

Neue Algorithmen zur Lösung der Prym-Green-Vermutung für Kurven von Geschlecht 24

Andreas Steenpaß

TU Kaiserslautern

5. Mai 2017

Aufbau

Prym-Green-Vermutung

Algorithmus und Implementierung

Ausblick

Prym-kanonische nodale Kurven

Beispiel

Sei C eine Prym-kanonische nodale Kurve von Geschlecht $g = 10$, gegeben durch ein homogenes Ideal $I \subseteq R := \mathbb{F}_p[x_0, \dots, x_8]$.

Prym-kanonische nodale Kurven

Beispiel

Sei C eine Prym-kanonische nodale Kurve von Geschlecht $g = 10$, gegeben durch ein homogenes Ideal $I \subseteq R := \mathbb{F}_p[x_0, \dots, x_8]$.

	0	1	2	3	4	5	6	7
0:	1	-	-	-	-	-	-	-
1:	-	18	55	75	54	20	3	-
2:	-	13	75	180	230	165	63	10
Σ :	1	31	130	255	284	185	66	10

Prym-kanonische nodale Kurven

Beispiel

Sei C eine Prym-kanonische nodale Kurve von Geschlecht $g = 10$, gegeben durch ein homogenes Ideal $I \subseteq R := \mathbb{F}_p[x_0, \dots, x_8]$.

	0	1	2	3	4	5	6	7
0:	1	-	-	-	-	-	-	-
1:	-	18	55	75	54	20	3	-
2:	-	13	75	180	230	165	63	10
Σ :	1	31	130	255	284	185	66	10

$$0 \leftarrow R/I \leftarrow R \leftarrow \begin{array}{c} R(-2)^{18} \\ \oplus \\ R(-3)^{13} \end{array} \leftarrow \begin{array}{c} R(-3)^{55} \\ \oplus \\ R(-4)^{75} \end{array} \leftarrow \begin{array}{c} R(-4)^{75} \\ \oplus \\ R(-5)^{180} \end{array} \leftarrow \dots$$


Prym-kanonische nodale Kurven

Beispiel

Sei C eine Prym-kanonische nodale Kurve von Geschlecht $g = 10$, gegeben durch ein homogenes Ideal $I \subseteq R := \mathbb{F}_p[x_0, \dots, x_8]$.

	0	1	2	3	4	5	6	7
0:	1	-	-	-	-	-	-	-
1:	-	18	55	75	54	20	3	-
2:	-	13	75	180	230	165	63	10
Σ :	1	31	130	255	284	185	66	10

$$0 \leftarrow R/I \leftarrow R \leftarrow \begin{array}{c} R(-2)^{18} \\ \oplus \\ R(-3)^{13} \end{array} \leftarrow \begin{array}{c} R(-3)^{55} \\ \oplus \\ R(-4)^{75} \end{array} \leftarrow \begin{array}{c} R(-4)^{75} \\ \oplus \\ R(-5)^{180} \end{array} \leftarrow \dots$$




Prym-kanonische nodale Kurven

Beispiel

Sei C eine Prym-kanonische nodale Kurve von Geschlecht $g = 10$, gegeben durch ein homogenes Ideal $I \subseteq R := \mathbb{F}_p[x_0, \dots, x_8]$.

	0	1	2	3	4	5	6	7
0:	1	-	-	-	-	-	-	-
1:	-	18	55	75	54	20	3	-
2:	-	13	75	180	230	165	63	10
Σ :	1	31	130	255	284	185	66	10

$$0 \leftarrow R/I \leftarrow R \leftarrow \begin{array}{c} R(-2)^{18} \\ \oplus \\ R(-3)^{13} \end{array} \leftarrow \begin{array}{c} R(-3)^{55} \\ \oplus \\ R(-4)^{75} \end{array} \leftarrow \begin{array}{c} R(-4)^{75} \\ \oplus \\ R(-5)^{180} \end{array} \leftarrow \dots$$



C ist Gorenstein.

