Certification of Minimal Approximant Bases

Pascal Giorgi [1], Vincent Neiger [2]

[1]Université de Montpellier, France

[2]Université de Limoges, France

ISSAC'2018, New York, USA
July 17, 2018

Approximant Bases

Let $F \in \mathbb{K}[X]^{m\times n}$ a matrix of power series truncated at order $\mathbf{d} = (d_1, \ldots, d_n)$ columnwise : $\forall 1 \le j \le n, \deg F_{*,j} < d_j$

- approximant of F at order $\mathbf{d}$:

$$p \in \mathbb{K}[X]^{1\times m} \text{ s.t. } pF = [0, \ldots, 0] \bmod X^{(d_1,\ldots,d_n)}$$

- the set $\mathcal{A}_{\mathbf{d}}(F)$ of all approximants of F forms a free $\mathbb{K}[X]$-module of rank m [Van Barel, Bultheel 1992].

A basis $P \in \mathbb{K}[X]^{m\times m}$ of $\mathcal{A}_{\mathbf{d}}(F)$ is called an approximant basis

Minimal Approximant Bases

Minimality

row-reduced over $\mathbb{K}[X]$, i.e. minimal row degree among all bases

$$P = \begin{bmatrix} 3x^3 & 2x^2 & x+3 \\ x^3+4x^2 & 2x^3+3x^2 & 5x^2 \\ x^3+6x^2+4x & 2x^3+8x^2+5 & 6x^2+3 \end{bmatrix}, \mathrm{rdeg}(P) = \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix}$$

Minimal Approximant Bases

Minimality

row-reduced over $\mathbb{K}[X]$, i.e. minimal row degree among all bases

$$P = \begin{bmatrix} 3x^3 & 2x^2 & x+3 \\ x^3 + 4x^2 & 2x^3 + 3x^2 & 5x^2 \\ x^3 + 6x^2 + 4x & 2x^3 + 8x^2 + 5 & 6x^2 + 3 \end{bmatrix}, \mathrm{rdeg}(P) = \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix}$$

$\Rightarrow$ row-reduction is related to the rdeg-leading matrix of P

$$\begin{bmatrix} 1 & & \\ & 1 & \\ & -1 & 1 \end{bmatrix} P = R = \begin{bmatrix} 3x^3 & 2x^2 & x+3 \\ x^3 + 4x^2 & 2x^3 + 3x^2 & 5x^2 \\ 2x^2 + 4x & 5x^2 + 5 & x^2 + 3 \end{bmatrix}, \mathrm{rdeg}(R) = \begin{bmatrix} 3 \\ 3 \\ 2 \end{bmatrix}$$

Shifted Minimal Approximant Bases

Shifted row degree (or $\mathbf{s}$-row degree)

degree measure for weighting the columns with a shift $\mathbf{s} = (s_1, \ldots, s_m)$

$$\mathrm{rdeg}_{\mathbf{s}}(P) = \mathrm{rdeg}(PX^{\mathbf{s}}) = \mathrm{rdeg}(P \begin{bmatrix} X^{s_1} & & \\ & \ddots & \\ & & X^{s_m} \end{bmatrix})$$

$\mathbf{s}$-minimal approximant bases

bases of $\mathcal{A}_{\mathbf{d}}(F)$ that have minimal $\mathbf{s}$-row degree among all bases ($\mathbf{s}$-reduced)

Shifted Minimal Approximant Bases

Shifted row degree (or **s**-row degree)

degree measure for weighting the columns with a shift $\mathbf{s} = (s_1, \ldots, s_m)$

$$\mathrm{rdeg}_{\mathbf{s}}(P) = \mathrm{rdeg}(PX^{\mathbf{s}}) = \mathrm{rdeg}(P \begin{bmatrix} X^{s_1} & & \\ & \ddots & \\ & & X^{s_m} \end{bmatrix})$$

s-minimal approximant bases

bases of $\mathcal{A}_{\mathbf{d}}(F)$ that have minimal **s**-row degree among all bases (**s**-reduced)

s-Popov approximant bases (uniqueness)

