# Coleman Integration in Larger Characteristic

ANTS XIII — University of Wisconsin, Madison

Alex J. Best

17/7/2018

Boston University

# The big picture

Kedlaya

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner,

Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more. . .

# The big picture

$$\text{CaDeVe-Go-GaGu-Sh-Tu} \xrightarrow{\text{More curves}} \text{Kedlaya}$$

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more...

# The big picture

$$\begin{tikzcd}
 & \text{BaBrKe} \arrow[d, "\text{Coleman integration}"] \\
\text{CaDeVe-Go-GaGu-Sh-Tu} \arrow[r, "\text{More curves}"] & \text{Kedlaya}
\end{tikzcd}$$

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more. . .

# The big picture

```tikzcd
\text{Ba-BaTu} \arrow[r] \arrow[d] & \text{BaBrKe} \arrow[d, "\text{Coleman integration}"] \\
\text{CaDeVe-Go-GaGu-Sh-Tu} \arrow[r, "\text{More curves}"] & \text{Kedlaya}
\end{tikzcd}
```

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more...

# The big picture

$$
\begin{tikzcd}
\text{Ba-BaTu} \arrow[r] \arrow[dd] & \text{BaBrKe} \arrow[dd, "\text{Coleman integration}"] & \\
& & \text{Ha} \arrow[dl, "\widetilde{O}(\sqrt{p})"] \\
\text{CaDeVe-Go-GaGu-Sh-Tu} \arrow[r, "\text{More curves}"] & \text{Kedlaya} &
\end{tikzcd}
$$

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more. . .

# The big picture

$$
\begin{tikzcd}
\text{Ba-BaTu} \arrow[rr] \arrow[dd] & & \text{BaBrKe} \arrow[dd, "\text{Coleman integration}"] & \\
& \text{Mi-ArBeCoMaTr} \arrow[rr] \arrow[dl] & & \text{Ha} \arrow[dl, "\widetilde{O}(\sqrt{p})"] \\
\text{CaDeVe-Go-GaGu-Sh-Tu} \arrow[rr, "\text{More curves}"] & & \text{Kedlaya} &
\end{tikzcd}
$$

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more. . .

# The big picture

$$
\begin{tikzcd}
 & & & \text{Be} \arrow[dl] \arrow[dd] \\
\text{Ba-BaTu} \arrow[rr] \arrow[dd] & & \text{BaBrKe} \arrow[dd, "\text{Coleman integration}"] & \\
 & \text{Mi-ArBeCoMaTr} \arrow[rr, crossing over] \arrow[dl] & & \text{Ha} \arrow[dl, "\widetilde{O}(\sqrt{p})"] \\
\text{CaDeVe-Go-GaGu-Sh-Tu} \arrow[rr, "\text{More curves}"] & & \text{Kedlaya} &
\end{tikzcd}
$$

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more...

## The big picture

$$
\begin{tikzcd}
 & \text{🏆} \arrow[rr] \arrow[dl] \arrow[dd] & & \text{Be} \arrow[dl] \arrow[dd] \\
\text{Ba-BaTu} \arrow[rr] \arrow[dd] & & \text{BaBrKe} \arrow[dd, "\text{Coleman integration}"] & \\
 & \text{Mi-ArBeCoMaTr} \arrow[rr] \arrow[dl] & & \text{Ha} \arrow[dl, "\widetilde{O}(\sqrt{p})"] \\
\text{CaDeVe-Go-GaGu-Sh-Tu} \arrow[rr, "\text{More curves}"] & & \text{Kedlaya} &
\end{tikzcd}
$$

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more. . .

# The big picture

$$
\begin{tikzcd}
 & \text{🏆} \arrow[rr] \arrow[dl] \arrow[dd] & & \text{Be} \arrow[dl] \arrow[dd] \\
\text{Ba-BaTu} \arrow[rr] \arrow[dd] & & \text{BaBrKe} \arrow[dd, "\text{Coleman integration}"] & \\
 & \text{Mi-ArBeCoMaTr} \arrow[rr] \arrow[dl] & & \text{Ha} \arrow[dl, "\widetilde{O}(\sqrt{p})"] \\
\text{CaDeVe-Go-GaGu-Sh-Tu} \arrow[rr, "\text{More curves}"] & & \text{Kedlaya} &
\end{tikzcd}
$$

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more...