Prym-kanonische nodale Kurven

Beispiel

Sei C eine Prym-kanonische nodale Kurve von Geschlecht $g = 10$, gegeben durch ein homogenes Ideal $I \subseteq R := \mathbb{F}_p[x_0, \dots, x_8]$.

	0	1	2	3	4	5	6	7
0:	1	-	-	-	-	-	-	-
1:	-	18	55	75	54	20	3	-
2:	-	13	75	180	230	165	63	10
Σ :	1	31	130	255	284	185	66	10

	0	1	2	3	4	5	6	7
0:	1	-	-	-	-	-	-	-
1:	-	18	42	-	-	-	-	-
2:	-	-	-	126	210	162	63	10
Σ :	1	18	42	126	210	162	63	10

C besitzt eine *reine Auflösung*.

Prym-Green-Vermutung

Prym-kanonische nodale Kurven von geradem Geschlecht besitzen eine reine Auflösung.

Prym-Green-Vermutung

Prym-kanonische nodale Kurven von geradem Geschlecht besitzen eine reine Auflösung.

Ausnahmen: $g = 8$,

	0	1	2	3	4	5
0:	1	-	-	-	-	-
1:	-	7	1	-	-	-
2:	-	1	35	56	35	8
Σ :	1	8	36	56	35	8

Prym-Green-Vermutung

Prym-kanonische nodale Kurven von geradem Geschlecht besitzen eine reine Auflösung.

Ausnahmen: $g = 8, 16,$

	0	1	2	3	4	5	6	7	8	...
0:	1	-	-	-	-	-	-	-	-	
1:	-	75	520	1755	3432	3575	1	-	-	
2:	-	-	-	-	-	1	6435	11440	11583	...
Σ :	1	75	520	1755	3432	3576	6436	11440	11583	...

Prym-Green-Vermutung

Prym-kanonische nodale Kurven von geradem Geschlecht besitzen eine reine Auflösung.

Ausnahmen: $g = 8, 16, ??$

Prym-Green-Vermutung

Prym-kanonische nodale Kurven von geradem Geschlecht besitzen eine reine Auflösung.

Ausnahmen: $g = 8, 16, ??$

Algorithmische Lösung:

- ▶ Auflösung berechnen
- ▶ Matrix reduzieren

Prym-Green-Vermutung

Prym-kanonische nodale Kurven von geradem Geschlecht besitzen eine reine Auflösung.

Ausnahmen: $g = 8, 16, ??$

Algorithmische Lösung:

- ▶ Auflösung berechnen
- ▶ Matrix reduzieren

Prym-Green-Vermutung

Prym-kanonische nodale Kurven von geradem Geschlecht besitzen eine reine Auflösung.

Ausnahmen: $g = 8, 16, ??$

Algorithmische Lösung:

- ▶ Auflösung berechnen
- ▶ Matrix reduzieren

Für $g = 24$ hat die Prym-Green-Matrix die Größe

$$3.250.026 \times 3.250.026 !$$

Aufbau

Prym-Green-Vermutung

Algorithmus und Implementierung

Ausblick

Algorithmus von Schreyer

Klassisch:

Die Nullreduktionen im Buchberger-Algorithmus liefern ein Erzeugendensystem des Syzygienmoduls.

Algorithmus von Schreyer

Klassisch:

Die Nullreduktionen im Buchberger-Algorithmus liefern ein Erzeugendensystem des Syzygienmoduls.