- $\mathrm{rdeg}_{\mathbf{s}}$-leading matrix $\rightarrow$ unitary lower triangular matrix
- cdeg-leading matrix $\rightarrow$ identity

Algorithms for Approximant Bases

- polynomial matrix $F \in \mathbb{K}[X]^{m \times n}$
- order $\mathbf{d} = (d_1, \ldots, d_n) \in \mathbb{Z}^n_{>0}$ with $D = |\mathbf{d}| = \sum_j d_j$
- shift $\mathbf{s} \in \mathbb{Z}^m$

Best known algorithms to date

cost in $O\tilde{~}(m^\omega D/m) = O\tilde{~}(m^{\omega-1} D)$

- minimal bases (unique order, no shift) [G., Jeannerod, Villard ISSAC'03]
- $\mathbf{s}$-minimal bases (unique order, small shifts) [Zhou, Labahn ISSAC'12]
- $\mathbf{s}$-Popov bases (all orders/shifts) [Jeannerod et al. ISSAC'16]

Algorithms for Approximant Bases

- polynomial matrix $F \in \mathbb{K}[X]^{m \times n}$
- order $\mathbf{d} = (d_1, \ldots, d_n) \in \mathbb{Z}_{>0}^n$ with $D = |\mathbf{d}| = \sum_j d_j$
- shift $\mathbf{s} \in \mathbb{Z}^m$

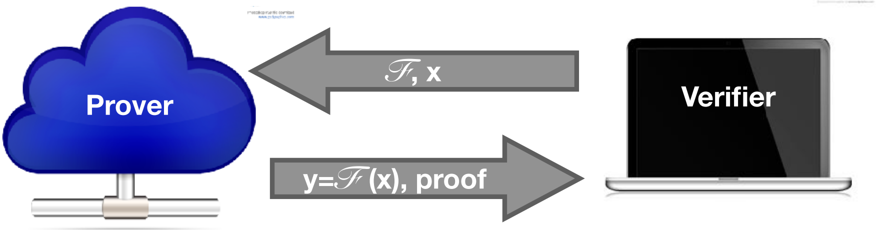
Best known algorithms to date

cost in $O\tilde{~}(m^\omega D/m) = O\tilde{~}(m^{\omega-1} D)$

- minimal bases (unique order, no shift) [G., Jeannerod, Villard ISSAC'03]
- $\mathbf{s}$-minimal bases (unique order, small shifts) [Zhou, Labahn ISSAC'12]
- $\mathbf{s}$-Popov bases (all orders/shifts) [Jeannerod et al. ISSAC'16]

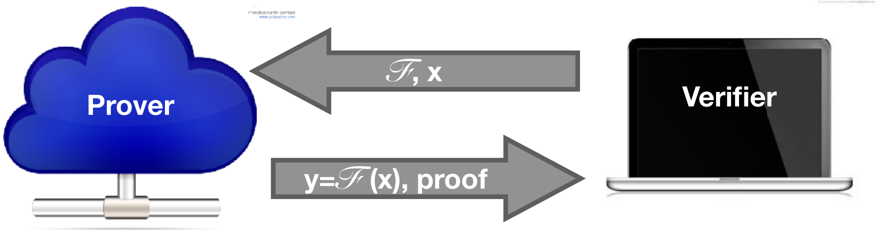
These are deterministic non-optimal algorithms, i.e. $Size(F) = mD$
when delegating computation $\rightarrow$ hope for faster verification

Verifying outsourced computation



- generating the proof must be negiglible
- verifying the proof must be easier than computing $\mathscr{F}(x)$
  $\rightarrow$ different models : interactive or static

Verifying outsourced computation



- generating the proof must be negiglible
- verifying the proof must be easier than computing $\mathscr{F}(x)$
  $\rightarrow$ different models : interactive or static