# The big picture

$$\begin{tikzcd}
 & \text{🏆} \arrow[rr] \arrow[dl] \arrow[dd] & & \text{Be} \arrow[dl] \arrow[dd] \\
\text{Ba-BaTu} \arrow[rr] \arrow[dd] & & \text{BaBrKe} \arrow[dd, "\text{Coleman integration}"] & \\
 & \text{Mi-ArBeCoMaTr} \arrow[rr] \arrow[dl] & & \text{Ha} \arrow[dl, "\widetilde{O}(\sqrt{p})"] \\
\text{CaDeVe-Go-GaGu-Sh-Tu} \arrow[rr, "\text{More curves}"] & & \text{Kedlaya} &
\end{tikzcd}$$

Arul, Balakrishnan, Best, Bradshaw, Castryk, Costa, Denef, Gaudry, Gurel, Harvey, Kedlaya, Magner, Minzlaff, Shieh, Triantafillou, Tuitman, Vercauteren, and more...

There is (at least) one dimension missing: Small $p$!

# Motivation

(Explicit) Coleman integration is a central tool in (non-abelian) Chabauty and useful in several other areas of arithmetic geometry, but, algorithms lag behind the related ones for zeta functions.

# Motivation

(Explicit) Coleman integration is a central tool in (non-abelian) Chabauty and useful in several other areas of arithmetic geometry, but, algorithms lag behind the related ones for zeta functions.

Longer term goals:

- Adapt descendents of Kedlaya's algorithm to compute (iterated) Coleman integrals, e.g.:

# Motivation

(Explicit) Coleman integration is a central tool in (non-abelian) Chabauty and useful in several other areas of arithmetic geometry, but, algorithms lag behind the related ones for zeta functions.

Longer term goals:

- Adapt descendents of Kedlaya's algorithm to compute (iterated) Coleman integrals, e.g.:
  - Larger characteristic (Harvey)

# Motivation

(Explicit) Coleman integration is a central tool in (non-abelian) Chabauty and useful in several other areas of arithmetic geometry, but, algorithms lag behind the related ones for zeta functions.

Longer term goals:

- Adapt descendents of Kedlaya's algorithm to compute (iterated) Coleman integrals, e.g.:
  - Larger characteristic (Harvey)
  - Average polynomial time (Harvey)

# Motivation

(Explicit) Coleman integration is a central tool in (non-abelian) Chabauty and useful in several other areas of arithmetic geometry, but, algorithms lag behind the related ones for zeta functions.

Longer term goals:

- Adapt descendents of Kedlaya's algorithm to compute (iterated) Coleman integrals, e.g.:
  - Larger characteristic (Harvey)
  - Average polynomial time (Harvey)
  - Effectiver average polynomial time (Harvey-Sutherland)

# Motivation

(Explicit) Coleman integration is a central tool in (non-abelian) Chabauty and useful in several other areas of arithmetic geometry, but, algorithms lag behind the related ones for zeta functions.

Longer term goals:

- Adapt descendents of Kedlaya's algorithm to compute (iterated) Coleman integrals, e.g.:
    - Larger characteristic (Harvey)
    - Average polynomial time (Harvey)
    - Effectiver average polynomial time (Harvey-Sutherland)
- Applications to rational points, combining congruence information for many primes, 1-step (Mordell-Weil) seiving.

# Coleman integration

Throughout we take $X/\mathbf{Q}_p$ a genus $g$ odd degree hyperelliptic curve, and $p$ an odd prime. We pick a lift of the Frobenius map, $\phi^*\colon X \to X$.

**Theorem (Coleman)**

*There is a* $\mathbf{Q}_p$*-linear map* $\int_b^x\colon \Omega^1_{A^\dagger} \otimes \mathbf{Q}_p \to A_{\mathrm{loc}}(X)$ *satisfying:*

1. $$\mathrm{d} \circ \int_b^x = \mathrm{id}\colon \Omega^1_{A^\dagger} \otimes \mathbf{Q}_p \to \Omega^1_{A^\dagger} \otimes \mathbf{Q}_p \leftarrow \textit{(FTC)}$$
2. $$\int_b^x \circ\, \mathrm{d}\colon A^\dagger \to A_{\mathrm{loc}}$$
3. $$\int_b^x \phi^*\omega = \phi^* \int_b^x \omega \leftarrow \textit{(Frobenius equivariance)}$$

# Reduction to reduction

Balakrishnan-Bradshaw-Kedlaya reduce the problem of computing all Coleman integrals of basis differentials $\omega_i$ of $H^i$ between $\infty \in X$ and a point $x \in X(\mathbf{Q}_p)$, to:

1. Finding "tiny integrals" between nearby points,
2. Writing $\phi^*\omega_i - \mathrm{d}f_i = \sum_j a_j\omega_j$ and evaluating the primitive $f_i$ for a point $P$ near $x$, for each $i$.

