Approximate Homomorphic Encryption

- Construction & Bootstrapping

Yongsoo Song, UC San Diego

ECC 2018, Osaka

Approximate Homomorphic Encryption

- Construction & Bootstrapping

Yongsoo Song, ~~UC San Diego~~
Microsoft Research, Redmond

ECC 2018, Osaka

Table of Contents

- **Background**
- Construction
  - [CKKS, AC17] Homomorphic Encryption for Arithmetic of Approximate Numbers
- Bootstrapping
  - [CHKKS, EC18] Bootstrapping for Approximate Homomorphic Encryption
- Related Works

Table of Contents

- **Background**
- Construction
  - [CKKS, AC17] Homomorphic Encryption for Arithmetic of Approximate Numbers

HEAAN (慧眼)

- Bootstrapping
  - [CHKKS, EC18] Bootstrapping for Approximate Homomorphic Encryption
- Related Works

Advanced Cryptography

- Protecting Computation, not just data

Advanced Cryptography

- Protecting Computation, not just data

- Differential Privacy
- Zero-knowledge Proof
- Multiparty Computation
- Attribute Based Encryption
- ...

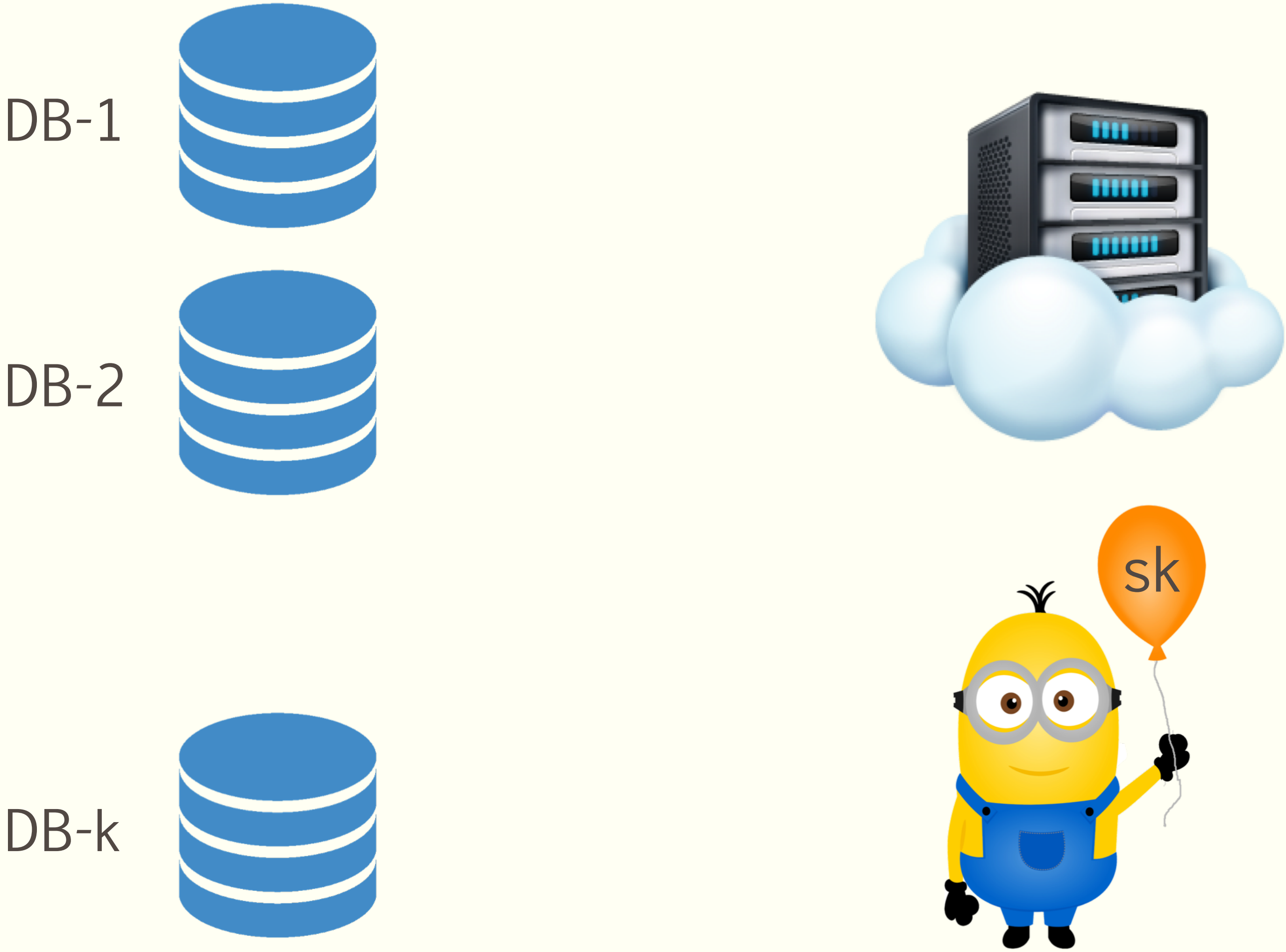
Advanced Cryptography

- Protecting Computation, not just data

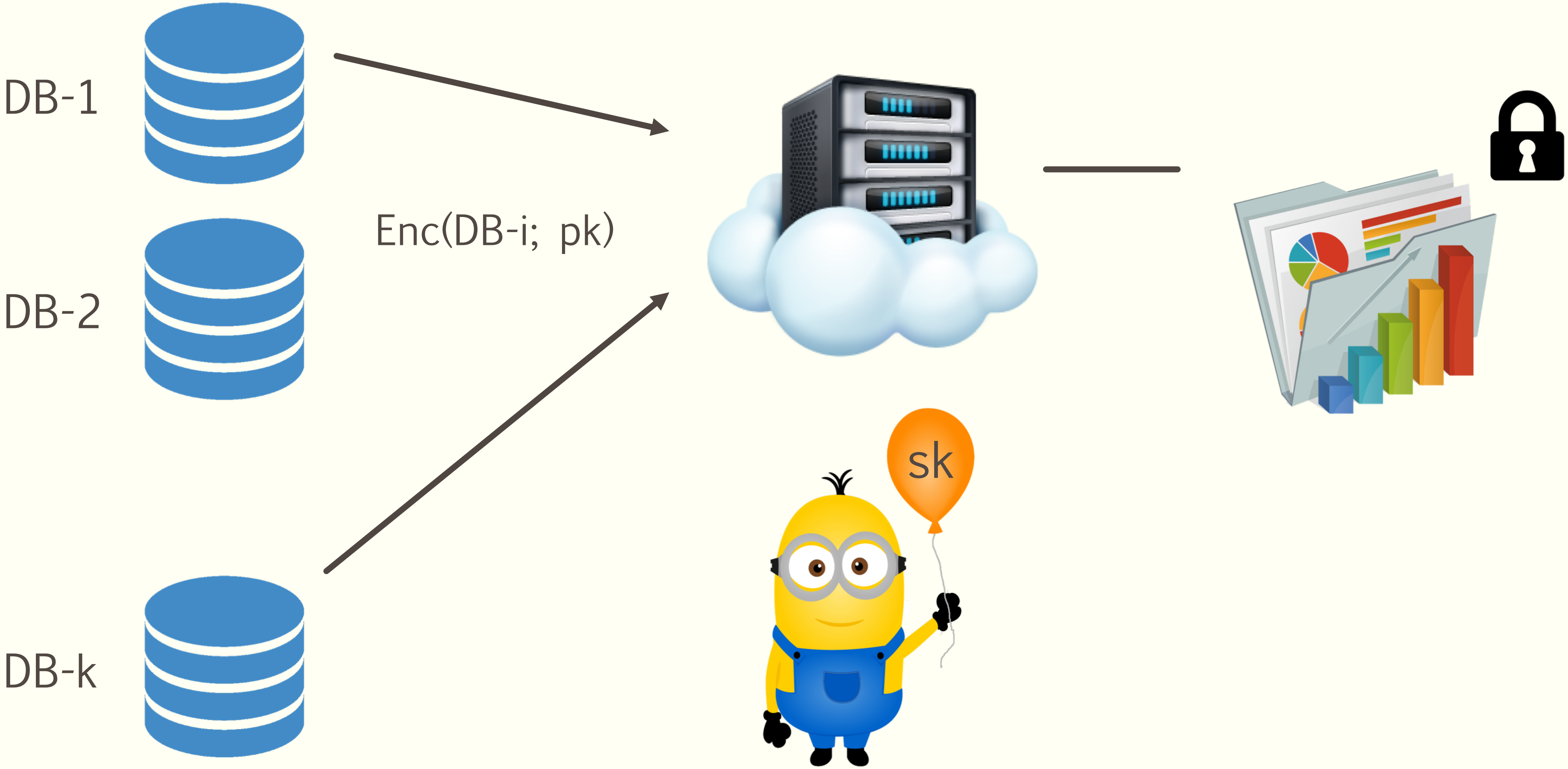
- Differential Privacy
- Zero-knowledge Proof
- Multiparty Computation
- Attribute Based Encryption
- …
- Homomorphic Encryption (2009~)

$$
\begin{tikzcd}
m \arrow[rr] \arrow[d, shift left, "\mathrm{Enc}"] & & \mathrm{F}(m) \arrow[d, shift left, "\mathrm{Enc}"] \\
\mathrm{Enc}(m) \arrow[rr] \arrow[u, shift left, "\mathrm{Dec}"] & & \mathrm{F}\big(\mathrm{Enc}(m)\big) \arrow[u, shift left, "\mathrm{Dec}"]
\end{tikzcd}
$$

