EFFICIENT REDUCTION OF BINARY FORMS

MICHAEL STOLL

ABSTRACT. We explain how one can efficiently reduce a binary form (in the sense of [SC03]).

1 Introduction

Let $F \in \mathbb{R}[x,z]$ be a binary form of some degree n. We assume that F is *stable*, i.e., F is not divisible (over $\mathbb{C}$) by the mth power of a linear form with $2m \geq n$. Then one can associate to F its *Julia covariant* $z(F)$ in the upper half plane $\mathcal{H}$, which is the unique point where the sum of the unit tangent vectors in direction of the roots of F (which are identified with points in $\mathbb{P}^1(\mathbb{C})$, viewed as the boundary at infinity of hyperbolic three-space $\mathcal{H}^3$; $\mathcal{H}$ can be seen as the subset of $\mathcal{H}^3$ that is fixed by the involution induced by complex conjugation on the boundary) vanishes; see [SC03, Cor. 5.4].

In terms of formulas, this description is equivalent to the following. Write $F(x,z) = \prod_{j=1}^{n}(\beta_j x - \alpha_j x)$ with $\alpha_j, \beta_j \in \mathbb{C}$ and set

$$\tilde{F}(t,u) = \prod_{j=1}^{n}(|\beta_j t - \alpha_j|^2 + |\beta_j|^2 u^2)$$

for $t \in \mathbb{R}$ (or $\mathbb{C}$) and $u \in \mathbb{R}_{>0}$. Then the Julia covariant of F is the unique point $t + iu \in \mathcal{H}$ that minimizes $\tilde{F}(t,u)/u^n$; see [SC03, Prop. 5.1].

The Julia covariant is defined (as a point in $\mathcal{H}^3$) for stable binary forms with complex coefficients and is equivariant with respect to the action of $\mathrm{SL}(2,\mathbb{C})$ on binary forms by linear substitution of variables on the one hand and by (generalized) Möbius transformation on $\mathcal{H}^3$ on the other hand. It restricts to an $\mathrm{SL}(2,\mathbb{R})$-equivariant map from stable binary forms with real coefficients to $\mathcal{H}$.

A stable binary form $F \in \mathbb{R}[x,z]$ is then *reduced*, if its covariant $z(F)$ lies in the standard fundamental domain

$$\mathcal{F} = \bigl\{z \in \mathcal{H} : -\tfrac{1}{2} \leq \operatorname{Re} z \leq \tfrac{1}{2},\ |z| \geq 1\bigr\}$$

for the action of $\mathrm{SL}(2,\mathbb{Z})$ on $\mathcal{H}$. The equivariance of $z(F)$ then implies that every form is $\mathrm{SL}(2,\mathbb{Z})$-equivalent to a reduced form, and that the latter is essentially unique (there can be several reduced orbit representatives when $z(F)$ lies on the boundary of $\mathcal{F}$).

In applications, it is useful to be able to determine a reduced form in the $\mathrm{SL}(2,\mathbb{Z})$-orbit of a given (stable) binary form, because the reduced representative tends to have smaller coefficients, which is helpful when one needs to do computations. For example, a reduction algorithm can be used to produce good models of hyperelliptic curves.

Date: August 8, 2025.

The goal of this note is to explain how to determine a transformation $\gamma \in \mathrm{SL}(2,\mathbb{Z})$ such that $F \cdot \gamma$ is reduced, in a reasonably efficient way. Here $\mathrm{SL}(2,\mathbb{Z})$ acts on binary forms on the right by linear substitution of the variables.

The Computer Algebra System Magma [BCP97] currently (version 2.28-26) includes a function `Reduce` that performs this task by first computing the Julia covariant $z(F)$ and then determining γ such that $\gamma \cdot z(F) \in \mathcal{F}$ (via the standard shift-and-invert procedure), where $\mathrm{SL}(2,\mathbb{Z})$ acts via Möbius transformations. Then $F \cdot \gamma^{-1}$ is reduced. The computation of the covariant proceeds by numerically minimizing $\tilde{F}(t,u)/u^n$, which in some cases is challenging due to numerical instabilities. We propose an algorithm that avoids these problems. The main idea is that we do not need to determine the covariant precisely; it suffices to decide whether its real part is positive or negative. Furthermore, the latter reduces to determining the minimum of $\tilde{F}(0,e^u)e^{-nu}$, which is a one-dimensional convex optimization problem instead of the two-dimensional original problem.