# Kedlaya's algorithm

**Theorem (Kedlaya)**

*The action of $\phi^*$ on $H^1(X)$ (and hence the zeta function of $X$) can be computed in time*

$$\widetilde{O}(p).$$

**Theorem (Harvey)**

*If $p > (2g+1)(2N-1)$ the action of $\phi^*$ on $H^1(X)$ can be computed in time*

$$\widetilde{O}(\sqrt{p}).$$

# How do these work?

# Primitive technology

So we need to compute $f$ along with $\omega - \mathrm{d}f$.

For vanilla Kedlaya this is "easy", the reduction procedure is transparent, whenever we subtract $\mathrm{d}g$ to reduce, add $g$ onto $f$.

For faster variants, this is not so simple!

Harvey uses **horizontal** and **vertical** reductions to find the action of Frobenius on cohomology,

Harvey uses **horizontal** and **vertical** reductions to find the action of Frobenius on cohomology, abstractly we have:

Spaces of differentials $W_t$, indexed by degree, each of dimension $2g$.

**Goal**

Reduce all differentials from $W_t$ to a cohomologus one in $W_0$, write in terms of fixed basis of $W_0$.

Harvey uses **horizontal** and **vertical** reductions to find the action of Frobenius on cohomology, abstractly we have:

Spaces of differentials $W_t$, indexed by degree, each of dimension $2g$.

**Goal**

Reduce all differentials from $W_t$ to a cohomologus one in $W_0$, write in terms of fixed basis of $W_0$.

Relations in the de Rham cohomology $\rightsquigarrow$ linear maps $R(t)\colon W_t \to W_{t-1}\,\forall t$, with $R(t)\omega \sim \omega$.

Harvey uses **horizontal** and **vertical** reductions to find the action of Frobenius on cohomology, abstractly we have:

Spaces of differentials $W_t$, indexed by degree, each of dimension $2g$.

**Goal**

Reduce all differentials from $W_t$ to a cohomologus one in $W_0$, write in terms of fixed basis of $W_0$.

Relations in the de Rham cohomology $\rightsquigarrow$ linear maps $R(t)\colon W_t \to W_{t-1} \, \forall t$, with $R(t)\omega \sim \omega$. Want to find

$$W_t \ni \omega \mapsto R(1)R(2) \cdots R(t-1)R(t)\omega \in W_0$$

Harvey uses **horizontal** and **vertical** reductions to find the action of Frobenius on cohomology, abstractly we have:

Spaces of differentials $W_t$, indexed by degree, each of dimension $2g$.

**Goal**

Reduce all differentials from $W_t$ to a cohomologus one in $W_0$, write in terms of fixed basis of $W_0$.

Relations in the de Rham cohomology $\rightsquigarrow$ linear maps $R(t)\colon W_t \to W_{t-1}\,\forall t$, with $R(t)\omega \sim \omega$. Want to find

$$W_t \ni \omega \mapsto R(1)R(2)\cdots R(t-1)R(t)\omega \in W_0$$

**Key fact**

Entries of $R(t)$ are fractions of *linear* functions of $t$, with $\mathbf{Z}_p$ coefficients; work of Bostan-Gaudry-Schost (& Harvey) $\implies$ products can be interpolated
$R(a,b) = R(a+1)\cdots R(b) \rightsquigarrow R(a+1+t, b+t)$

This interpolation is what gives us a $\widetilde{O}(\sqrt{p})$ algorithm.

We also want an evaluation of the primitive $f$ for which $\omega - \mathrm{d}f_\omega = R(t)\omega$. Writing the primitives in terms of the basis we get

This interpolation is what gives us a $\widetilde{O}(\sqrt{p})$ algorithm.

We also want an evaluation of the primitive $f$ for which $\omega - \mathrm{d}f_\omega = R(t)\omega$. Writing the primitives in terms of the basis we get

**Vital remark**

We must use evaluations of primitives here, instead of trying to compute $f$ as a power series.

### Stumbling block

This is no longer linear in the index! You cannot apply BGS to this recurrence.

## Stumbling block

This is no longer linear in the index! You cannot apply BGS to this recurrence.

## Horner to the rescue

Instead of computing a series $\sum_{i=0}^{N} a_i x^i$ by computing sequentially

$$\left(\sum_{i=t}^{N} a_i x^i\right)_{t=N,N-1,\ldots,0}$$

We instead compute

$$((\cdots((a_N)x + a_{N-1})x + \cdots)x + a_0)$$

from the inside to the out. This *is* an iterated composition of linear functions, each of which is linear in the index $t$.

## Stumbling block

This is no longer linear in the index! You cannot apply BGS to this recurrence.

## Horner to the rescue

Instead of computing a series $\sum_{i=0}^{N} a_i x^i$ by computing sequentially

$$\left(\sum_{i=t}^{N} a_i x^i\right)_{t=N,N-1,\ldots,0}$$

We instead compute

$$((\cdots((a_N)x + a_{N-1})x + \cdots)x + a_0)$$

from the inside to the out. This *is* an iterated composition of linear functions, each of which is linear in the index $t$.