Sometimes the proof is unnecessary :
$\rightarrow$ Freivalds' verification of matrix mul. $(uA)B = uC$

Certifying linear algebra

Generic approaches exist

- Interactive proof for boolean circuits [Goldwasser, Kalai, Rothblum '08 ; Thaler '13]
- matrix mul. reduction $\rightarrow$ rerun with Freivalds [Kaltofen, Nehrig, Saunders ISSAC'11]

✗ prover or verifier time might not be optimal

Certifying linear algebra

Generic approaches exist

- Interactive proof for boolean circuits [Goldwasser, Kalai, Rothblum '08 ; Thaler '13]
- matrix mul. reduction $\rightarrow$ rerun with Freivalds [Kaltofen, Nehrig, Saunders ISSAC'11]

✗ prover or verifier time might not be optimal

Optimal ad'hoc verifications exist [Dumas,Kaltofen ISSAC'14]

✓ prover and verifier time can be "optimal"

✓ independent of the circuit (certifying result rather than execution)

Certifying linear algebra

Generic approaches exist

- Interactive proof for boolean circuits [Goldwasser, Kalai, Rothblum '08 ; Thaler '13]
- matrix mul. reduction $\rightarrow$ rerun with Freivalds [Kaltofen, Nehrig, Saunders ISSAC'11]

✗ prover or verifier time might not be optimal

Optimal ad'hoc verifications exist [Dumas,Kaltofen ISSAC'14]

✓ prover and verifier time can be "optimal"

✓ independent of the circuit (certifying result rather than execution)

How to optimally certify/verify approximant bases ?

Main result

Given P a $\mathbf{s}$-minimal basis of $\mathcal{A}_{\mathbf{d}}(F)$ with $Size(P) = O(mD)$

Static proof for $\mathbf{s}$-minimal approximant bases

- additional effort :$O(m^{\omega-1}D)$ prover
- Monte Carlo verification : $O(mD + m^{\omega-1}(m+n))$ verifier
- probability of error $\leq \frac{D}{\#S}$ for $S \subset \mathbb{K}$.

$\Rightarrow$ almost optimal certificate ($D \gg m^2$ often the case in practice)
$\Rightarrow$ total prover time remains in $\tilde{O}(m^{\omega-1}D)$

Main result

Given P a $\mathbf{s}$-minimal basis of $\mathcal{A}_{\mathbf{d}}(F)$ with $Size(P) = O(mD)$

$Size(P) = O(mD)$ not in general

$\Rightarrow$ but bases computed by best known algorithms have such property

- $|\mathrm{rdeg}(P)| \in O(D)$ [Van Barel, Bultheel '92 ; Zhou, Labahn ISSAC'12]
- $|\mathrm{cdeg}(P)| \leq D$ ($\mathbf{s}$-Popov) [Jeannerod et al. ISSAC'16]

How to certify approximant basis

1. Minimal : P is $\mathbf{s}$-reduced
2. Approximant : $PF = 0 \bmod X^{(d_1,\ldots,d_n)}$
3. Basis : rows of P generate $\mathcal{A}_{\mathbf{d}}(F)$

How to certify approximant basis

1. Minimal : P is $\mathbf{s}$-reduced

This amounts to check non-singularity of the $\mathrm{rdeg}_{\mathbf{s}}$-leading matrix of P
$\Rightarrow$ can be done at a cost $O(m^{\omega})$

How to certify approximant basis

2. Approximant : $PF = 0 \bmod X^{(d_1,\ldots,d_n)}$

not trivial $\rightarrow$ computing $PF \bmod X^{(d_1,\ldots,d_n)}$ costs $\tilde{O}(m^{\omega-1}D)$.

How to certify approximant basis

2. Approximant : $PF = 0 \bmod X^{(d_1,\dots,d_n)}$

not trivial $\rightarrow$ computing $PF \bmod X^{(d_1,\dots,d_n)}$ costs $\tilde{O}(m^{\omega-1}D)$.

