Integration-by-parts reduction via algebraic geometry method

Yang Zhang

Johannes Gutenberg University Mainz
ETH Zurich

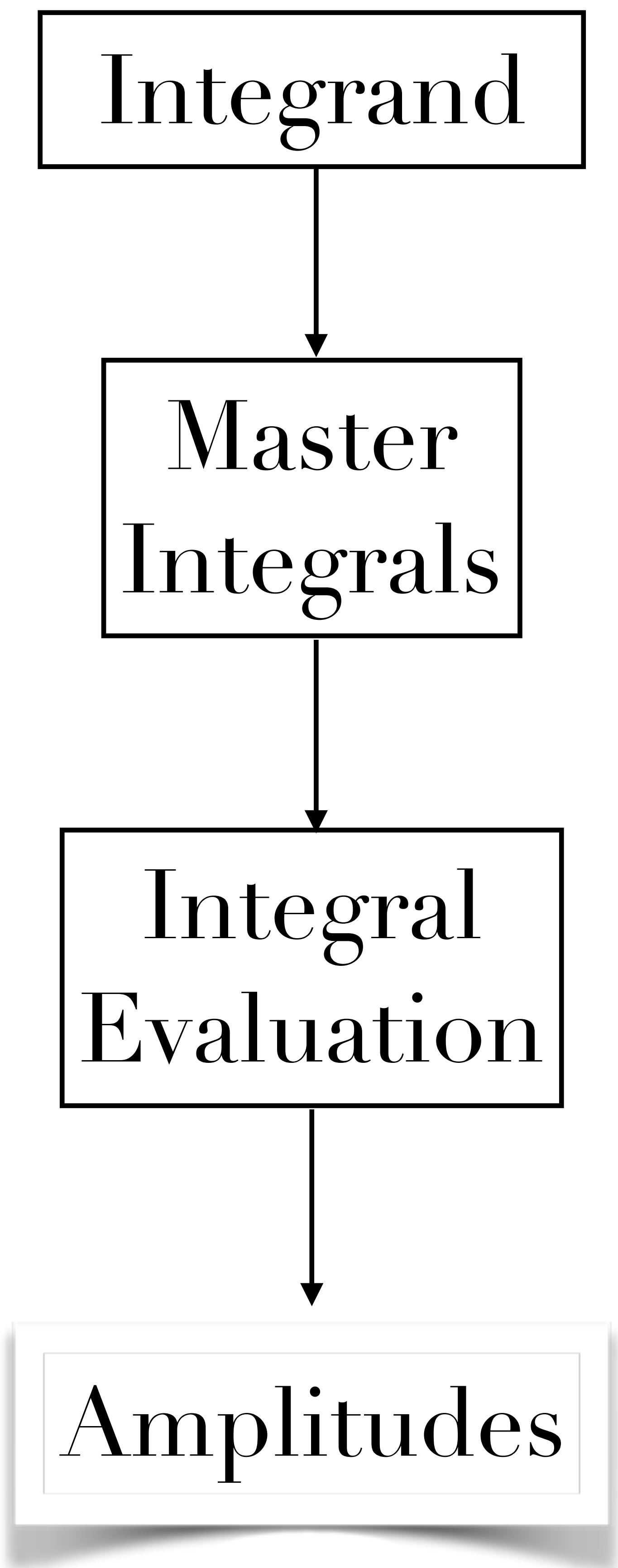
1805.01873

with
Janko Boehm, Hans Schoenemann (Kaiserslautern)
Alessandro Georgoudis (Uppsala), Kasper Larsen (Southampton)