Notation

$$\begin{array}{c|ccc} x & -y & -z & 0 \\ y & x & 0 & -z \\ z & 0 & x & y \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\left. \begin{array}{l} w^2 - xz \\ wx - yz \\ x^2 - wy \\ xy - z^2 \\ y^2 - wz \end{array} \right|$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\begin{array}{l|l} w^2 - xz & -x \\ wx - yz & w \\ x^2 - wy & \\ xy - z^2 & \\ y^2 - wz & \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\begin{array}{l|l} w^2 - xz & -x \\ wx - yz & w \\ x^2 - wy & -z \\ xy - z^2 & \\ y^2 - wz & \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\begin{array}{l|l} \mathbf{w}^2 - xz & -\mathbf{x} \\ \mathbf{wx} - yz & \mathbf{w} \\ \mathbf{x}^2 - wy & -z \\ \mathbf{xy} - z^2 & 0 \\ \mathbf{y}^2 - wz & 0 \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\begin{array}{l|ll} \mathbf{w}^2 - xz & -x & \\ \mathbf{wx} - yz & \mathbf{w} & -x \\ \mathbf{x}^2 - wy & -z & \mathbf{w} \\ \mathbf{xy} - z^2 & 0 & \\ \mathbf{y}^2 - wz & 0 & \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\begin{array}{l|ll} w^2 - xz & -x & \\ wx - yz & w & -x \\ x^2 - wy & -z & w \\ xy - z^2 & 0 & -z \\ y^2 - wz & 0 & \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\begin{array}{l|ll} \mathbf{w}^2 - xz & -x & y \\ \mathbf{wx} - yz & \mathbf{w} & -\mathbf{x} \\ \mathbf{x}^2 - wy & -z & \mathbf{w} \\ \mathbf{xy} - z^2 & 0 & -z \\ \mathbf{y}^2 - wz & 0 & \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\begin{array}{l|ll} \mathbf{w}^2 - xz & -x & y \\ \mathbf{wx} - yz & \mathbf{w} & -\mathbf{x} \\ \mathbf{x}^2 - wy & -z & \mathbf{w} \\ \mathbf{xy} - z^2 & 0 & 0 \\ \mathbf{y}^2 - wz & 0 & \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$$\begin{array}{l|ll} \mathbf{w}^2 - xz & -x & y \\ \mathbf{wx} - yz & \mathbf{w} & -\mathbf{x} \\ \mathbf{x}^2 - wy & -z & \mathbf{w} \\ \mathbf{xy} - z^2 & 0 & 0 \\ \mathbf{y}^2 - wz & 0 & 0 \end{array}$$

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$w^2 - xz$	$-x$	y	0	$-z$	0	$-y^2 + wz$
$wx - yz$	w	$-x$	$-y$	0	z	z^2
$x^2 - wy$	$-z$	w	0	$-y$	0	0
$xy - z^2$	0	0	w	x	$-y$	$-yz$
$y^2 - wz$	0	0	$-z$	$-w$	x	w^2

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$w^2 - xz$	$-x$	y	0	$-z$	0	$-y^2 + wz$
$wx - yz$	w	$-x$	$-y$	0	z	z^2
$x^2 - wy$	$-z$	w	0	$-y$	0	0
$xy - z^2$	0	0	w	x	$-y$	$-yz$
$y^2 - wz$	0	0	$-z$	$-w$	x	w ²
<hr/>						
	0	y	$-x$	w	$-z$	1
	$-y^2 + wz$	z^2	$-wy$	yz	$-w^2$	x

Algorithmus von Schreyer

Beispiel

$R = \mathbb{Q}[w, x, y, z]$ mit der graduiert rückwärts-lexikogr. Ordnung

$w^2 - xz$	$-x$	y	0	$-z$	0	$-y^2 + wz$
$wx - yz$	w	$-x$	$-y$	0	z	z^2
$x^2 - wy$	$-z$	w	0	$-y$	0	0
$xy - z^2$	0	0	w	x	$-y$	$-yz$
$y^2 - wz$	0	0	$-z$	$-w$	x	w^2
<hr/>						
	0	y	$-x$	w	$-z$	1
	$-y^2 + wz$	z^2	$-wy$	yz	$-w^2$	x

Wir erhalten also

$$0 \leftarrow R/I \leftarrow R \leftarrow R(-2)^5 \leftarrow \begin{matrix} R(-3)^5 \\ \oplus \\ R(-4) \end{matrix} \leftarrow \begin{matrix} R(-4) \\ \oplus \\ R(-5) \end{matrix} \leftarrow 0$$