Proposition : Freivalds + [G. '18]

verify $PF = G \bmod X^{(d_1,\dots,d_n)}$ at optimal cost $O(mD)$

How to certify approximant basis

2. Approximant : $PF = 0 \bmod X^{(d_1,\dots,d_n)}$

not trivial $\rightarrow$ computing $PF \bmod X^{(d_1,\dots,d_n)}$ costs $\tilde{O}(m^{\omega-1}D)$.

Proposition : Freivalds + [G. '18]

verify $PF = G \bmod X^{(d_1,\dots,d_n)}$ at optimal cost $O(mD)$

- check $(uP)F = uG \bmod X^{(d_1,\dots,d_n)}$ for a random vector u

How to certify approximant basis

2. Approximant : $PF = 0 \bmod X^{(d_1,\dots,d_n)}$

not trivial $\rightarrow$ computing $PF \bmod X^{(d_1,\dots,d_n)}$ costs $\tilde{O}(m^{\omega-1}D)$.

Proposition : Freivalds + [G. '18]

verify $PF = G \bmod X^{(d_1,\dots,d_n)}$ at optimal cost $O(mD)$

- check $(uP)F = uG \bmod X^{(d_1,\dots,d_n)}$ for a random vector u
- check for a random $\alpha \in S \subset \mathbb{K}$, $\delta = \max(d_1,\dots,d_n)$ that

$$\begin{bmatrix}1 & \alpha & \dots & \alpha^{\delta-1}\end{bmatrix} \begin{bmatrix} uP_0 & & & \\ uP_1 & \ddots & & \\ \vdots & \ddots & \ddots & \\ uP_{\delta-1} & \dots & uP_1 & uP_0 \end{bmatrix} \begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_{\delta-1} \end{bmatrix} = \begin{bmatrix}1 & \alpha & \dots & \alpha^{\delta-1}\end{bmatrix} \begin{bmatrix} uG_0 \\ uG_1 \\ \vdots \\ uG_{\delta-1} \end{bmatrix}$$

How to certify approximant basis

2. Approximant : $PF = 0 \bmod X^{(d_1,\ldots,d_n)}$

not trivial $\rightarrow$ computing $PF \bmod X^{(d_1,\ldots,d_n)}$ costs $O\tilde{~}(m^{\omega-1}D)$.

Proposition : Freivalds + [G. '18]

verify $PF = G \bmod X^{(d_1,\ldots,d_n)}$ at optimal cost $O(mD)$

- check $(uP)F = uG \bmod X^{(d_1,\ldots,d_n)}$ for a random vector u
- check for a random $\alpha \in S \subset \mathbb{K}$, $\delta = \max(d_1,\ldots,d_n)$ that

$$[1 \ \alpha \ \ldots \ \alpha^{\delta-1}] \begin{bmatrix} uP_0 & & & \\ uP_1 & \ddots & & \\ \vdots & \ddots & \ddots & \\ uP_{\delta-1} & \ldots & uP_1 & uP_0 \end{bmatrix} \begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_{\delta-1} \end{bmatrix} = uG(\alpha)$$

How to certify approximant basis

2. Approximant : $PF = 0 \bmod X^{(d_1,\ldots,d_n)}$

not trivial $\rightarrow$ computing $PF \bmod X^{(d_1,\ldots,d_n)}$ costs $\tilde{O}(m^{\omega-1}D)$.

