T=0
123: if(p==3, rp = -d, rp = -b*inv(3,p));
124: r = p*(rp%p);
125: transcoord(r,0,0,1);
126: x3 = a3/p^2; x6 = a6/p^4;
127:
128: \\ Test for Type IV*
129: if(reducible(x3^2+4*x6,p) == FALSE,
130:     if(quadroots(1,x3,-x6,p) == TRUE, cp = 3, cp = 1);
131:     Kp = "IV*"; fp = n-6; break;
132: );
133:
134: /* --- STEP 9 --- (distinct 3 roots case) */
135: \\ change coordinates so that p^3|a3, p^5|a6
136: if(p==2, t = x6, t = x3*inv(2,p));
137: t = -p^2*(t%p);
138: transcoord(0,0,t,1);
139:
140: \\ Test for Type III*
141: if(reducible(a4, p^4) == FALSE, Kp = "III*"; fp = n-7; cp = 2; break;);
142:
143: /* --- STEP 10 --- (Test for Type II*) */
144: if(reducible(a6, p^6) == FALSE, Kp = "II*"; fp = n-8; cp = 1; break;);
145:
146: /* --- STEP 11 --- Equation non-minimal : divide each a_i by p^i and start again */
147: transcoord(0,0,0, p);
148: );
149:
150: return([Kp,fp,cp, [a1,a2,a3,a4,a6]]);
151: }

```

幾つか実行例を挙げてみる.

例 4. $y^2 + y = x^3 - x^2 - 7820x - 263580$ なる楕円曲線 (11A2(C)) では $p = 11$ で Kodaira 型が I_1 となる. 上のプログラムを使うと,

```

? local_red(0,-1,1,-7820,-263580, 11)
%2 = ["I1", 1, 1, [0, 14, 11, -7755, -302610]]

```

となり Kodaira 型が I_1 となっていることが分かる. 幾つか試してみると次の様な結果が得られた.

表 1 Kodaira 型の計算

	a_1	a_2	a_3	a_4	a_6	p	Kodaira	c_p
11A2(C)	0	-1	1	-7820	-263580	11	I_1	1
11A1(B)	0	-1	1	-10	-20	11	I_5	5
24A4(A)	0	-1	0	1	0	2	III	2
27A4(C)	0	0	1	-30	63	3	IV	1
32A3(C)	0	0	0	-11	-14	2	I_0^*	1
32A1(B)	0	0	0	4	0	2	I_3^*	4
693D5	1	-1	0	-40671	3167194	3	I_8^*	4
20A3(D)	0	1	0	-36	-140	2	IV^*	1
24A3(D)	0	-1	0	-64	220	2	III^*	2
24A5(F)	0	-1	0	-384	-2772	2	II^*	1
24A6(E)	0	-1	0	-16	-180	2	II^*	1
696B1	0	1	0	8	-16	2	II^*	1

因みに、上記の Kodaira 型を求める関数は Pari/GP の中で標準的に用意されていて、次の様にすればよい。

```
?E = ellinit([0,-1,1,-7820,-263580]); \\ 楕円曲線の初期化
?elllocalred(E,11)
```

参考文献

- [数論 3] 黒川信重・栗原将人・斉藤毅, 数論 **3**, 岩波書店, 2001.
- [志村] Goro Shimura, *Introduction to the Arithmetic Theory of Automorphic Functions*, Princeton University Press (1994)
- [Sil1] Silverman, J., *The Arithmetic of Elliptic Curves*, Graduate Texts in Math. **106**, New York: Springer-Verlag, 1975.
- [Sil2] Silverman, J., *Advanced Topics in the Arithmetic of Elliptic Curves*, Graduate Texts in Math. **106**, New York: Springer-Verlag, 1975.
- [Ta] J. Tate, Algorithm for determining the type of a syngular fiber in an elliptic pencil, In *Modular Functions of One Variable IV*, Lect. Notes in Math. 476, B.J. Birch and W. Kuyk, eds., Springer-Verlag, Berlin, 1975, 33-52.
- [Har] Robin Hartshorne, *Algebraic Geometry, Graduate Texts in Math.* **52**, New York: Springer-Verlag, 2000.
- [Li] Stephen Lichtenbaum, *Curves over discrete valuation ring*, Amer. J. Math. 90 (1968), 380-403.