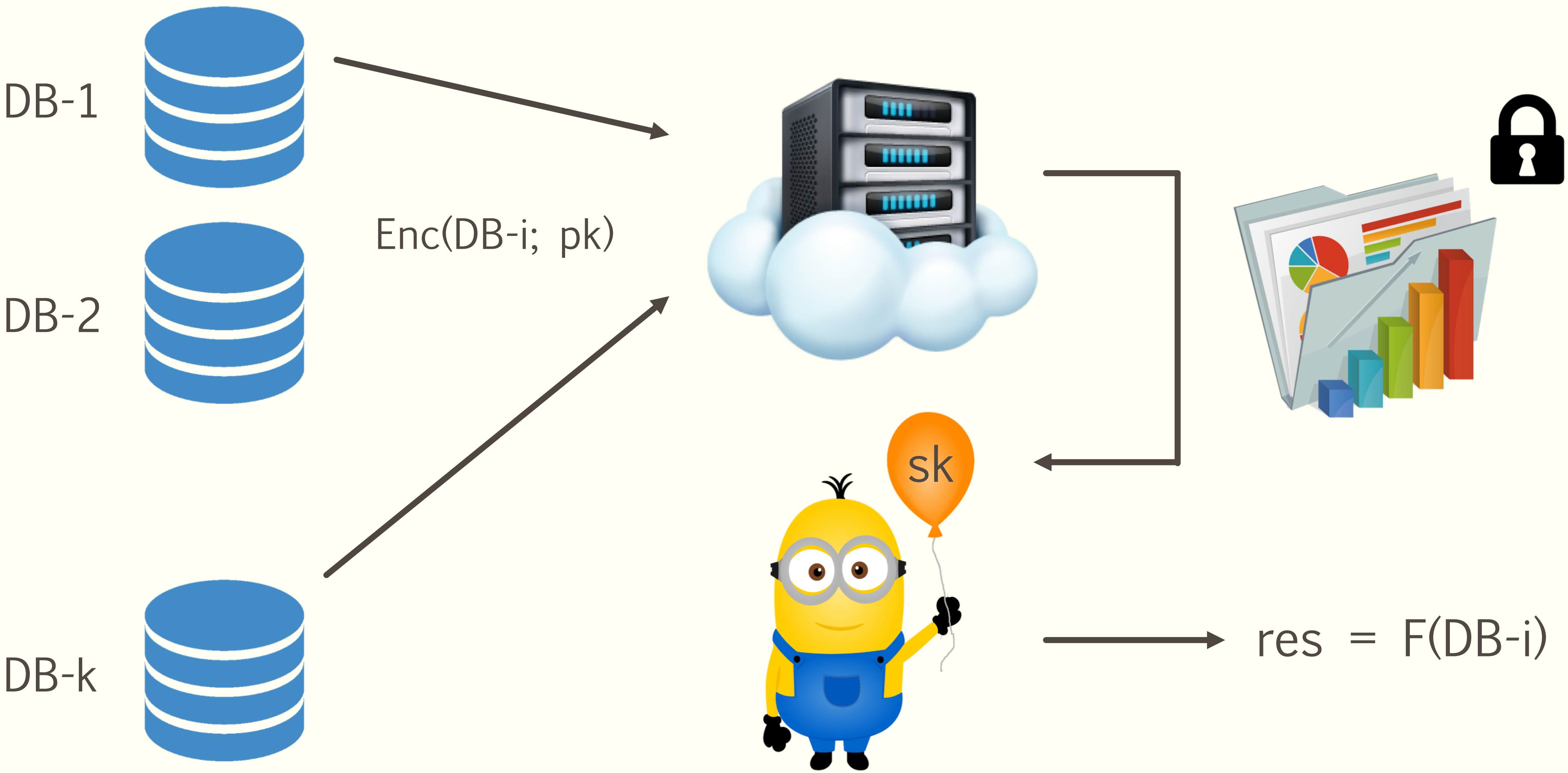
Homomorphic Encryption



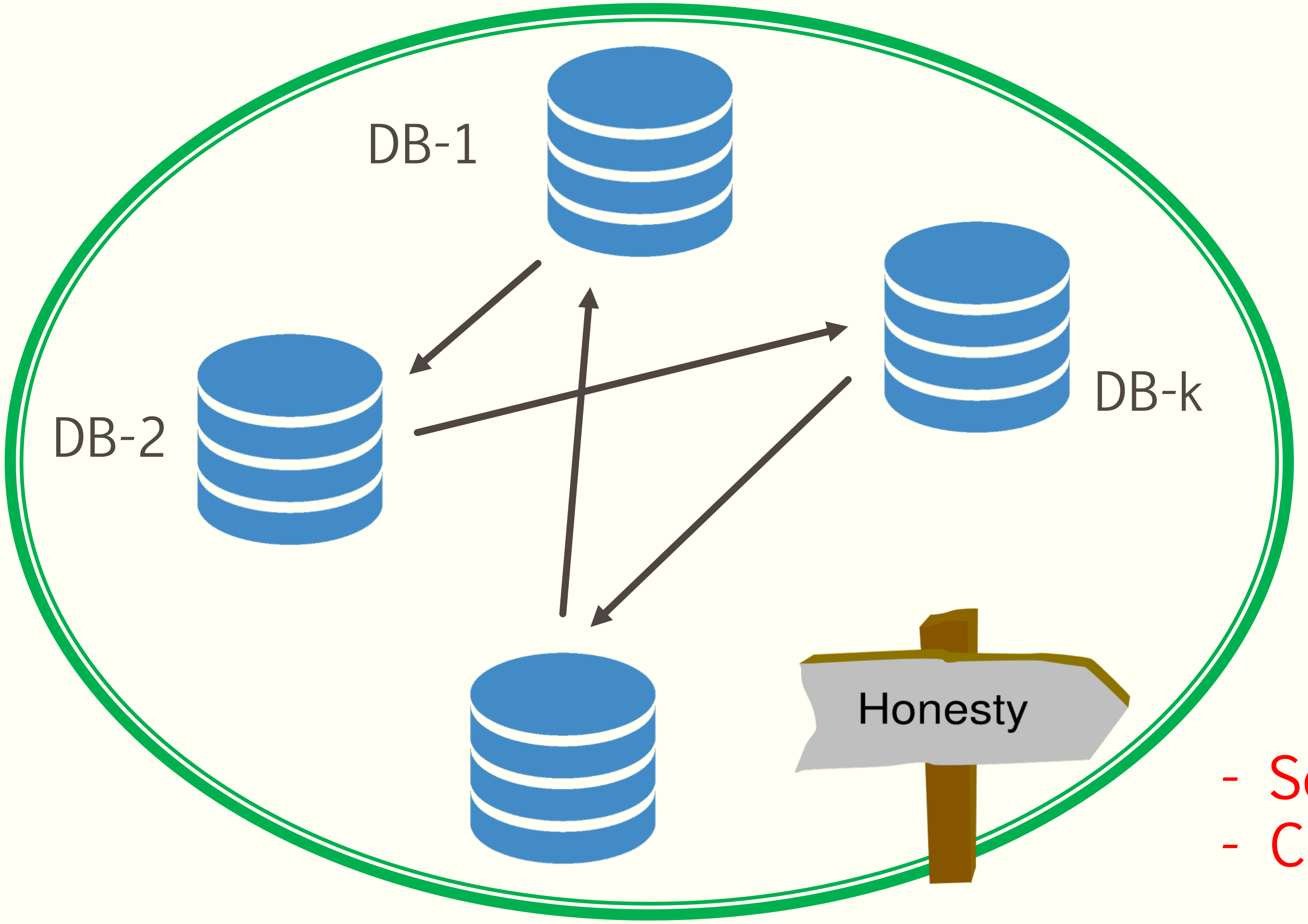
Homomorphic Encryption



Homomorphic Encryption



Multi-Party Computation



res = F(DB-i)

- Security Model
- Communication Costs |F|, k

Comparison: HE vs MPC

	Homomorphic Encryption	Multi-Party Computation
Re-usability	One-time encryption No further interaction	Single-use encryption Interaction between parties each time
Model		
Speed		

Comparison: HE vs MPC

	Homomorphic Encryption	Multi-Party Computation
Re-usability	One-time encryption No further interaction	Single-use encryption Interaction between parties each time
Model	Semi-honest Cloud + Trusted SK Owner	Semi-honest parties without collusion
Speed		

Comparison: HE vs MPC

	Homomorphic Encryption	Multi-Party Computation
Re-usability	One-time encryption No further interaction	Single-use encryption Interaction between parties each time
Model	Semi-honest Cloud + Trusted SK Owner	Semi-honest parties without collusion
Speed	Slow in computation (but can speed-up using SIMD)	Slow in communication (due to large circuit to be exchanged)

Summary of Progresses

- 2009-10: Plausibility
  - [GH11] A single bit operation takes 30 minutes
- 2011-12: Large Circuits