Proposition : Freivalds + [G. '18]

verify $PF = G \bmod X^{(d_1,\ldots,d_n)}$ at optimal cost $O(mD)$

- check $(uP)F = uG \bmod X^{(d_1,\ldots,d_n)}$ for a random vector u
- check for a random $\alpha \in S \subset \mathbb{K}$, $\delta = \max(d_1,\ldots,d_n)$ that

$$\left[uP(\alpha)\ \ldots\ \alpha^{\delta-j}u(P \operatorname{rem} X^j)(\alpha)\ \ldots\ \alpha^{\delta-1}uP_0\right]\begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_{\delta-1}\end{bmatrix} = uG(\alpha)$$

Horner's intermediate values for $\alpha^{\delta-1}rev(uP)$ on $X = \alpha^{-1}$

How to certify approximant basis

3. Basis : rows of P generate $\mathcal{A}_{\mathbf{d}}(F)$

How to certify approximant basis

3. Basis : rows of P generate $\mathcal{A}_{\mathbf{d}}(F)$

Proposed lemma

rows of P generate $\mathcal{A}_{\mathbf{d}}(F)$ if and only if

- $PF = 0 \bmod X^{\mathbf{d}}$
- $det(P) = X^{\delta}$ for $0 < \delta \leq D$ [Beckermann, Labahn '97]
- the matrix $\begin{bmatrix} P(0) & C \end{bmatrix} \in \mathbb{K}^{m \times (m+n)}$ has full rank, where $C = PFX^{-\mathbf{d}} \bmod X$ (our certificate)

How to certify approximant basis

3. Basis : rows of P generate $\mathcal{A}_{\mathbf{d}}(F)$

Proposed lemma

rows of P generate $\mathcal{A}_{\mathbf{d}}(F)$ if and only if

- $PF = 0 \bmod X^{\mathbf{d}}$
- $det(P) = X^{\delta}$ for $0 < \delta \leq D$ [Beckermann, Labahn '97]
- the matrix $\begin{bmatrix} P(0) & C \end{bmatrix} \in \mathbb{K}^{m \times (m+n)}$ has full rank, where
  $C = PFX^{-\mathbf{d}} \bmod X$ (our certificate)

Idea of proof :

$$\mathcal{A}_{\mathbf{d}}(F) \simeq \ker\left(\begin{bmatrix} F \\ -X^{\mathbf{d}} \end{bmatrix}\right)$$

$$PF = 0 \bmod X^{\mathbf{d}} \iff \begin{bmatrix} P & PFX^{-\mathbf{d}} \end{bmatrix} \begin{bmatrix} F \\ -X^{\mathbf{d}} \end{bmatrix} = 0$$

Our protocol for certifying approximant bases

Prover (compute) — $\tilde{O}(m^{\omega-1}D)$

1. compute P a $\mathbf{s}$-minimal basis of $\mathcal{A}_{\mathbf{d}}(F)$ — $\hookrightarrow \tilde{O}(m^{\omega-1}D)$
2. compute $C = PFX^{-\mathbf{d}} \mod X$ — ? ? ?

$\Rightarrow$ send (P, C) to the verifier

Verifier (check) — $O(mD + m^{\omega-1}(m+n))$

1. non-singularity of $\mathit{leadmat}_{\mathrm{rdeg}_{\mathbf{s}}}(P)$ — $\hookrightarrow O(m^{\omega})$
2. full rank of $\begin{bmatrix} P(0) & C \end{bmatrix}$ — $\hookrightarrow O(m^{\omega-1}n)$
3. $\det(P(\alpha)) = \det(P(1))\alpha^{|\mathrm{rdeg}_{\mathbf{s}}(P)|-|\mathbf{s}|}$ with α random in $S \subset \mathbb{K}$ — $\hookrightarrow O(mD + m^{\omega})$
4. $PF = CX^{\mathbf{d}} \mod X^{(d_1+1,\ldots,d_n+1)}$ — $\hookrightarrow O(mD)$

How to efficiently generate the certificate

Compute C as the term of degree 0 in $PFX^{-\mathbf{d}}$:
$\rightarrow$ goal : no more than $\tilde{O}(m^{\omega-1}D)$

Easy when $n = m$ and $\mathbf{d} = (D/m, \ldots, D/m)$,

$$C = \sum_{k=1}^{D/m} P_k F_{D/m-k}$$

$\Rightarrow$ this costs at most $D/m \cdot O(m^{\omega}) = O(m^{\omega-1}D)$

How to efficiently generate the certificate

Taking care of unbalanced degrees $\mathbf{d} = (d_1, \ldots, d_n)$, with $D = |\mathbf{d}| = \sum d_j$