FNSNF
FONDS NATIONAL SUISSE
SCHWEIZERISCHER NATIONALFONDS
FONDO NAZIONALE SVIZZERO
SWISS NATIONAL SCIENCE FOUNDATION

Loopfest 2018
Michigan State University
July 20, 2018

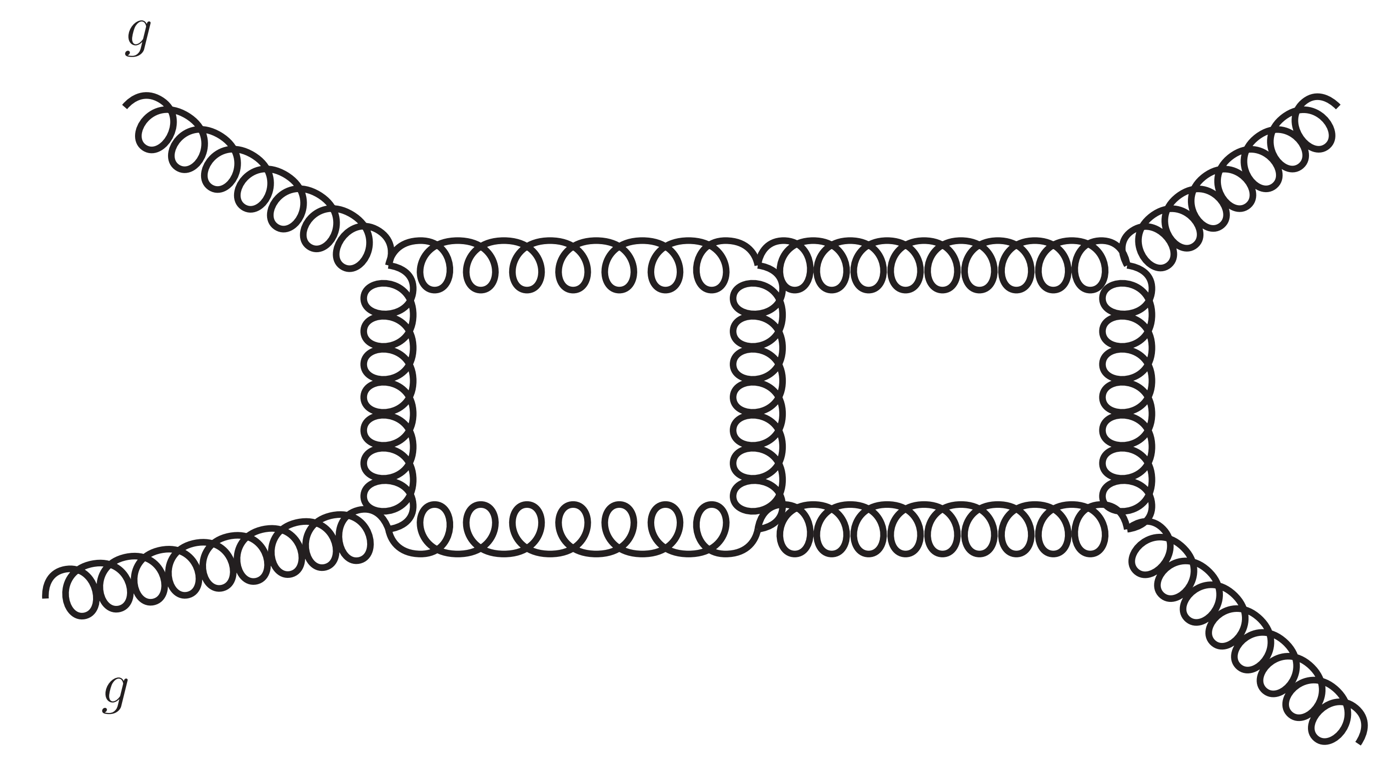
Integration-by-Parts (IBP) reduction



Feynman Rules,
Generalized Unitarity

Integration-by-parts (IBP)

Numeric: Mellin-Barnes, SecDec …
Analytic: Differential Equation, Dimensional recursion, Bootstrap …



$$\int \frac{d^D l_1}{i\pi^{D/2}} \cdots \int \frac{d^D l_L}{i\pi^{D/2}} \frac{1}{D_1^{\alpha_1} \ldots D_k^{\alpha_k}}, \quad \alpha_i \in \mathbb{Z}$$

Integration-by-parts reduction can be a bottleneck of high energy physics computations

Integration-by-Parts (IBP) reduction

$$\int \frac{d^D l_1}{i\pi^{D/2}} \dots \int \frac{d^D l_L}{i\pi^{D/2}} \frac{\partial}{\partial l_i^\mu} \frac{v_i^\mu}{D_1^{\alpha_1} \dots D_k^{\alpha_k}} = 0$$

IBP: Chetyrkin, Tkachov 1981
Laporta 2001

state-of-art IBP programs

FIRE (Smirnov)
Reduze (von Manteuffel, Studerus)
LiteRed (Lee)
Kira (Maierhofer, Usovitsch, Uwer)

and private codes with new ideas

Schabinger 2011
Manteuffel, Schabinger 2015
Bern, Enciso, Ita, Zeng 2017
Chawdhry, Lim, Mitov 2018
...

Our approach

Module intersection	to put constraints on the integral space (no-double-propagator/ unitarity cuts) and trim linear system from IBPs dramatically
Localization	Simplify the multi-parameter algebra computation

Based on mathematical ideas and experiences from computational algebraic geometry

Integration-by-Parts (IBP) reduction integral selection

Smaller linear system, fewer integrals

Gluza, Kajda, Kosower 2010
Roman Lee 2014

Integrals without "doubled propagators"

$$\int \frac{d^D l_1}{i\pi^{D/2}} \cdots \int \frac{d^D l_L}{i\pi^{D/2}} \frac{1}{D_1^{\alpha_1} \ldots D_m^{\alpha_m} D_{m+1}^{\alpha_{m+1}} \ldots D_k^{\alpha_k}}, \quad \begin{cases} \alpha_i \leq 1, & 1 \leq i \leq m \\ \alpha_i \leq 0, & m < i \leq k \end{cases}$$

Small subset of integrals

$$\int \frac{d^D l_1}{i\pi^{D/2}} \cdots \int \frac{d^D l_L}{i\pi^{D/2}} \frac{\partial}{\partial l_i^\mu} \frac{v_i^\mu D_{m+1}^{-\alpha_{m+1}} \ldots D_k^{-\alpha_k}}{D_1 \ldots D_m} = 0$$

Require $\boxed{v_i^\mu \frac{\partial}{\partial l_i^\mu} D_j \propto D_j}$

not a linear constraint, but a syzygy constraint in algebraic geometry

standard Schreyer's algorithm is not very practical for solve these constraint, new ingredients needed …

Baikov representation

$$\int \frac{d^D l_1}{i\pi^{D/2}} \dots \int \frac{d^D l_L}{i\pi^{D/2}} \frac{1}{D_1^{\alpha_1} \dots D_k^{\alpha_k}} = C(s_{ij}, L, E, D) \int dz_1 \dots dz_k \frac{F^{\frac{D-L-E-1}{2}}}{z_1^{\alpha_1} \dots z_k^{\alpha_k}}$$ Baikov 1994

$F \equiv \det(S)$ S is the Gram matrix for external and internal momenta

- Linear propagators
- Automate integrand reduction (vs. OPP / Groebner basis methods)
- Automate unitarity cuts
- Convenient for reduction problems and deriving differential equation

Baikov representation, IBP without "doubled propagators"

$$\int \frac{d^D l_1}{i\pi^{D/2}} \cdots \int \frac{d^D l_L}{i\pi^{D/2}} \frac{1}{D_1^{\alpha_1} \ldots D_m^{\alpha_m} D_{m+1}^{\alpha_{m+1}} \ldots D_k^{\alpha_k}}, \quad \begin{cases} \alpha_i \leq 1, & 1 \leq i \leq m \\ \alpha_i \leq 0, & m < i \leq k \end{cases}$$

Ita 2015
Larsen and YZ, 2015

Just consider IBPs

$$0 = \left(\prod_{i=1}^{k} \int dz_i \right) \sum_{j=1}^{k} \frac{\partial}{\partial z_j} \left(a_j(z) \det(S)^{\frac{D-L-E-1}{2}} \frac{1}{z_1 \ldots z_m} \right)$$

Polynomials!