  - [GHS12b] 120 blocks of AES-128 (30K gates) in 36 hours

Summary of Progresses

- 2009-10: Plausibility
  - [GH11] A single bit operation takes 30 minutes
- 2011-12: Large Circuits
  - [GHS12b] 120 blocks of AES-128 (30K gates) in 36 hours
- 2013-15: Efficiency
  - [HS14] IBM's open-source library HElib
  - Implementation of Brakerski-Gentry-Vaikuntanathan (BGV) scheme
  - The same 30K-gate circuit in 4 minutes

Summary of Progresses

- 2009-10: Plausibility
  - [GH11] A single bit operation takes 30 minutes
- 2011-12: Large Circuits
  - [GHS12b] 120 blocks of AES-128 (30K gates) in 36 hours
- 2013-15: Efficiency
  - [HS14] IBM's open-source library HElib
  - Implementation of Brakerski-Gentry-Vaikuntanathan (BGV) scheme
  - The same 30K-gate circuit in 4 minutes
- 2015-today: Usability
  - Various schemes with different advantages
  - Simpler and faster implementations
  - Real-world tasks: Big data analysis, Machine learning
  - Standardization meetings (2017~)
  - iDASH competitions (2014~)

4 Big Takeaways from Satya Nadella's Talk at Microsoft Build

By JONATHAN VANIAN May 7, 2018

Microsoft CEO Satya Nadella is trying to distinguish the business technology giant from its technology brethren by focusing on digital privacy.