- all columns in F cannot have large degree, i.e. $|\mathrm{cdeg}(F)| = D$
- same remark on the rows of P when $|\mathrm{rdeg}(P)| = O(D)$ [1]

1. similar idea with $|\mathrm{cdeg}(P)| \leq D$

How to efficiently generate the certificate

Taking care of unbalanced degrees $\mathbf{d} = (d_1, \dots, d_n)$, with $D = |\mathbf{d}| = \sum d_j$

- all columns in F cannot have large degree, i.e. $|\mathrm{cdeg}(F)| = D$
- same remark on the rows of P when $|\mathrm{rdeg}(P)| = O(D)$ [1]

Extracting non-zero values according to the degrees

- $\#$ of rows in P with degree $\geq k$ is no more than D/k
- $\#$ of columns in F with degree $\geq k$ is no more than D/k

$$C = \sum_{k=1}^{\max(\mathbf{d})} P_k^* F_{\mathbf{d}-k}^*$$

- $\forall k < D/m$ each product costs $O(m^\omega)$
- $\forall k \geq D/m$ each product costs $O((D/k)^{\omega-1} m)$

Total cost in $O(m^{\omega-1} D)$

1. similar idea with $|\mathrm{cdeg}(P)| \leq D$

Our protocol for certifying approximant bases

Prover — $\tilde{O}(m^{\omega-1}D)$

1. compute P a $\mathbf{s}$-minimal basis of $\mathcal{A}_{\mathbf{d}}(F)$ — $\hookrightarrow \tilde{O}(m^{\omega-1}D)$
2. compute $C = PFX^{-\mathbf{d}} \bmod X$ — $\hookrightarrow O(m^{\omega-1}D)$

$\Rightarrow$ send (P, C) to the verifier

Verifier — $O(mD + m^{\omega-1}(m+n))$

1. check non-singularity of $leadmat_{\mathrm{rdeg}_{\mathbf{s}}}(P)$ — $\hookrightarrow O(m^{\omega})$
2. check full rank of $\begin{bmatrix} P(0) & C \end{bmatrix}$ — $\hookrightarrow O(m^{\omega-1}n)$
3. check $\det(P(\alpha)) = \det(P(1))\alpha^{|\mathrm{rdeg}_{\mathbf{s}}(P)|-|\mathbf{s}|}$ with α random in $S \subset \mathbb{K}$ — $\hookrightarrow O(mD + m^{\omega})$
4. check $PF = CX^{\mathbf{d}} \bmod X^{(d_1+1,\ldots,d_n+1)}$ — $\hookrightarrow O(mD)$

Conclusion

Almost optimal non-interactive certificate

- negligeable overhead for the *Prover*, only $O(m^{\omega-1}D)$
- verification time in $O(mD)$ + checking rank/det over $\mathbb{K}$
- probability of error $\leq \frac{D}{S}$ for $S \subset \mathbb{K}$ [Freivalds ; Schwartz, Zippel]
- certificate space is small, i.e. $O(mn)$

Conclusion

Almost optimal non-interactive certificate

- negligeable overhead for the *Prover*, only $O(m^{\omega-1}D)$
- verification time in $O(mD)$ + checking rank/det over $\mathbb{K}$
- probability of error $\leq \frac{D}{S}$ for $S \subset \mathbb{K}$ [Freivalds ; Schwartz, Zippel]
- certificate space is small, i.e. $O(mn)$

Remark

- turn "easily" into optimal interactive protocol by [Dumas, Kaltofen ISSAC'14]
- a LinBox's implementation should be available soon