Require

1. no shifted exponent: $\sum_{j=1}^{k} a_j(z) \frac{\partial F}{\partial z_j} + \beta(z) F = 0$ These $(a_1(z), \ldots a_k(z))$ form a module $M_1 \subset R^k$.

2. no doubled propagator: $a_i(z) \in \langle z_i \rangle, \quad 1 \leq i \leq m$ These $(a_1(z), \ldots a_k(z))$ form a module $M_2 \subset R^k$.

Both M_1 and M_2 are pretty simple ...

YZ 2016

$$\boxed{M_1 \cap M_2}$$

Intersection of two modules

Computation of the first module M_1

$$\sum_{j=1}^{k} a_j(z)\frac{\partial F}{\partial z_j} + \beta(z)F = 0$$

- syzygy for the $\{\frac{\partial F}{\partial z_1}, \ldots, \frac{\partial F}{\partial z_k}, F\}$ Müller 1993
- $\mathrm{Ann}(F^s)$, annihilator of F^s in Weyl algebra.

If F is a determinant matrix whose elements are free variables, this kind of syzygy module is simple.

$$A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{pmatrix}$$

Roman Lee's idea

No computation is needed

syzygy generators (Laplace expansion)

$$\sum_j a_{k,j}\frac{\partial(\det A)}{\partial a_{i,j}} - \delta_{k,i}\cdot \det A = 0$$

provides all syzygy generators

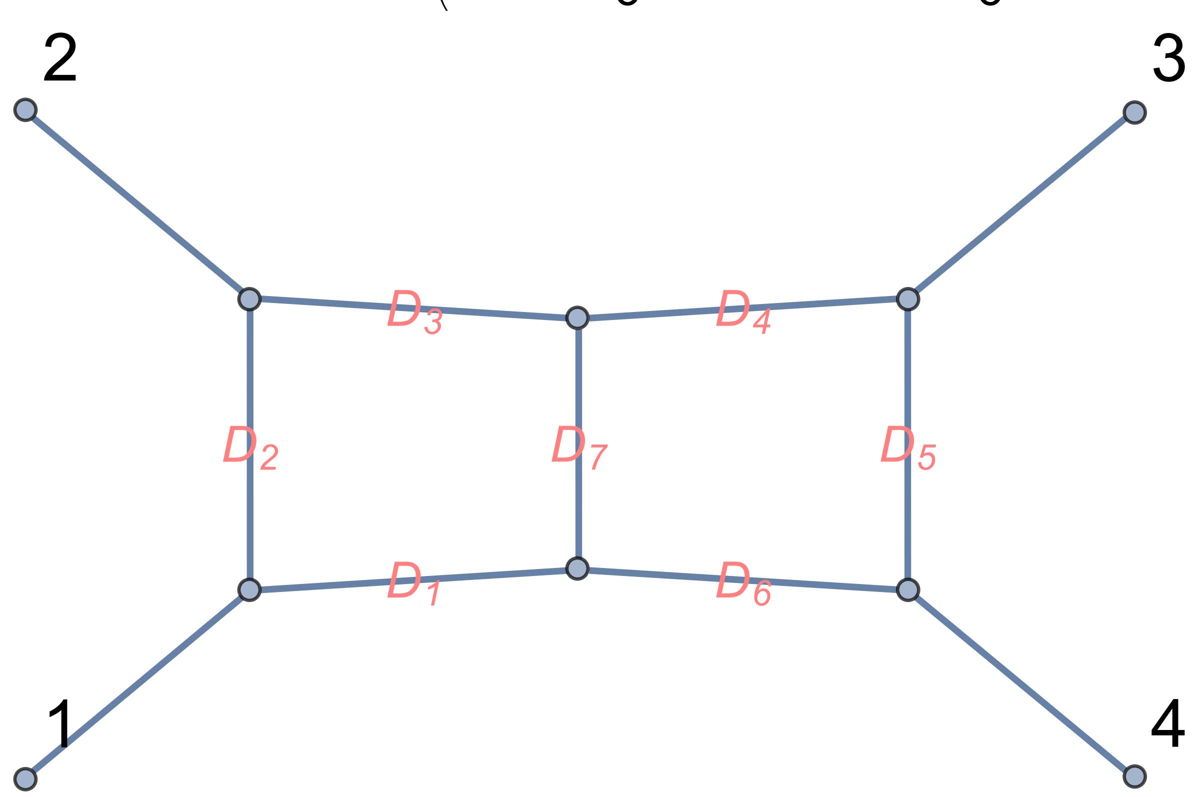
Completeness proved by Gulliksen–Negard and Jozefiak exact sequences

Boehm, Georgoudis, Larsen, Schulze, YZ 2017

Example, massless double box

$\mathbb{Q}(s,t)[z_1,\ldots z_9]$: 2 parameters, 9 variables (Each row is a module generator)