From Dr. Kristin Lauter's Keynote Talk at iDASH 2018

4 Big Takeaways from Satya Nadella's Talk at Microsoft Build

One way Nadella is attempting to convince businesses that Microsoft (MSFT, +3.63%) can improve its AI technology while protecting user data is by promoting a computing technique called homomorphic encryption. Although still a research-heavy technique, homomorphic encryption would presumably let companies analyze and crunch encrypted data without needing to unscramble that information.

Nadella is pitching the technique as a way for companies to “learn, train on encrypted data.” The executive didn’t explain how far along Microsoft is on advancing the encryption technique, but the fact that he mentioned the wonky terms shows that the company is touting user privacy as a selling point for its Azure cloud business.

From Dr. Kristin Lauter's Keynote Talk at iDASH 2018

4 Big Takeaways from Satya Nadella's Talk at Microsoft Build

One way Nadella is attempting to convince businesses that Microsoft (MSFT, +3.63%) can improve its AI technology while protecting user data is by promoting a computing technique called homomorphic encryption. Although still a research-heavy technique, homomorphic encryption would presumably let companies analyze and crunch encrypted data without needing to unscramble that information.

Nadella is pitching the technique as a way for companies to "learn, train on encrypted data." The executive didn't explain how far along Microsoft is on advancing the encryption technique, but the fact that he mentioned the wonky terms shows that the company is touting user privacy as a selling point for its Azure cloud business.

IBM

Google

ENVEIL ENCRYPTED VEIL

DUALITY TECHNOLOGIES

From Dr. Kristin Lauter's Keynote Talk at iDASH 2018

Best Performing HE Schemes

Type	Classical HE	Fast Bootstrapping	Approximate Encryption
Scheme	[BGV12] BGV [Bra12, FV12] B/FV	[DM15] FHEW [CGGI16] TFHE	[CKKS17] HEAAN
Plaintext			
Operation			
Library			

Best Performing HE Schemes

Type	Classical HE	Fast Bootstrapping	Approximate Encryption
Scheme	[BGV12] BGV [Bra12, FV12] B/FV	[DM15] FHEW [CGGI16] TFHE	[CKKS17] HEAAN
Plaintext	Finite Field Packing		
Operation	Addition, Multiplication		
Library	HElib (IBM) SEAL (Microsoft Research) Palisade (Duality inc.)		

Best Performing HE Schemes

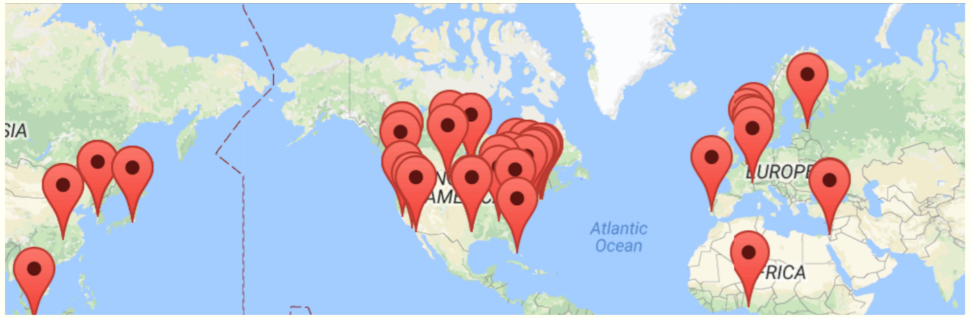
Type	Classical HE	Fast Bootstrapping	Approximate Encryption
Scheme	[BGV12] BGV [Bra12, FV12] B/FV	[DM15] FHEW [CGGI16] TFHE	[CKKS17] HEAAN
Plaintext	Finite Field Packing	Binary string	
Operation	Addition, Multiplication	Look-up table & bootstrapping	
Library	HElib (IBM) SEAL (Microsoft Research) Palisade (Duality inc.)	TFHE (inpher, gemalto, etc.)	

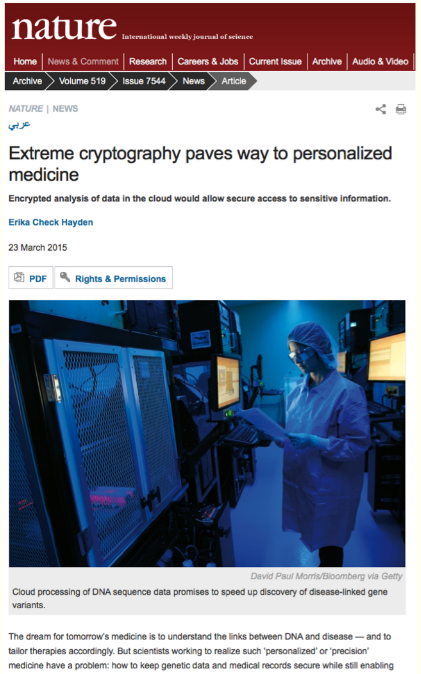
Best Performing HE Schemes

Type	Classical HE	Fast Bootstrapping	Approximate Encryption
Scheme	[BGV12] BGV [Bra12, FV12] B/FV	[DM15] FHEW [CGGI16] TFHE	[CKKS17] HEAAN
Plaintext	Finite Field Packing	Binary string	Real/Complex numbers Packing
Operation	Addition, Multiplication	Look-up table & bootstrapping	Fixed-point Arithmetic
Library	HElib (IBM) SEAL (Microsoft Research) Palisade (Duality inc.)	TFHE (inpher, gemalto, etc.)	HEAAN (SNU)

iDASH Security & Privacy Workshop

- An interdisciplinary challenge on genomic privacy research
- Motivated by real world biomedical applications
- Participation of privacy technology experts (academia and industry)
- Developed practical yet rigorous solutions for privacy preserving genomic data sharing and analysis
- Reported in the media (e.g., Nature News, GenomeWeb)





nature International weekly journal of science

Home | News & Comment | Research | Careers & Jobs | Current Issue | Archive | Audio & Video

Archive > Volume 519 > Issue 7544 > News > Article

NATURE | NEWS

عربي

Extreme cryptography paves way to personalized medicine

Encrypted analysis of data in the cloud would allow secure access to sensitive information.

Erika Check Hayden

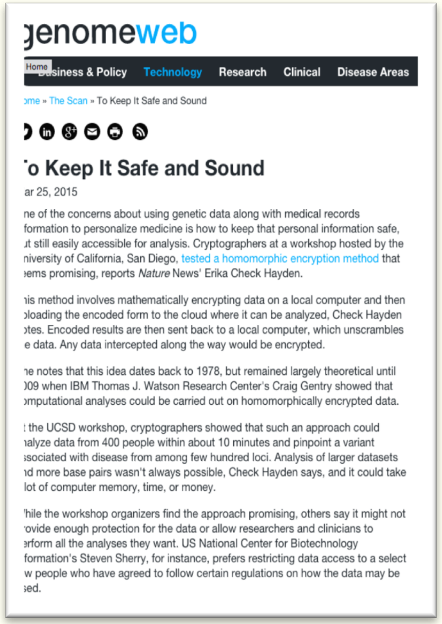
23 March 2015

PDF | Rights & Permissions

David Paul Morris/Bloomberg via Getty

Cloud processing of DNA sequence data promises to speed up discovery of disease-linked gene variants.