Algorithmus von Schreyer

Verbesserungen für den Schreyer-Algorithmus:

1. Terme niederer Ordnung können weggelassen werden.

Algorithmus von Schreyer

Verbesserungen für den Schreyer-Algorithmus:

1. Terme niederer Ordnung können weggelassen werden.
2. Die Terme müssen nicht geordnet werden.

Algorithmus von Schreyer

Verbesserungen für den Schreyer-Algorithmus:

1. Terme niederer Ordnung können weggelassen werden.
2. Die Terme müssen nicht geordnet werden.
3. Manche Teilergebnisse können wiederverwendet werden (caching).

Implementierung

Timings für die Berechnung einer freien Auflösung
einer Prym-kanonischen nodalen Kurve mit $g = 16$:

- ▶ Eröcal, Motsak, Schreyer, S.: Refined Algorithms
to Compute Syzygies (JSC 2016):

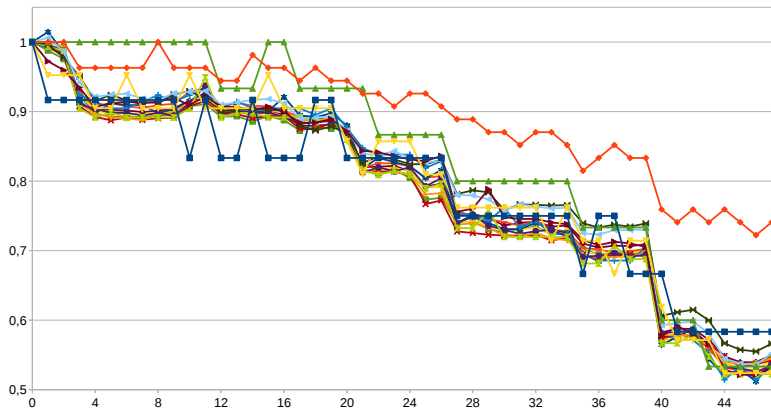
18.612 s

Implementierung

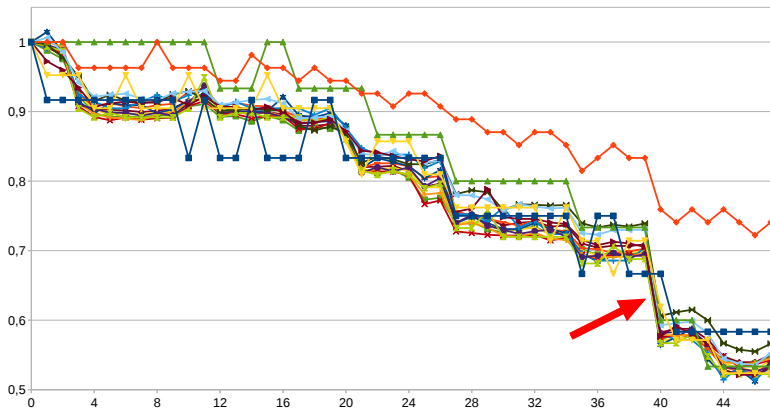
Timings für die Berechnung einer freien Auflösung
einer Prym-kanonischen nodalen Kurve mit $g = 16$:

- ▶ Eröcal, Motsak, Schreyer, S.: Refined Algorithms
to Compute Syzygies (JSC 2016): 18.612 s
- ▶ mit einer einzeiligen (!) Änderung: 114 s