$$M_1 = \begin{pmatrix} z_1-z_2 & z_1-z_2 & -s+z_1-z_2 & 0 & 0 & 0 & z_1-z_2-z_6+z_9 & t+z_1-z_2 & 0 \\ 0 & 0 & 0 & s-z_6+z_9 & -t-z_6+z_9 & -z_6+z_9 & z_1-z_2-z_6+z_9 & 0 & -z_6+z_9 \\ s+z_2-z_3 & z_2-z_3 & z_2-z_3 & 0 & 0 & 0 & z_2-z_3+z_4-z_9 & -t+z_2-z_3 & 0 \\ 0 & 0 & 0 & z_4-z_9 & t+z_4-z_9 & -s+z_4-z_9 & z_2-z_3+z_4-z_9 & 0 & z_4-z_9 \\ -z_1+z_8 & -t-z_1+z_8 & s-z_1+z_8 & 0 & 0 & 0 & -z_1-z_5+z_6+z_8 & -z_1+z_8 & 0 \\ 0 & 0 & 0 & -s-z_5+z_6 & -z_5+z_6 & -z_5+z_6 & -z_1-z_5+z_6+z_8 & 0 & t-z_5+z_6 \\ 2z_1 & z_1+z_2 & -s+z_1+z_3 & 0 & 0 & 0 & z_1-z_6+z_7 & z_1+z_8 & 0 \\ 0 & 0 & 0 & s-z_3-z_6+z_7 & -z_6+z_7-z_8 & -z_1-z_6+z_7 & z_1-z_6+z_7 & 0 & -z_2-z_6+z_7 \\ -z_1-z_6+z_7 & -z_1+z_7-z_9 & s-z_1-z_4+z_7 & 0 & 0 & 0 & -z_1+z_6+z_7 & -z_1-z_5+z_7 & 0 \\ 0 & 0 & 0 & -s+z_4+z_6 & z_5+z_6 & 2z_6 & -z_1+z_6+z_7 & 0 & z_6+z_9 \end{pmatrix}$$



$$M_2 = \begin{pmatrix} z_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & z_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & z_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & z_4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & z_5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & z_6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & z_7 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$M_1 \cap M_2$ (even without cut) is computed in ~4 seconds, with **Singular** 4.1.1, much faster than Schreyer algorithm for syzygies

Singular: an open source algebra system written in C++, also is a C++ library

However, for research problems, more ingredients are needed …

Localization trick

Analytic Mandelstam variables and masses (parameters) slow down the computation

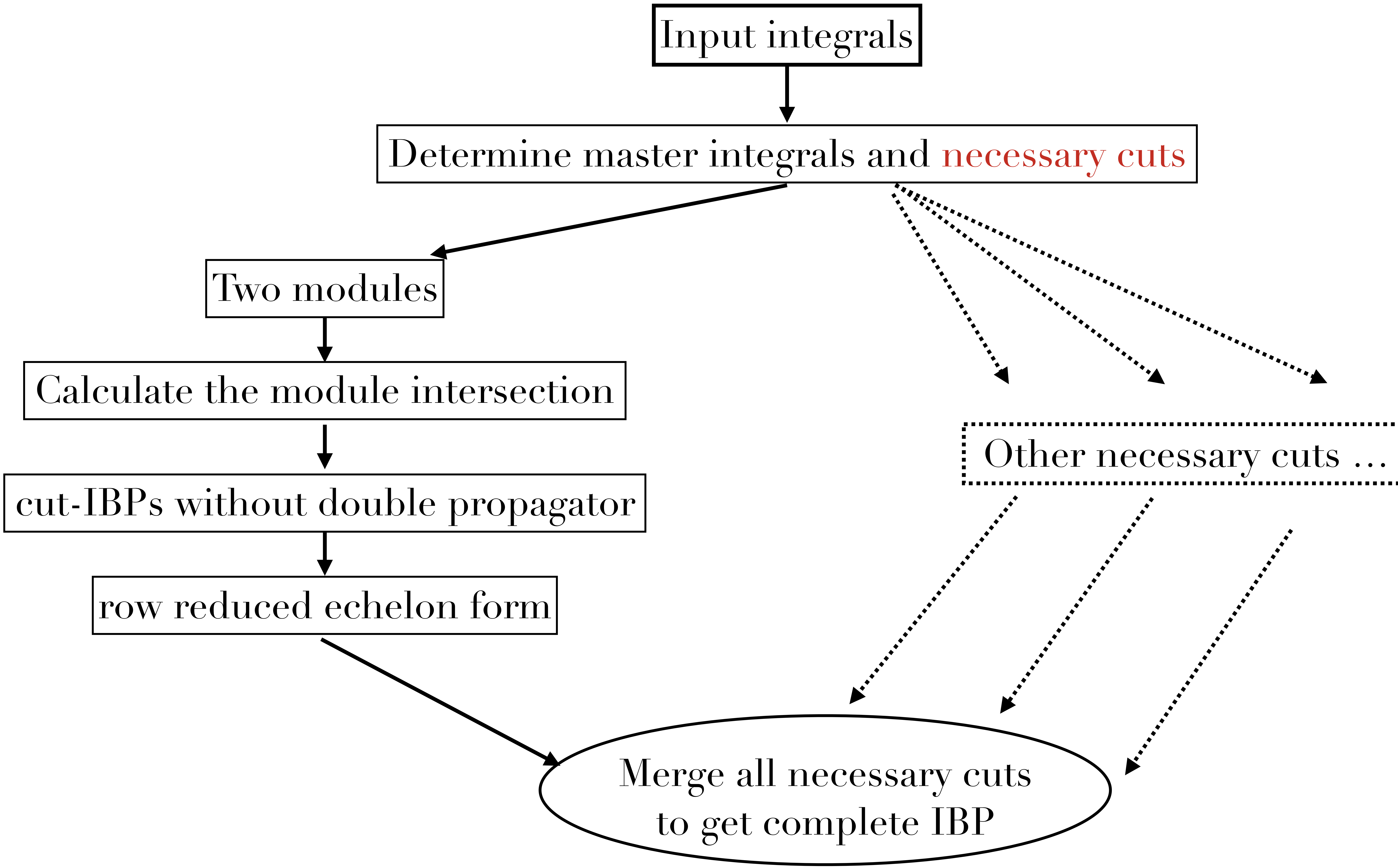
idea from primary decomposition algorithm, Gianni, Trager, Zacharias 1988

Treat parameters as variables, and compute in a block monomial ordering

[variables] > [parameters]

this trick makes all polynomials homogeneous
and avoids complicated computations with fractions of parameters