Thank You

Certificate : sketch of proof

[Zhou, Labahn ISSAC'13, Neiger's PhD '16]

$$\mathcal{A}_{\mathbf{d}}(F) \qquad \simeq \qquad \ker(\begin{bmatrix} F \\ -X^{\mathbf{d}} \end{bmatrix})$$

$$PF = 0 \bmod X^{\mathbf{d}} \quad \Longleftrightarrow \quad \begin{bmatrix} P & Q \end{bmatrix} \begin{bmatrix} F \\ -X^{\mathbf{d}} \end{bmatrix} = 0$$

Column image of kernel bases :

$$\ker(\begin{bmatrix} F \\ -X^{\mathbf{d}} \end{bmatrix}) = \begin{bmatrix} 0_{m\times n} & I_m \end{bmatrix} V \text{ with } \quad V \in \mathrm{GL}_{m+n}(\mathbb{K}[X])$$

- P basis :
  $\begin{bmatrix} P & Q \end{bmatrix} = \ker(\begin{bmatrix} F \\ -X^{\mathbf{d}} \end{bmatrix}) \Longrightarrow \mathrm{rank}(\begin{bmatrix} P & Q \end{bmatrix}) = \mathrm{rank}(\begin{bmatrix} P(0) & Q(0) \end{bmatrix}) = m$
- P not basis :
  $\begin{bmatrix} P & Q \end{bmatrix} = U \begin{bmatrix} A & AFX^{-\mathbf{d}} \end{bmatrix}$ with $\det(U) = X^{\delta}$
  $\Longrightarrow \mathrm{rank}(\begin{bmatrix} P(0) & Q(0) \end{bmatrix}) < m$

Verifying truncated polynomial matrix product

The polynomial case [G. '18]

Let $A = a_0 + a_1X + \dots + a_{k-1}X^{k-1}$ and $B = b_0 + b_1X + \dots + b_{k-1}X^{k-1}$, sampling random value $X = \alpha$ in $C = AB \bmod X^k$ corresponds to :

$$\begin{bmatrix} 1 & \alpha & \dots & \alpha^{k-1} \end{bmatrix} \begin{bmatrix} a_0 & & & \\ a_1 & \ddots & & \\ \vdots & \ddots & \ddots & \\ a_{k-1} & \dots & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{k-1} \end{bmatrix} = \begin{bmatrix} 1 & \alpha & \dots & \alpha^{k-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{k-1} \end{bmatrix}$$

Verifying truncated polynomial matrix product

The polynomial case [G. '18]

Let $A = a_0 + a_1 X + \cdots + a_{k-1} X^{k-1}$ and $B = b_0 + b_1 X + \cdots + b_{k-1} X^{k-1}$, sampling random value $X = \alpha$ in $C = AB \bmod X^k$ corresponds to :

$$\begin{bmatrix} 1 & \alpha & \dots & \alpha^{k-1} \end{bmatrix} \begin{bmatrix} a_0 & & & \\ a_1 & \ddots & & \\ \vdots & \ddots & \ddots & \\ a_{k-1} & \dots & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{k-1} \end{bmatrix} = C(\alpha)$$

Verifying truncated polynomial matrix product

The polynomial case [G. '18]

Let $A = a_0 + a_1X + \cdots + a_{k-1}X^{k-1}$ and $B = b_0 + b_1X + \cdots + b_{k-1}X^{k-1}$, sampling random value $X = \alpha$ in $C = AB \bmod X^k$ corresponds to :

$$\begin{bmatrix} A(\alpha) & \dots & \alpha^{k-j}(A \operatorname{rem} X^j)(\alpha) & \dots & \alpha^{k-1}a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix} = C(\alpha)$$

Verifying truncated polynomial matrix product

The polynomial case [G. '18]

Let $A = a_0 + a_1 X + \cdots + a_{k-1} X^{k-1}$ and $B = b_0 + b_1 X + \cdots + b_{k-1} X^{k-1}$, sampling random value $X = \alpha$ in $C = AB \bmod X^k$ corresponds to :

$$\begin{bmatrix} A(\alpha) & \dots & \alpha^{k-j}(A \operatorname{rem} X^j)(\alpha) & \dots & \alpha^{k-1} a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix} = C(\alpha)$$