Code-Restrukturierung



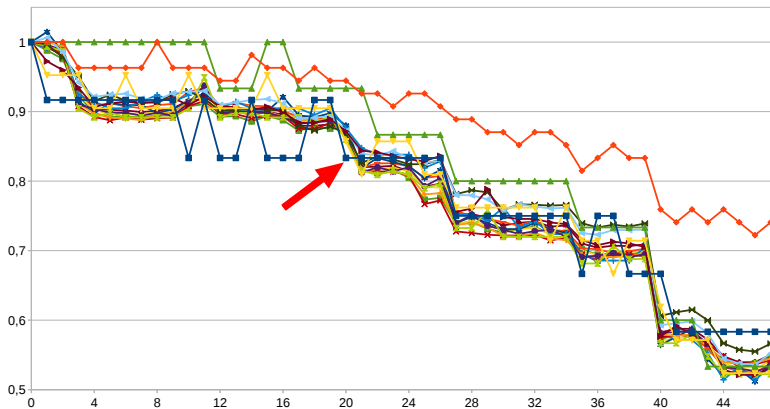
Code-Restrukturierung



in der richtigen Reihenfolge nach Reduzierern suchen

Verbesserung: 0.127

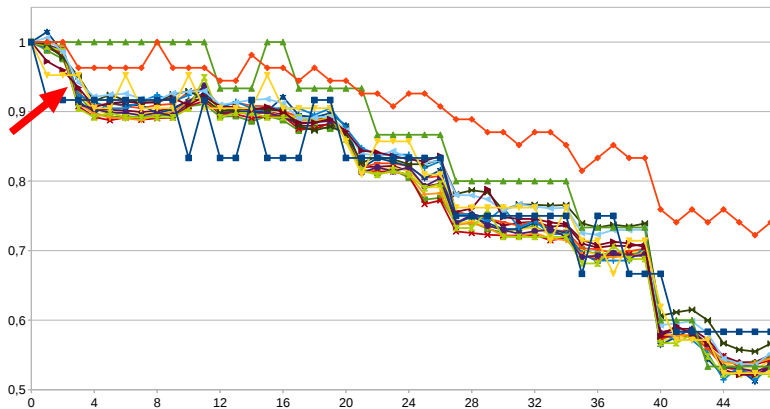
Code-Restrukturierung



unnötige Checks auslassen

Verbesserung: 0.053

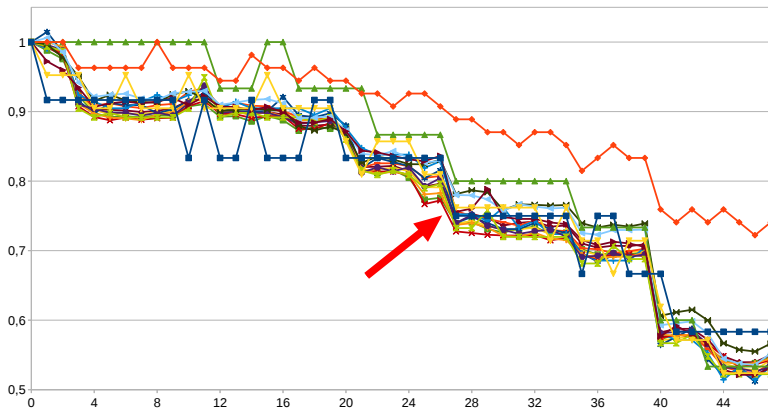
Code-Restrukturierung



C++-Klassen entfernen

Verbesserung: 0.052

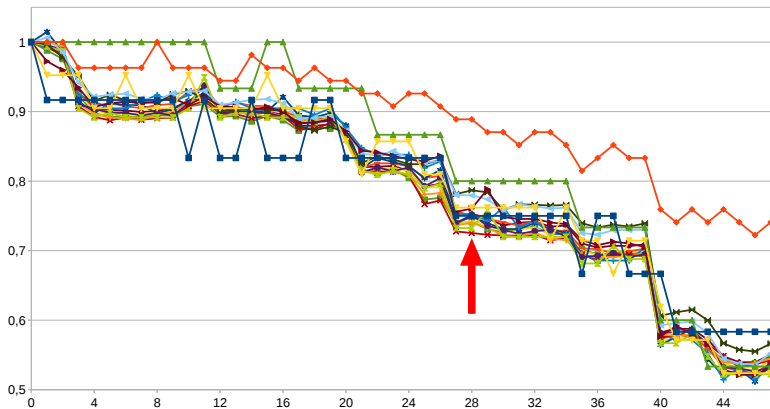
Code-Restrukturierung



die Terme am Ende der Rechnung geschickter sortieren

Verbesserung: 0.046

Code-Restrukturierung



die Funktionen im Quellcode umsortieren

Verschlechterung: 0.0026

Berechnung der Pym-Green-Matrix

$$I = \underbrace{\langle g_1, \dots, g_s \rangle}_{\text{GB}} \subset R := K[x_1, \dots, x_n] \quad \text{und}$$

$$x_n \nmid \text{LT}(g_j) \quad \forall j = 1, \dots, s$$

Berechnung der Pryn-Green-Matrix

$$I = \underbrace{\langle g_1, \dots, g_s \rangle}_{\text{GB}} \subset R := K[x_1, \dots, x_n] \quad \text{und}$$

$$x_n \nmid \text{LT}(g_j) \quad \forall j = 1, \dots, s$$

$\Rightarrow x_n$ ist ein Nicht-Nullteiler in R/I .

Berechnung der Pryn-Green-Matrix

$$I = \underbrace{\langle g_1, \dots, g_s \rangle}_{\text{GB}} \subset R := K[x_1, \dots, x_n] \quad \text{und}$$

$$x_n \nmid \text{LT}(g_j) \quad \forall j = 1, \dots, s$$

$\Rightarrow x_n$ ist ein Nicht-Nullteiler in R/I .

$\Rightarrow (0 \longleftarrow R/I \longleftarrow R \longleftarrow R^{r_1} \longleftarrow \dots) \otimes_R R/\langle x_n \rangle$ ist exakt.