The dream for tomorrow's medicine is to understand the links between DNA and disease — and to tailor therapies accordingly. But scientists working to realize such 'personalized' or 'precision' medicine have a problem: how to keep genetic data and medical records secure while still enabling



genomeweb

Home | Business & Policy | Technology | Research | Clinical | Disease Areas

ome » The Scan » To Keep It Safe and Sound

o Keep It Safe and Sound

ar 25, 2015

ne of the concerns about using genetic data along with medical records formation to personalize medicine is how to keep that personal information safe, it still easily accessible for analysis. Cryptographers at a workshop hosted by the niversity of California, San Diego, tested a homomorphic encryption method that eems promising, reports *Nature* News' Erika Check Hayden.

is method involves mathematically encrypting data on a local computer and then ploading the encoded form to the cloud where it can be analyzed, Check Hayden otes. Encoded results are then sent back to a local computer, which unscrambles e data. Any data intercepted along the way would be encrypted.

e notes that this idea dates back to 1978, but remained largely theoretical until 009 when IBM Thomas J. Watson Research Center's Craig Gentry showed that omputational analyses could be carried out on homomorphically encrypted data.

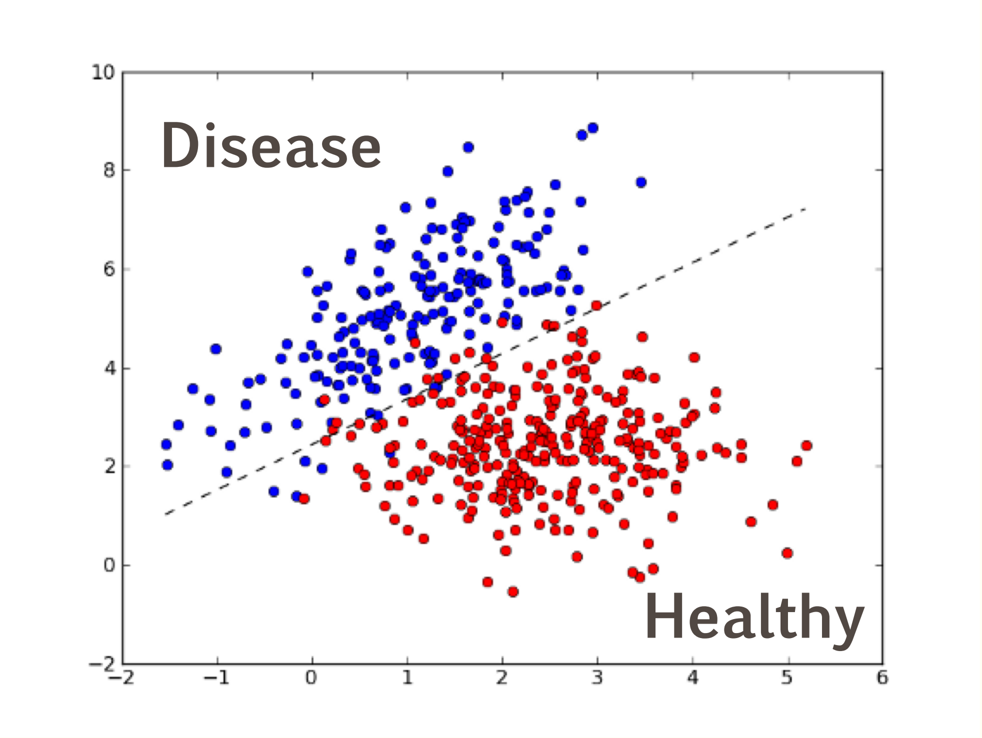
the UCSD workshop, cryptographers showed that such an approach could nalyze data from 400 people within about 10 minutes and pinpoint a variant ssociated with disease from among few hundred loci. Analysis of larger datasets d more base pairs wasn't always possible, Check Hayden says, and it could take lot of computer memory, time, or money.

hile the workshop organizers find the approach promising, others say it might not ovide enough protection for the data or allow researchers and clinicians to erform all the analyses they want. US National Center for Biotechnology formation's Steven Sherry, for instance, prefers restricting data access to a select w people who have agreed to follow certain regulations on how the data may be sed.

http://www.nature.com/news/extreme-cryptography-paves-way-to-personalized-medicine-1.17174

iDASH 2017 – Logistic Regression Model Training

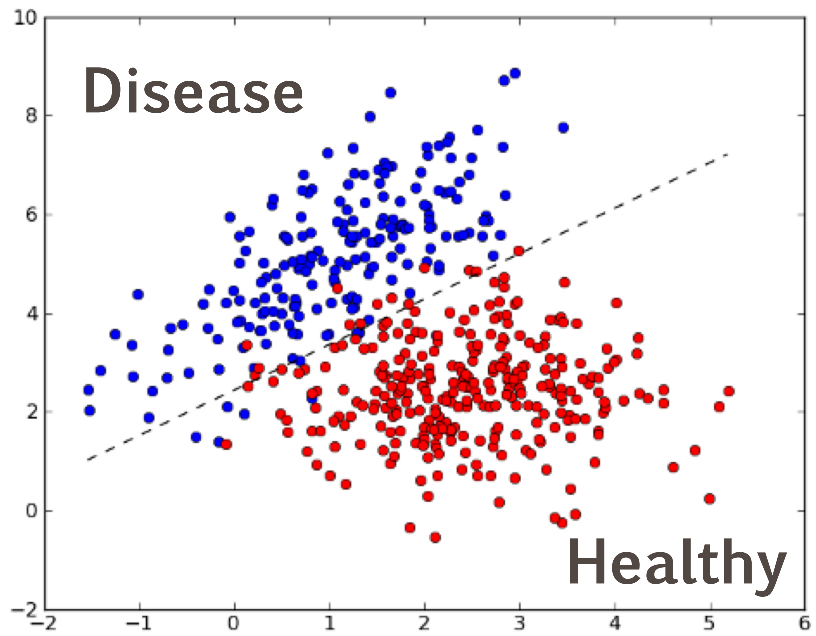
- A machine learning model to predict the disease
- 1500 records + 18 features for training



iDASH 2017 – Logistic Regression Model Training

- A machine learning model to predict the disease
- 1500 records + 18 features for training