⇒ verification in $O(k)$ using Horner's algo. on $\alpha^{k-1}\mathrm{rev}(A)$ with $X = \alpha^{-1}$

⇒ proba error $< \frac{k}{\#S}$ for $S \subset \mathbb{K}$ [Schwartz, Zippel '79]

Verifying truncated polynomial matrix product

The polynomial matrix case

Let $P \in \mathbb{K}[X]^{m\times m}$, $F, G \in \mathbb{K}[X]^{m\times n}$, $\mathbf{t} = (t_1, \ldots, t_n)$ and $\delta = \max(\mathbf{t})$
How to check $PF = G \bmod X^{\mathbf{t}}$?

1. shrink matrix row dimension *a la Freidvalds*, random $u \in \mathbb{K}^{1\times m}$
   $\rightarrow p = uP \in \mathbb{K}[x]^{1\times m}$ and $g = uG \in \mathbb{K}[X]^{1\times n}$
2. apply idea of [G. '18] with vector/matrix

$$\begin{bmatrix}1 & \alpha & \ldots & \alpha^{\delta-1}\end{bmatrix} \begin{bmatrix} p_0 & & & \\ p_1 & \ddots & & \\ \vdots & \ddots & \ddots & \\ p_{\delta-1} & \ldots & p_1 & p_0 \end{bmatrix} \begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_{\delta-1} \end{bmatrix} = g(\alpha)$$

Verifying truncated polynomial matrix product

The polynomial matrix case

Let $P \in \mathbb{K}[X]^{m \times m}$, $F, G \in \mathbb{K}[X]^{m \times n}$, $\mathbf{t} = (t_1, \ldots, t_n)$ and $\delta = \max(\mathbf{t})$
How to check $PF = G \bmod X^{\mathbf{t}}$?

1. shrink matrix row dimension *a la Freidvalds*, random $u \in \mathbb{K}^{1 \times m}$
   $\rightarrow p = uP \in \mathbb{K}[x]^{1 \times m}$ and $g = uG \in \mathbb{K}[X]^{1 \times n}$
2. apply idea of [G. '18] with vector/matrix

$$\underbrace{\begin{bmatrix} p(\alpha) & \ldots & \alpha^{\delta-j}(p \operatorname{rem} X^j)(\alpha) & \ldots & \alpha^{\delta-1} p_0 \end{bmatrix}}_{\in \mathbb{K}^{1 \times m\delta}} \begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_{\delta-1} \end{bmatrix} = g(\alpha)$$

Verifying truncated polynomial matrix product

The polynomial matrix case

Let $P \in \mathbb{K}[X]^{m\times m}$, $F, G \in \mathbb{K}[X]^{m\times n}$, $\mathbf{t} = (t_1, \dots, t_n)$ and $\delta = \max(\mathbf{t})$
How to check $PF = G \bmod X^{\mathbf{t}}$?

1. shrink matrix row dimension *a la Freidvalds*, random $u \in \mathbb{K}^{1\times m}$
   $\to p = uP \in \mathbb{K}[x]^{1\times m}$ and $g = uG \in \mathbb{K}[X]^{1\times n}$
2. apply idea of [G. '18] with vector/matrix

$$\underbrace{\begin{bmatrix} p(\alpha) & \dots & \alpha^{\delta-j}(p \,\mathrm{rem}\, X^j)(\alpha) & \dots & \alpha^{\delta-1}p_0 \end{bmatrix}}_{\in \mathbb{K}^{1\times m\delta}} \begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_{\delta-1} \end{bmatrix} = g(\alpha)$$

$\Rightarrow$ verification in $O(size(P) + m\sum t_i)$
$\Rightarrow$ proba error $< \frac{\delta}{\#S}$ for $S \subset \mathbb{K}$