Our algorithm



Non-trivial example

Nonplanar hexagon-box

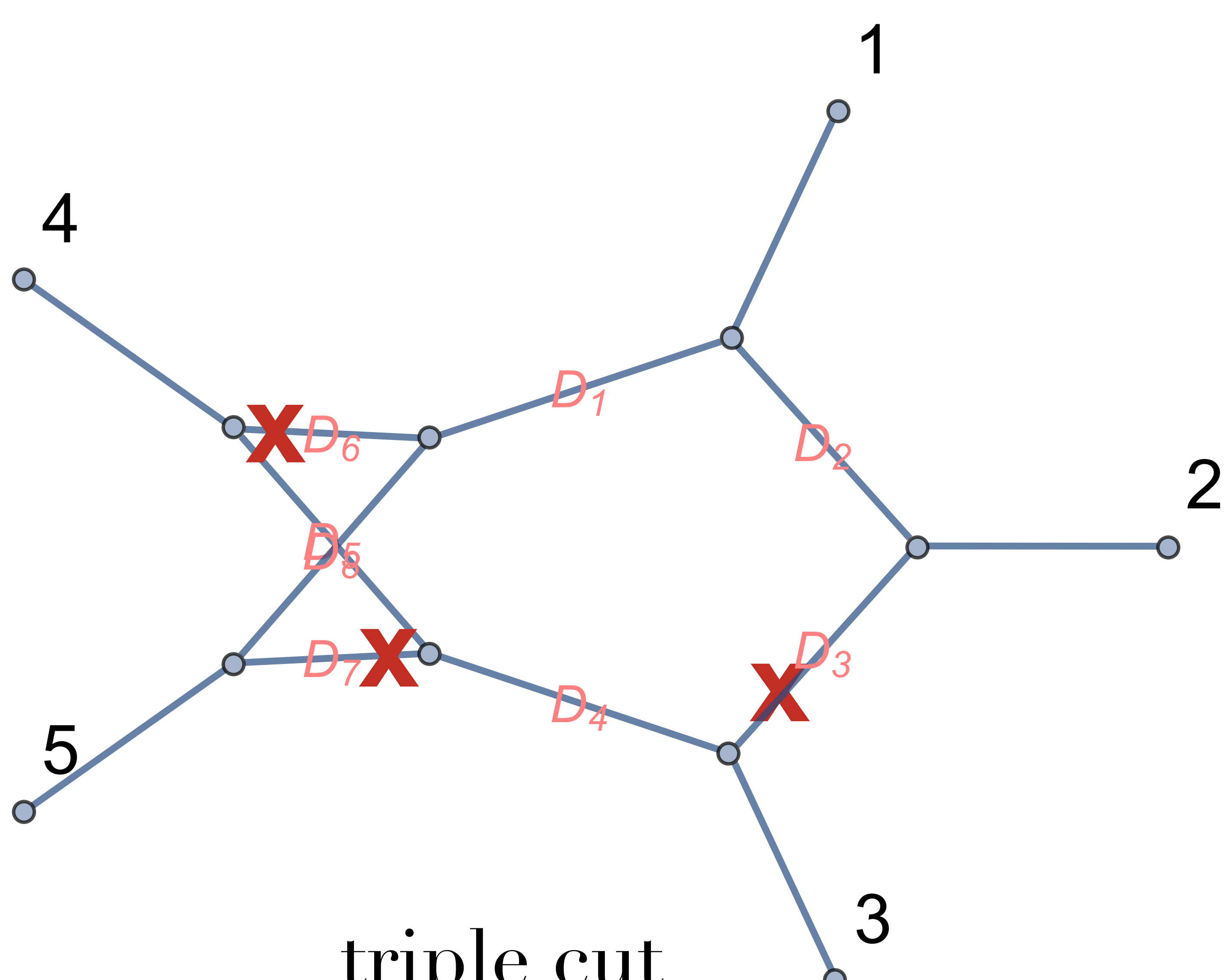
(3,6,7) cut

10 necessary cuts needed (8 triple cuts, 2 quadruple cuts)

$\mathbb{Q}(s_{12}, s_{13}, s_{14}, s_{23}, s_{24})[z_1, z_2, z_4, z_5, z_8, z_9, z_{10}, z_{11}]$: 5 parameters, 11-3=8 variables