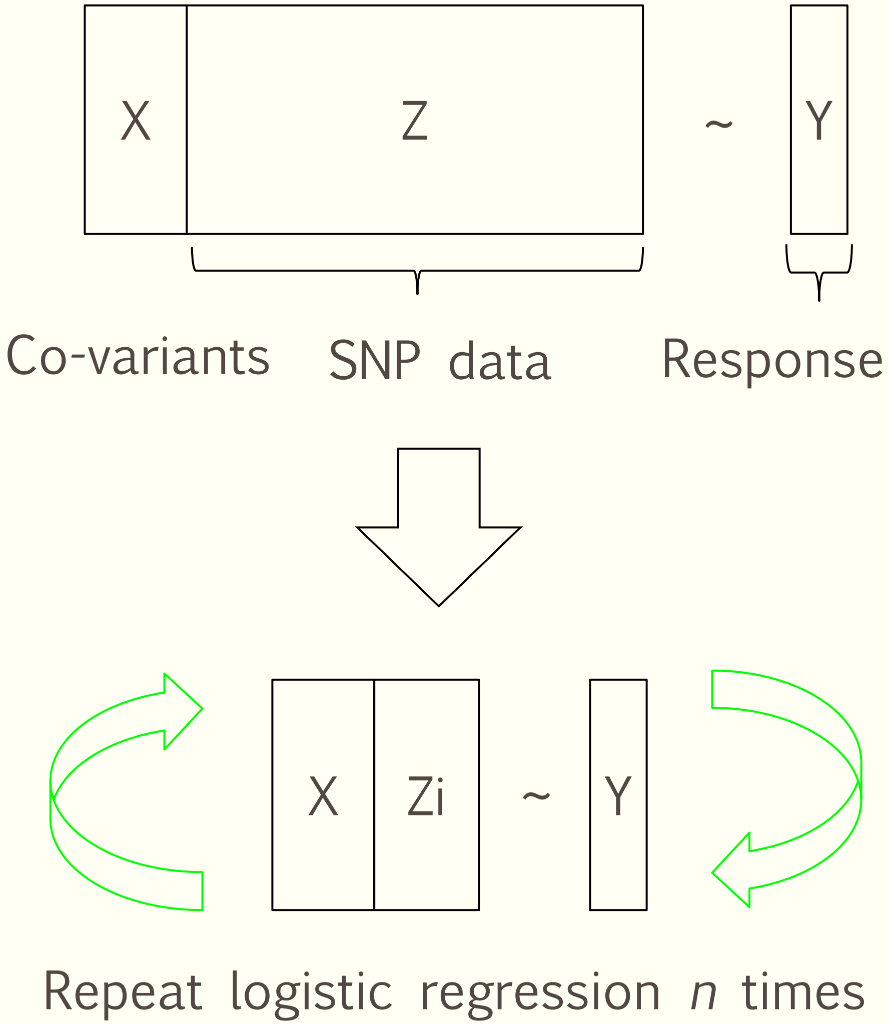
Teams	AUC 0.7136	Secure learning		Overall time (mins)
		Time (mins)	Memory (MB)	
SNU	0.6934	10.250	2775.333	10.360
CEA LIST	0.6930	2206.057	238.255	2207.363
KU Leuven	0.6722	155.695	7266.727	160.912
EPFL	0.6584	15.089	1498.513	16.739
MSR	0.6574	385.021	26299.344	396.390
Waseda*	0.7154	2.077	7635.600	5.332
Saarland**	N/A	48.356	29752.527	57.344



* Interactive mechanism, no complete guarantee on 80-bit security at "analyst" side

iDASH 2018 – Semi-Parallel GWAS

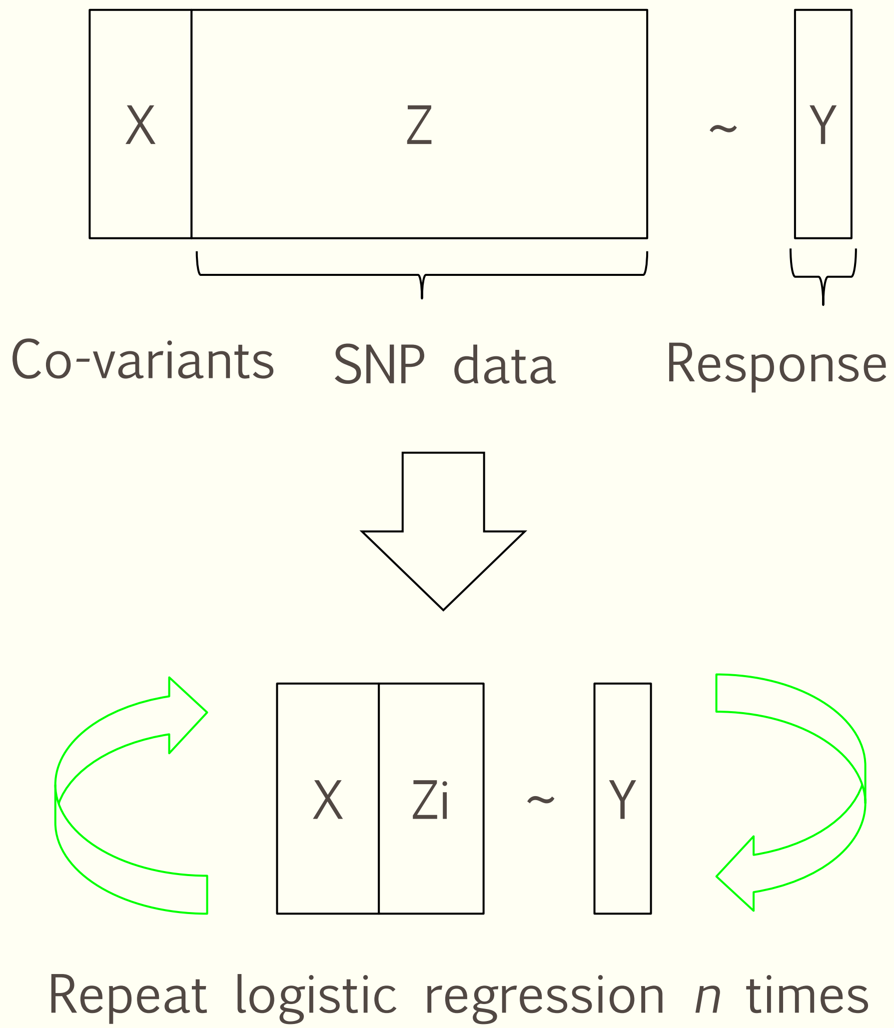
- Compute Genome Wide Association Studies (GWAS)
- 3 Co-variants [age, height, weight] + 14,841 SNPS



iDASH 2018 – Semi-Parallel GWAS

- Compute Genome Wide Association Studies (GWAS)
- 3 Co-variants [age, height, weight] + 14,841 SNPS