One can then also use this approach to determine the Julia covariant $z(F)$, by pinning down its real part using the procedure mentioned above and a bisection scheme. The imaginary part is then given as e^u, where u is the minimizing value along the vertical geodesic (parameterized by arc length) given by the real part.

2 The reduction algorithm

The main building block is the following procedure. For a multiset S of points in $\mathbb{P}^1(\mathbb{C})$, we define $z(S)$ to be $z(F)$, where F is any nonzero binary form whose roots are given by S. The condition on S below is equivalent to saying that F is stable.

Algorithm 2.1 (Left or Right?)**.**

function RePos?(F)
Input: A multiset $S = \{\alpha_j : 1 \le j \le n\} \subset \mathbb{P}^1(\mathbb{C})$ such that all multiplicities are $< n/2$.
Output: **true**, if $\mathrm{Re}\, z(S) \ge 0$, **false** otherwise.

1. $S' \leftarrow \{\alpha \in S : \alpha \neq \infty\}$; $m \leftarrow n - \#S'$
2. $S \leftarrow \{\alpha \in S' : \alpha \neq 0\}$; $m \leftarrow m - (\#S' - \#S)$
3. Find $\eta_0 \in \mathbb{R}$ such that η_0 minimizes
$$\eta \longmapsto h(\eta) = \sum_{\alpha \in S} \log\left(|\alpha| e^{-\eta} + |\alpha|^{-1} e^{\eta}\right) - m\eta\,.$$
4. Compute
$$\delta = \sum_{\alpha \in S} \frac{\mathrm{Re}(\alpha)}{|\alpha|^2 + e^{2\eta_0}}\,.$$
5. If $\delta \ge 0$ then return **true**, else return **false**.

In practice, it makes sense to also return δ and η_0.

For Step 3, one can use Newton's method to find the unique zero of h', falling back to a bisection step when the Newton step does not decrease the absolute value of h' sufficiently.

Lemma 2.2. *Algorithm 2.1 is correct.*

Proof. Writing $\alpha_j = (x_j : y_j)$, by [SC03, Section 5] we have to minimize $\log(\tilde{F}(0, e^\eta)e^{-n\eta})$, where

$$\log(\tilde{F}(t,u)/u^n) = \sum_{j=1}^{n} \log\bigl(|y_j t - x_j|^2/u + |y_j|^2 u\bigr)\,,$$

so

$$\log(\tilde{F}(0, e^\eta)e^{-n\eta}) = \sum_{j=1}^{n} \log\bigl(|x_j|^2 e^{-\eta} + |y_j|^2 e^{\eta}\bigr)\,.$$

Scaling any pair (x_j, y_j) by a common nonzero factor only changes this function by adding a constant. So we can assume that $x_j y_j = 1$ or $(x_j, y_j) \in \{(1,0),(0,1)\}$. The terms with $x_j = 0$ then give a contribution of η each, and the terms with $y_j = 0$ give a contribution of $-\eta$ each. For $\alpha_j \notin \{0,\infty\}$, we can write the summand as

$$\log\bigl(|\alpha_j|/e^\eta + e^\eta/|\alpha_j|\bigr)\,.$$

This shows that the function h given in the algorithm is correct (up to adding a constant).

Now we note that for any $r \in \mathbb{R}_{>0}$, the function

$$\eta \longmapsto \log(r/e^\eta + e^\eta/r)$$

is strictly convex: its second derivative is

$$\frac{4}{(r/e^\eta + e^\eta/r)^2} = \frac{1}{\cosh(\eta - \log r)^2} > 0\,.$$

This implies that h is also strictly convex. Also, the stability condition implies that $h(\eta) \to +\infty$ as $\eta \to \pm\infty$; so h has a unique minimum.