$z_3 \to 0, z_6 \to 0, z_7 \to 0$

$$M_1' = \begin{pmatrix}
z_1-z_2 & z_1-z_2 & -s12+z_1-z_2 & -s12-s13+z_1-z_2 & s14+z_1-z_2-z_8+z_{10} & z_1-z_2-z_8+z_{10} & 0 & 0 & -s12-s13-s14+z_1-z_2 & 0 & 0 \\
0 & 0 & 0 & 0 & s14+z_1-z_2-z_8+z_{10} & z_1-z_2-z_8+z_{10} & s12+s13+s14-z_8+z_{10} & -z_8+z_{10} & 0 & -z_8+z_{10} & s12-z_8+z_{10} \\
s12+z_2 & z_2 & z_2 & -s23+z_2 & s12+s24+z_2-z_8+z_{11} & s12+z_2-z_8+z_{11} & 0 & 0 & -s23-s24+z_2 & 0 & 0 \\
0 & 0 & 0 & 0 & s12+s24+z_2-z_8+z_{11} & s12+z_2-z_8+z_{11} & s12+s23+s24-z_8+z_{11} & -z_8+z_{11} & 0 & s12-z_8+z_{11} & -z_8+z_{11} \\
s13+s23-z_4 & s23-z_4 & -z_4 & -z_4 & -2\,s12-s13-s14-s23-s24-z_5+z_8-z_9-z_{10}-z_{11} & -s12-z_5+z_8-z_9-z_{10}-z_{11} & 0 & 0 & s12+s13+s14+s23+s24-z_4 & 0 & 0 \\
0 & 0 & 0 & 0 & -2\,s12-s13-s14-s23-s24-z_5+z_8-z_9-z_{10}-z_{11} & -s12-z_5+z_8-z_9-z_{10}-z_{11} & -2\,s12-s13-s14-s23-s24+z_4-z_5+z_8-z_9-z_{10}-z_{11} & -s12-s13-s23+z_4-z_5+z_8-z_9-z_{10}-z_{11} & 0 & -s12-s23+z_4-z_5+z_8-z_9-z_{10}-z_{11} & -s12-s13+z_4-z_5+z_8-z_9-z_{10}-z_{11} \\
-s12-s13-s23+z_4-z_9 & -s12-s13-s14-s23+z_4-z_9 & -s12-s13-s14-s23-s24+z_4-z_9 & z_4-z_9 & z_5 & z_5 & 0 & 0 & z_4-z_9 & 0 & 0 \\
0 & 0 & 0 & 0 & z_5 & z_5 & -z_4+z_5+z_9 & s12+s13+s23-z_4+z_5+z_9 & 0 & s12+s13+s14+s23-z_4+z_5+z_9 & s12+s13+s23+s24-z_4+z_5+z_9 \\
2\,z_1 & z_1+z_2 & -s12+z_1 & -s12-s13-s23+z_1+z_4 & -s12-s13-s23+z_1+z_4-z_8-z_9 & z_1-z_8 & 0 & 0 & z_1+z_9 & 0 & 0 \\
0 & 0 & 0 & 0 & -s12-s13-s23+z_1+z_4-z_8-z_9 & z_1-z_8 & -z_8-z_9 & -z_1-z_8 & 0 & -z_2-z_8 & s12-z_1+z_2-z_8 \\
-z_1-z_8 & -z_1-z_{10} & -z_1+z_8-z_{10}-z_{11} & s12+s13+s23-z_1-z_4+z_5+z_9 & s12+s13+s23-z_1-z_4+z_5+z_8+z_9 & -z_1+z_8 & 0 & 0 & -z_1 & 0 & 0 \\
0 & 0 & 0 & 0 & s12+s13+s23-z_1-z_4+z_5+z_8+z_9 & -z_1+z_8 & z_8 & 2\,z_8 & 0 & z_8+z_{10} & z_8+z_{11}
\end{pmatrix}$$



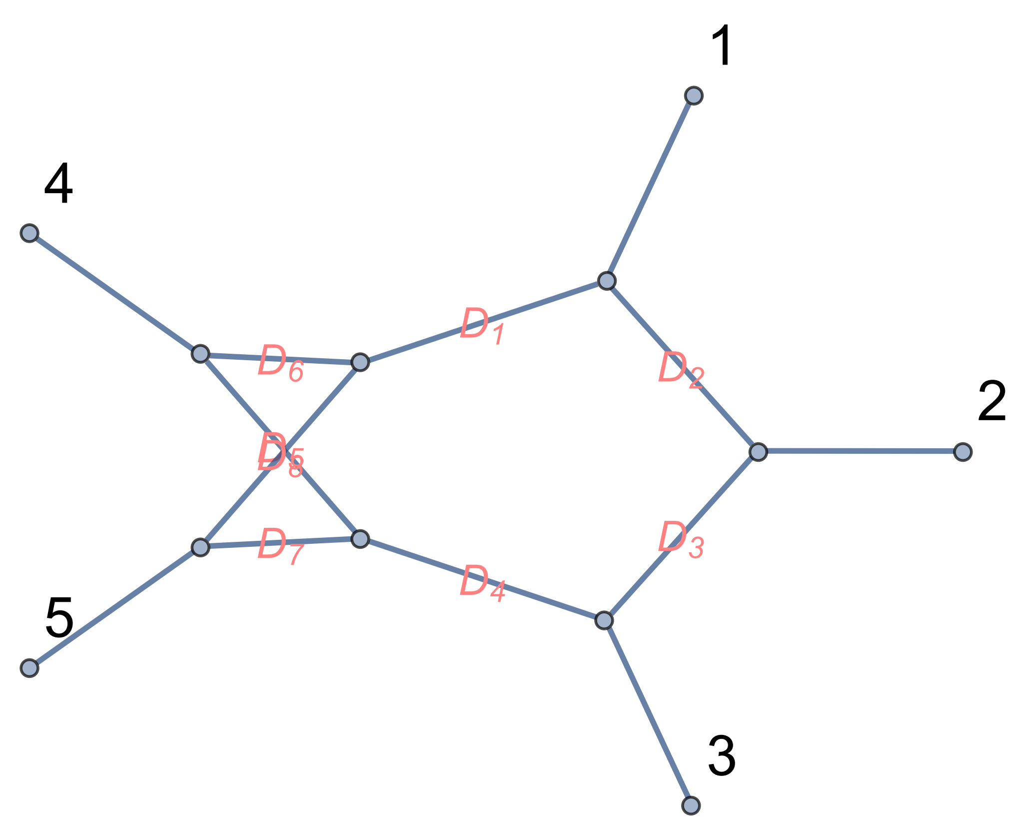
triple cut

$$M_2' = \begin{pmatrix}
z_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & z_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & z_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & z_5 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & z_8 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

$$M_1' \cap M_2' = ?$$

Localization trick

Boehm, Schoeneman, University of Kaiserslautern



Module intersection for (1,4,6,7)

$\mathbb{Q}(s_{12}, s_{13}, s_{14}, s_{23}, s_{24})[z_2, z_3, z_5, z_8, z_9, z_{10}, z_{11}]$ with
$z_2 > z_3 > z_5 > z_8 > z_9 > z_{10} > z_{11}$

? (does not finish)

$\mathbb{Q}[z_2, z_3, z_5, z_8, z_9, z_{10}, z_{11}, s_{12}, s_{13}, s_{14}, s_{23}, s_{24}]$ with
$[z_2, z_3, z_5, z_8, z_9, z_{10}, z_{11}] > [s_{12}, s_{13}, s_{14}, s_{23}, s_{24}]$

minutes with **Singular** on a laptop, one core

It is not a good idea to take $s_{12} \to 1$!