Berechnung der Prym-Green-Matrix

$$I = \underbrace{\langle g_1, \dots, g_s \rangle}_{\text{GB}} \subset R := K[x_1, \dots, x_n] \quad \text{und}$$

$$x_n \nmid \text{LT}(g_j) \quad \forall j = 1, \dots, s$$

$\Rightarrow x_n$ ist ein Nicht-Nullteiler in R/I .

$\Rightarrow (0 \longleftarrow R/I \longleftarrow R \longleftarrow R^{r_1} \longleftarrow \dots) \otimes_R R/\langle x_n \rangle$ ist exakt.

Verbesserung für $g = 16$:

62.3 s \longrightarrow 12.4 s

Berechnung der Prym-Green-Matrix

$$I = \underbrace{\langle g_1, \dots, g_s \rangle}_{\text{GB}} \subset R := K[x_1, \dots, x_n] \quad \text{und}$$

$$x_n \nmid \text{LT}(g_j) \quad \forall j = 1, \dots, s$$

$\Rightarrow x_n$ ist ein Nicht-Nullteiler in R/I .

$\Rightarrow (0 \longleftarrow R/I \longleftarrow R \longleftarrow R^{r_1} \longleftarrow \dots) \otimes_R R/\langle x_n \rangle$ ist exakt.

Verbesserung für $g = 16$:

62.3 s \longrightarrow 12.4 s

Zur Erinnerung:

Eröcal, Motsak, Schreyer, S.: Refined Algorithms
to Compute Syzygies (JSC 2016):

18.612 s

Aufbau

Prym-Green-Vermutung

Algorithmus und Implementierung


Ausblick

Projektstatus

Algorithmische Lösung:

Projektstatus

Algorithmische Lösung:

- ▶ Auflösung berechnen 

Projektstatus

Algorithmische Lösung:

- ▶ Auflösung berechnen ✓
- ▶ Matrix reduzieren ✗

Projektstatus

Algorithmische Lösung:

- ▶ Auflösung berechnen ✓
- ▶ Matrix abspeichern ✗
- ▶ Matrix reduzieren ✗

Prym-Green-Matrix für $g = 14$ (1932×1932)