It follows that $(0, e^{\eta_0})$ is the (unique) point on $\{0\} \times \mathbb{R}_{>0}$ minimizing $\tilde{F}(t,u)/u^n$ there.

Also,

$$\frac{d}{dt} \log \frac{\tilde{F}(t, e^{\eta_0})}{e^{n\eta_0}}\Big|_{t=0} = -2\delta\,.$$

Consider the geodesic γ from $ie^{\eta_0} \in \mathcal{H}$ to $z(F)$. Acting by an element of $\mathrm{SL}(2,\mathbb{R})$ that moves γ to the geodesic from 0 to $i\infty$, we see that the argument above shows that $\log(\tilde{F}(t,u)/u^n)$ is strictly convex along γ as well. Unless $z(F) = ie^{\eta_0}$ (in which case $\delta = 0$), the derivative of $\log(\tilde{F}(t,u)/u^n)$ along γ (taken in the direction of $z(F)$) at ie^{η_0} must be strictly negative. But as $z(F)$ is not on the imaginary axis when $\delta \neq 0$, this derivative is a positive multiple of -2δ when $\operatorname{Re} z(F) > 0$ and a negative multiple of -2δ when $\operatorname{Re} z(F) < 0$. This shows that the returned boolean value is correct. □

To reduce a stable binary form F with multiset $S \subset \mathbb{P}^1(\mathbb{C})$ of roots, we now proceed as follows. First we determine an integer m such that $m - \frac{1}{2} \le \operatorname{Re} z(F) \le m + \frac{1}{2}$. To this end, we use Algorithm 2.1 repeatedly with $\{\alpha - (k + \frac{1}{2}) : \alpha \in S\}$ for suitable integers k. Then we shift (the roots of) F by m to obtain $|\operatorname{Re} z(F)| \le \frac{1}{2}$. Next, we determine if $|z(F)| \ge 1$ or not. If so, we are done. Otherwise, we replace F by $F(-z, x)$ and repeat. Note that checking whether $|z(F)| \ge 1$ or not can be done by applying Algorithm 2.1 again; this time with $S' = \{(\alpha + 1)/(\alpha - 1) : \alpha \in S\}$. This transformation sends the circular arc going from -1 to 1 in $\mathcal{H}$ to the vertical geodesic from 0 to ∞. Then $\operatorname{Re} z(S') \ge 0$ is equivalent to $|z(S)| \ge 1$.

To get something reasonably efficient, we double the step size in the first part of this procedure until we have enclosed $\operatorname{Re} z(F)$ between two numbers in $\mathbb{Z} + \frac{1}{2}$. Then we use binary subdivision to pin down the vertical strip containing $z(F)$.

In the following, we write $S - r$ for $\{\alpha - r : \alpha \in S\}$, where $r \in \mathbb{R}$, with the understanding that $\infty - r = \infty$. The action of $\mathrm{SL}(2,\mathbb{Z})$ on binary forms is on the right and given by

$$F(x,z) \cdot \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) = F(ax + bz, cx + dz)\,.$$

Algorithm 2.3 (Reduction)**.**
Input: $F \in \mathbb{Z}[x,z]$ homogeneous of degree n, stable.
Output: $\gamma \in \mathrm{SL}(2,\mathbb{Z})$ such that $F \cdot \gamma$ is reduced.