All module intersections for 10 necessary cuts found analytically

Module intersections to get algebraic constraint

Hexagon-box

efficient implement in **Singular** by Boehm

cuts	timing (sec)	RAM (GB)	generator size (raw, MB)	generator size (trimmed, MB)
1,5,7	218	4.3	68	10
2,5,7	43	1.1	25	1.4
2,5,8	303	6.7	49	3.1
2,6,7	743	9.8	100	2.8
3,5,8	404	7.4	97	3.7
3,6,7	699	11	80	3.6
3,6,8	24	1	10	1.6
4,6,8	797	13.7	21	1.6
1,4,5,8	53	1.7	4.4	3.6
1,4,6,7	196	3.0	9.4	4.1

Heuristic generator trimming algorithm, YZ

trimmed IBPs

Hexagon-box, reduce all rank-4 numerators to master integrals

cuts	# IBPs	# Integrals	Bytes (MB)	Density
1,5,7	1144	1177	1.2	1.4%
2,5,7	1170	1210	0.99	1.3%
2,5,8	1152	1190	1.1	1.5%
2,6,7	1118	1155	1.0	1.5%
3,5,8	1160	1202	1.2	1.5%
3,6,7	1173	1217	1.3	1.7%
3,6,8	1135	1176	0.77	1.2%
4,6,8	1140	1176	0.94	1.2%
1,4,5,8	700	723	0.69	1.7%
1,4,6,7	683	706	0.66	1.6%

Heuristic linear trimming algorithm, YZ powered by SpaSM package (Bouillaguet)

- remove linearly dependent IBPs
- remove IBPs irrelevant to targets
- prefer IBPs with small byte counts

Linear Reduction

- Sparse RREF (row reduced echelon form) with weighted Markowitz pivoting strategy
- Finite field numeric zero guessing
- Integral basis change
- different Mandelstam variable choices for different cuts
- Interpolation if necessary:

  heuristic multivariate separate numerator/denominator interpolation algorithm

Many of these ideas have been implemented in modern IBP solvers

Preliminary Mathematica code, YZ

Efficient implement in **Singular** by Bendle and Boehm to be available

Linear Reduction

Hexagon-box, reduce all rank-4 numerators to master integrals

cuts	# IBPs	# Integrals
1,5,7	1144	1177
2,5,7	1170	1210
2,5,8	1152	1190
2,6,7	1118	1155
3,5,8	1160	1202
3,6,7	1173	1217
3,6,8	1135	1176
4,6,8	1140	1176
1,4,5,8	700	723
1,4,6,7	683	706

Preliminary Mathematica code, YZ

bivariate numerator/denominator interpolation

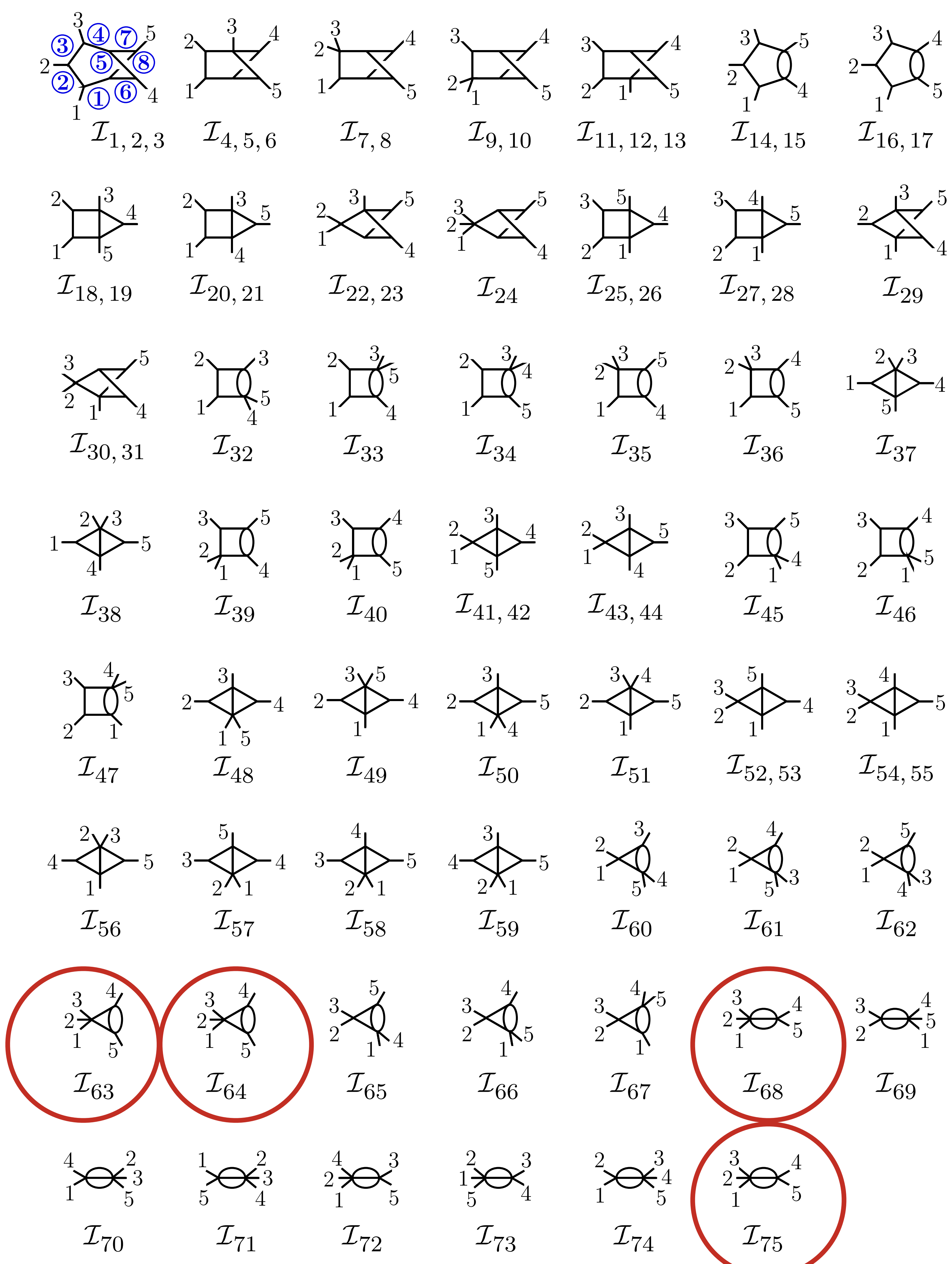
RREF: 2.5 hours, one core, 1.8 GB RAM (440 times)