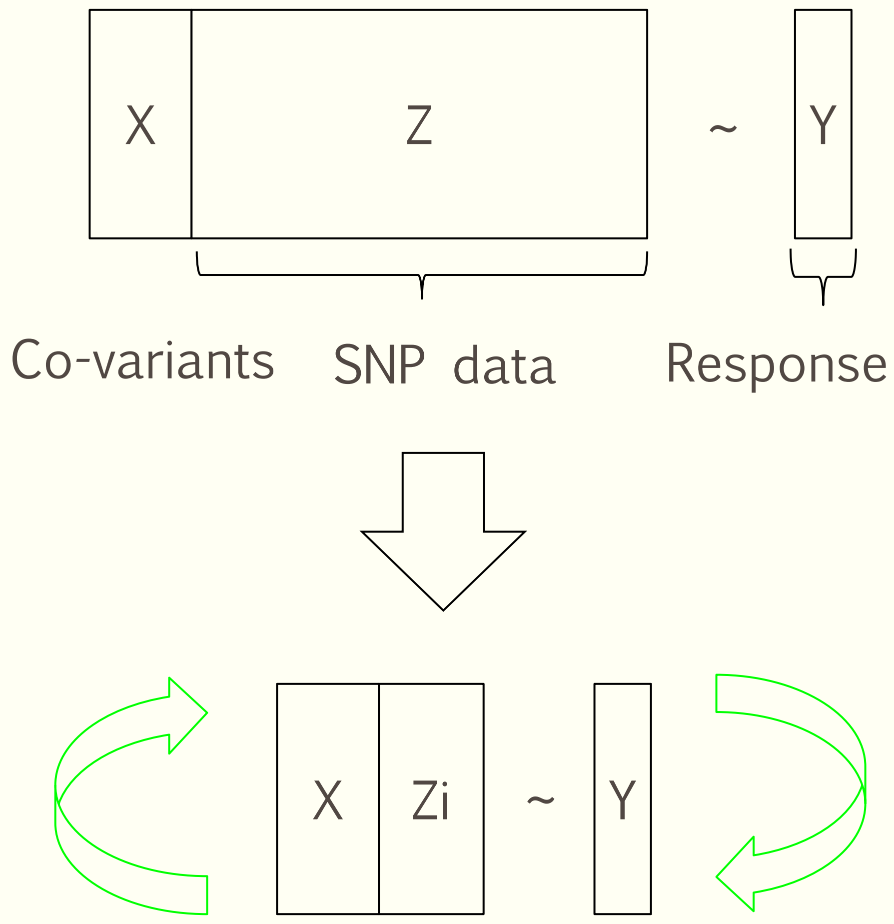
Team	Submission	Schemes	Time (mins)	Memory (MB)	Accuracy
A*FHE	A*FHE 1	HEAAN	922.48	3,777	0.999
	A*FHE 2		1,632.97	4,093	0.905
Chimera	Version 1	TFHE+HEAAN (Chimera)	201.73	10,375	0.993
	Version 2		215.95	15,166	0.35
Delft Blue	Delft Blue	HEAAN	1,844.82	10,814	0.969
UCSD	Log Reg	HEAAN	1.66	14,901	0.993
	Lin Reg	pkg: RNS HEAAN	0.42	3,387	0.989
Duality Inc	Log Reg	HEAAN	3.80	10,230	0.993
	Chi2 test	pkg: PALISADE	0.09	1,512	0.983
SNU	SNU 1	HEAAN	52.49	15,204	0.984
	SNU 2		52.37	15,177	0.988
IBM	IBM-Complex	HEAAN	23.35	8,651	0.911
	IBM- Real	pkg: HElIb	52.65	15,613	0.526



iDASH 2018 – Semi-Parallel GWAS

- Compute Genome Wide Association Studies (GWAS)
- 3 Co-variants [age, height, weight] + 14,841 SNPS

Team	Submission	Schemes	Time (mins)	Memory (MB)	Accuracy
A*FHE	A*FHE 1	HEAAN	922.48	3,777	0.999
	A*FHE 2		1,632.97	4,093	0.905
Chimera	Version 1	TFHE+HEAAN (Chimera)	201.73	10,375	0.993
	Version 2		215.95	15,166	0.35
Delft Blue	Delft Blue	HEAAN	1,844.82	10,814	0.969
UCSD	Log Reg	HEAAN	1.66	14,901	0.993
	Lin Reg	pkg: RNS HEAAN	0.42	3,387	0.989
Duality Inc	Log Reg	HEAAN	3.80	10,230	0.993
	Chi2 test	pkg: PALISADE	0.09	1,512	0.983
SNU	SNU 1	HEAAN	52.49	15,204	0.984
	SNU 2		52.37	15,177	0.988
IBM	IBM-Complex	HEAAN	23.35	8,651	0.911
	IBM- Real	pkg: HElIb	52.65	15,613	0.526



Repeat logistic regression *n* times

Table of Contents

Approximate Computation

- Numerical Representation

  Encode m into an integer $m \approx px$ for a scaling factor p. $\sqrt{2} \mapsto 1412 \approx \sqrt{2} \cdot 10^3$

Approximate Computation

- Numerical Representation

  Encode m into an integer $m \approx px$ for a scaling factor p. $\sqrt{2} \mapsto 1412 \approx \sqrt{2} \cdot 10^3$

- Fixed-Point Multiplication

  Compute $m = m_1 m_2$ and extract its significant digits $m' \approx p^{-1} \cdot m$

  $1.234 \times 5.678 = (1234 \cdot 10^{-3}) \times (5678 \cdot 10^{-3}) = 7006652 \cdot 10^{-6} \mapsto 7007 \cdot 10^{-3} = 7.007$

Approximate Computation

- Numerical Representation

  Encode m into an integer $m \approx px$ for a scaling factor p. $\sqrt{2} \mapsto 1412 \approx \sqrt{2} \cdot 10^3$

- Fixed-Point Multiplication

  Compute $m = m_1 m_2$ and extract its significant digits $m' \approx p^{-1} \cdot m$

  $1.234 \times 5.678 = (1234 \cdot 10^{-3}) \times (5678 \cdot 10^{-3}) = 7006652 \cdot 10^{-6} \mapsto 7007 \cdot 10^{-3} = 7.007$

- LWE problem (Regev, 2005)

  $(b, \vec{a})$ such that $\langle (b, \vec{a}), (1, \vec{s}) \rangle = e \pmod{q}$

Approximate Computation

- Numerical Representation

  Encode m into an integer $m \approx px$ for a scaling factor p. $\sqrt{2} \mapsto 1412 \approx \sqrt{2} \cdot 10^3$

- Fixed-Point Multiplication

  Compute $m = m_1 m_2$ and extract its significant digits $m' \approx p^{-1} \cdot m$