$\gamma \leftarrow \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ // *initialize*
while true do
  $S \leftarrow$ multiset of roots of F in $\mathbb{P}^1(\mathbb{C})$
  // *Find* $l, u \in \mathbb{Z}$ *with* $l - \frac{1}{2} \le \operatorname{Re} z(F) \le u - \frac{1}{2}$.
  if RePos?$(S - \frac{1}{2})$ **then**
    $k \leftarrow 0$; $l \leftarrow 0$; $u \leftarrow l + 1$
    while RePos?$(S - (u + \frac{1}{2}))$ **do**
      $k \leftarrow k + 1$; $l \leftarrow u$; $u \leftarrow u + 2^k$
    end while
  else
    $k \leftarrow 0$; $u \leftarrow 0$; $l \leftarrow u - 1$
    while not RePos?$(S - (l + \frac{1}{2}))$ **do**
      $k \leftarrow k + 1$; $u \leftarrow l$; $l \leftarrow l - 2^k$
    end while
  end if
  // *Now* $l - \frac{1}{2} \le \operatorname{Re} z(F) \le u - \frac{1}{2}$. *Bisect to get* $u = l + 1$.
  while $u - l > 1$ **do**
    $m \leftarrow \lfloor (l + u)/2 \rfloor$
    if RePos?$(S - (m + \frac{1}{2}))$ **then** $l \leftarrow m$ **else** $u \leftarrow m$ **end if**
  end while
  // *Now* $l - \frac{1}{2} \le \operatorname{Re} z(F) \le u - \frac{1}{2} = l + \frac{1}{2}$*; shift by* l.
  $F \leftarrow F(x + lz, z)$; $\gamma \leftarrow \gamma \cdot \left(\begin{smallmatrix} 1 & l \\ 0 & 1 \end{smallmatrix}\right)$
  // *Check whether* $|z(F)| \ge 1$. *If so, we are done.*
  if RePos?$(\{(\alpha + 1)/(\alpha - 1) : \alpha \in S\})$ **then return** γ **end if**
  // $|z(F)| < 1$*: invert and repeat.*
  $F \leftarrow F(-z, x)$; $\gamma \leftarrow \gamma \cdot \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$
end while

3 Computation of the covariant

We can use the RePos? function also to find $z(F)$. To this end, we modify the reduction algorithm as follows.

Algorithm 3.1 (Covariant)**.**
Input: $F \in \mathbb{Z}[x,z]$ homogeneous of degree n, stable; an error bound ε.
Output: $z(F) \in \mathcal{H}$ to precision ε.

$S \leftarrow$ multiset of roots of F in $\mathbb{P}^1(\mathbb{C})$
if RePos?(S) **then**
  $k \leftarrow 0$; $l \leftarrow 0$; $u \leftarrow l + 1$
  while RePos?$(S - u)$ **do**
    $k \leftarrow k + 1$; $l \leftarrow u$; $u \leftarrow u + 2^k$
  end while
else
  $k \leftarrow 0$; $u \leftarrow 0$; $l \leftarrow u - 1$
  while not RePos?$(S - l)$ **do**
    $k \leftarrow k + 1$; $u \leftarrow l$; $l \leftarrow l - 2^k$
  end while
end if
// Now $l \leq \operatorname{Re} z(F) \leq u$. *Bisect until desired precision is reached.*
while $u - l > \varepsilon$ **do**
  $m \leftarrow (l + u)/2$
  if RePos?$(S - m)$ **then** $l \leftarrow m$ **else** $u \leftarrow m$ **end if**
end while
return $m + e^{\eta_0 i}$, where η_0 is computed in RePos?$(S - m)$

References

[BCP97] Wieb Bosma, John Cannon, and Catherine Playoust, *The Magma algebra system. I. The user language*, J. Symbolic Comput. **24** (1997), no. 3-4, 235–265, DOI 10.1006/jsco.1996.0125. Computational algebra and number theory (London, 1993). MR1484478 ↑1

[SC03] Michael Stoll and John E. Cremona, *On the reduction theory of binary forms*, J. Reine Angew. Math. **565** (2003), 79–99, DOI 10.1515/crll.2003.106. MR2024647 ↑(document), 1, 2

MATHEMATISCHES INSTITUT, UNIVERSITÄT BAYREUTH, 95440 BAYREUTH, GERMANY

Email address: Michael.Stoll@uni-bayreuth.de

URL: http://www.mathe2.uni-bayreuth.de/stoll/