The evaluation is only correct at the end!

In matrix form we are augmenting the reduction matrices in the following sort of way:

$$\begin{array}{cc}
 & \begin{array}{cccc} x^{t-2g}\,\mathrm{d}x/y & \cdots & x^{t}\,\mathrm{d}x/y & f(P) \end{array} \\
\begin{array}{c} x^{t-2g}\,\mathrm{d}x/y \\ \vdots \\ x^{t-2g}\,\mathrm{d}x/y \\ f(P) \end{array} &
\left(\begin{array}{ccc|c}
(2t-1)r_{0,0} + 2s'_{0,0} & \cdots & (2t-1)r_{2g-1,0} + 2s'_{2g-1,0} & \\
\vdots & \ddots & \vdots & \\
(2t-1)r_{0,2g-1} + 2s'_{0,2g-1} & \cdots & (2t-1)r_{2g-1,2g-1} + 2s'_{2g-1,2g-1} & \\
\hline
-S_0(x) & \cdots & -S_{2g-1}(x) & y^{-2}D_V(t)
\end{array}\right)
\end{array}$$

so that we keep in memory a vector $v \in W_t \times \mathbf{Q}_p$.

# Many integrals simultaneously

We may wish to do this with multiple points in several residue disks. Instead of repeating the whole procedure (repeating computing the Frobenius matrix), augment with many points.

$$\begin{array}{cc}
 & \begin{array}{cccccc} x^{t-2g}\,\mathrm{d}x/y & \cdots & x^{t}\,\mathrm{d}x/y & f(P_1) & \cdots & f(P_L) \end{array} \\
\begin{array}{c} x^{t-2g}\,\mathrm{d}x/y \\ \vdots \\ x^{t-2g}\,\mathrm{d}x/y \\ f(P_1) \\ \vdots \\ f(P_L) \end{array} &
\left(\begin{array}{ccc|ccc}
(2t-1)r_{0,0}+2s'_{0,0} & \cdots & (2t-1)r_{2g-1,0}+2s'_{2g-1,0} & & & \\
\vdots & \ddots & \vdots & & & \\
(2t-1)r_{0,2g-1}+2s'_{0,2g-1} & \cdots & (2t-1)r_{2g-1,2g-1}+2s'_{2g-1,2g-1} & & & \\
\hline
-S_0(x(P_1)) & \cdots & -S_{2g-1}(x(P_1)) & y^{-2}(P_1)D_V(t) & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
-S_0(x(P_L)) & \cdots & -S_{2g-1}(x(P_L)) & 0 & \cdots & y(P_L)^{-2}D_V(t)
\end{array}\right)
\end{array}$$

# Many integrals simultaneously

We may wish to do this with multiple points in several residue disks. Instead of repeating the whole procedure (repeating computing the Frobenius matrix), augment with many points.

$$
\begin{array}{cc}
 & \begin{array}{cccccc} x^{t-2g}\,\mathrm{d}x/y & \cdots & x^{t}\,\mathrm{d}x/y & f(P_1) & \cdots & f(P_L) \end{array} \\
\begin{array}{c} x^{t-2g}\,\mathrm{d}x/y \\ \vdots \\ x^{t-2g}\,\mathrm{d}x/y \\ f(P_1) \\ \vdots \\ f(P_L) \end{array} &
\left(\begin{array}{ccc|ccc}
(2t-1)r_{0,0}+2s'_{0,0} & \cdots & (2t-1)r_{2g-1,0}+2s'_{2g-1,0} & & & \\
\vdots & \ddots & \vdots & & & \\
(2t-1)r_{0,2g-1}+2s'_{0,2g-1} & \cdots & (2t-1)r_{2g-1,2g-1}+2s'_{2g-1,2g-1} & & & \\
\hline
-S_0(x(P_1)) & \cdots & -S_{2g-1}(x(P_1)) & y^{-2}(P_1)D_V(t) & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
-S_0(x(P_L)) & \cdots & -S_{2g-1}(x(P_L)) & 0 & \cdots & y(P_L)^{-2}D_V(t)
\end{array}\right)
\end{array}
$$

**Note**

This matrix and products of it have the same fixed form, when running BGS we don't try and interpolate entries that are always $0 \rightsquigarrow$ better run time.

Core algorithm is potentially more general? Evaluation of primitives is “integration”, but maybe of a different type?

# Some amusing factoids

It is faster to evaluate the power series this way than to evaluate the power series you get from vanilla Kedlya, even when multiple points are needed computing the power series is no use.

# Thanks for listening!

Questions/comments?