$s_{12}, s_{23}, s_{34}, s_{45}, s_{15}$

Interpolation: 23 minutes, one core, 15 GB RAM

31 minutes, one core, 1.5 GB RAM $s_{12}, s_{13}, s_{14}, s_{23}, s_{24}$

Merge all the cuts



Reduce to

75 “pre”-master integrals

↓

73 master integrals

Towards an “industry”-level implement of our IBP algorithm

Dominik Bendle, Janko Boehm, Alessandro Georgoudis, YZ

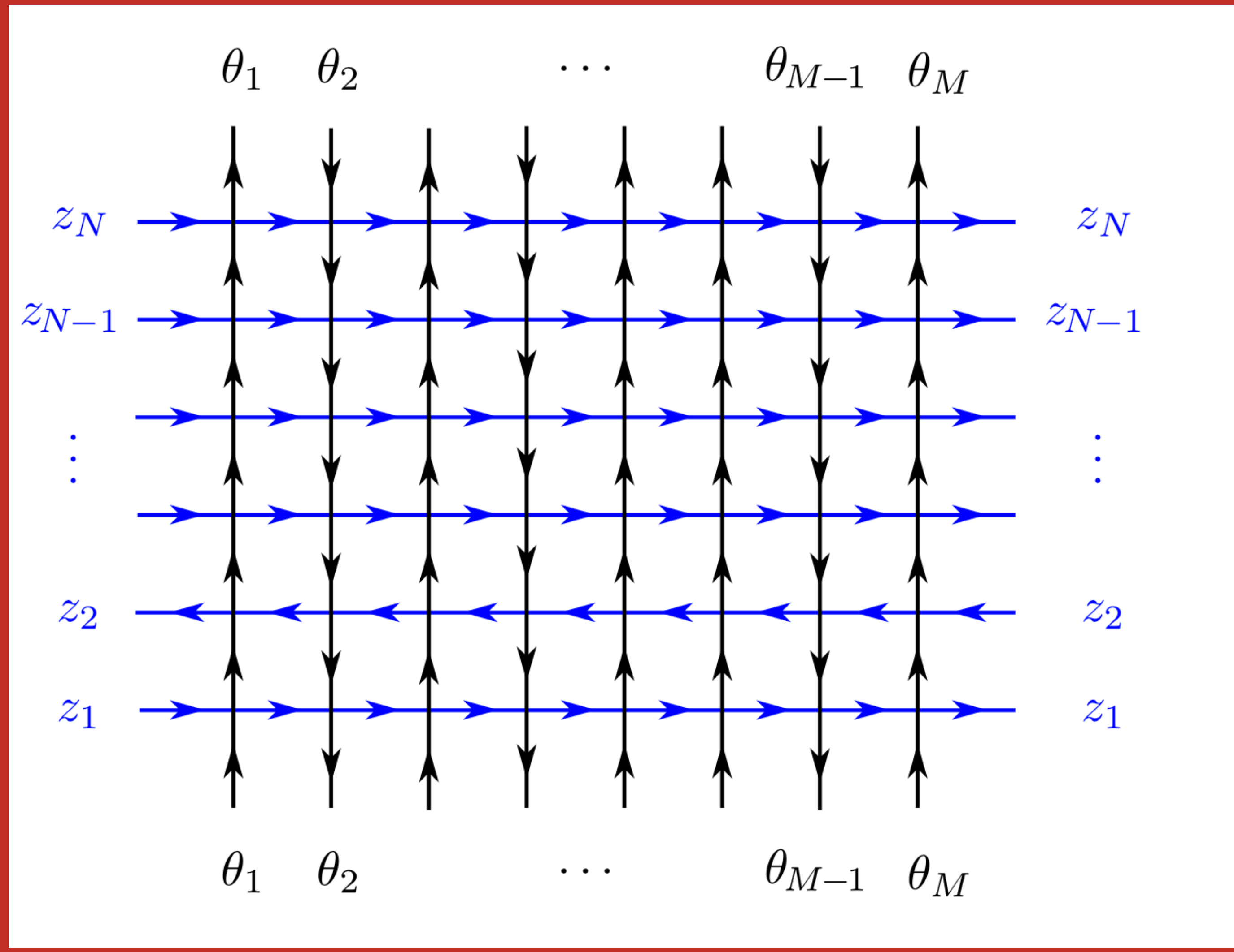
- more flexible choice of cuts
- adaptive kinematic variable choice for specific cuts
- more efficient parallelization strategy
- automatic selection of the pivot strategy
- all codes rewritten in **Singular**/C++

Also a different implement under development by Larsen

Summary

- Module intersection + Localization, towards a powerful automatic IBP program
- Very adaptable with modern efficient IBP algorithms
- Besides high-energy physics, algebro-geometry-inspired reduction method

can apply on statistical physics



6-vertex model for 2D ferroelectricity

Exact partition function can be computed by the reduction of Bethe-Ansatz equations via algebraic geometry

Jiang, YZ 2017