  $1.234 \times 5.678 = (1234 \cdot 10^{-3}) \times (5678 \cdot 10^{-3}) = 7006652 \cdot 10^{-6} \mapsto 7007 \cdot 10^{-3} = 7.007$

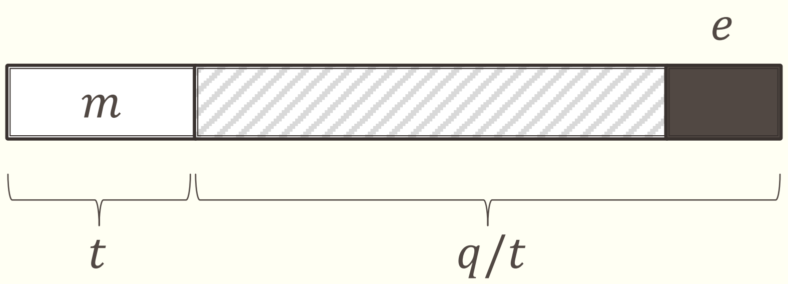
- LWE problem (Regev, 2005)

  $(b, \vec{a})$ such that $\langle (b, \vec{a}), (1, \vec{s}) \rangle = e \pmod q$

- Previous HE

  $\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m)$, $\langle \mathrm{ct}, \mathrm{sk} \rangle = \frac{q}{t} m + e \pmod q$

  Modulo t plaintext vs Rounding operation

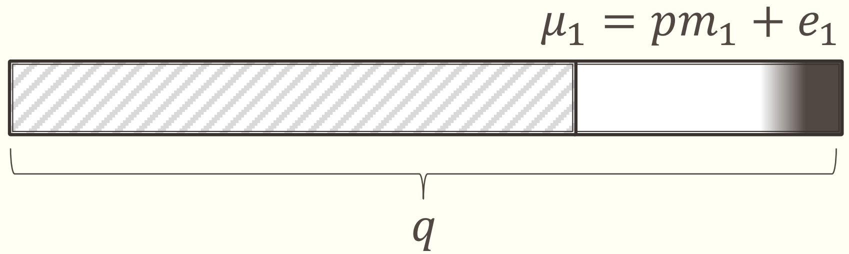


HEAAN

- A New Message Encoding

$$\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m), \quad \langle \mathrm{ct}, \mathrm{sk} \rangle = pm + e \pmod{q}$$

Consider e as part of approximation error



HEAAN

- A New Message Encoding

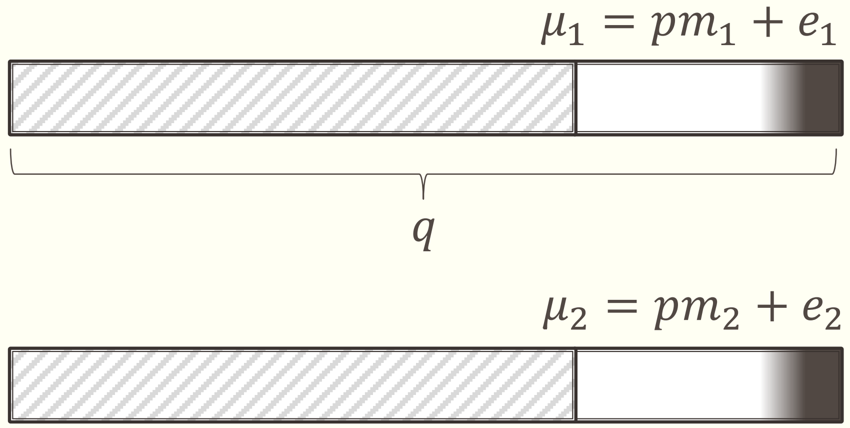
  $$\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m), \quad \langle \mathrm{ct}, \mathrm{sk} \rangle = pm + e \pmod q$$

  Consider e as part of approximation error

- Homomorphic Operations

  Input: $\mu_1 \approx pm_1,\ \mu_2 \approx pm_2$

  Addition: $\mu_1 + \mu_2 \approx p \cdot (m_1 + m_2)$



HEAAN

- A New Message Encoding

  $$\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m), \quad \langle \mathrm{ct}, \mathrm{sk} \rangle = pm + e \pmod{q}$$

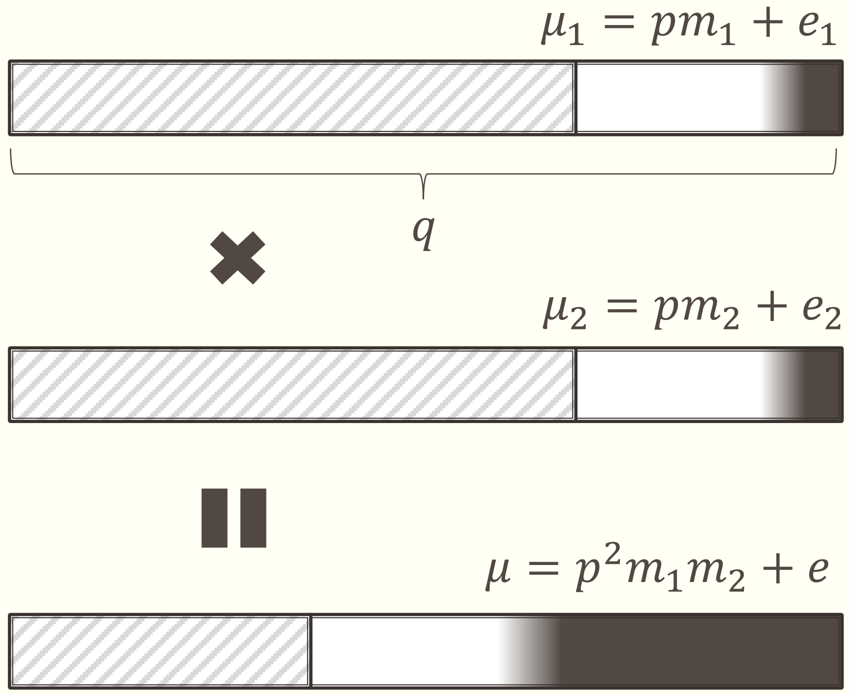
  Consider e as part of approximation error

- Homomorphic Operations

  Input: $\mu_1 \approx pm_1,\ \mu_2 \approx pm_2$

  Addition: $\mu_1 + \mu_2 \approx p \cdot (m_1 + m_2)$

  Multiplication: $\mu = \mu_1\mu_2 \approx p^2 \cdot m_1 m_2$



HEAAN

- A New Message Encoding

  $\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m), \quad \langle \mathrm{ct}, \mathrm{sk} \rangle = pm + e \pmod{q}$

  Consider e as part of approximation error

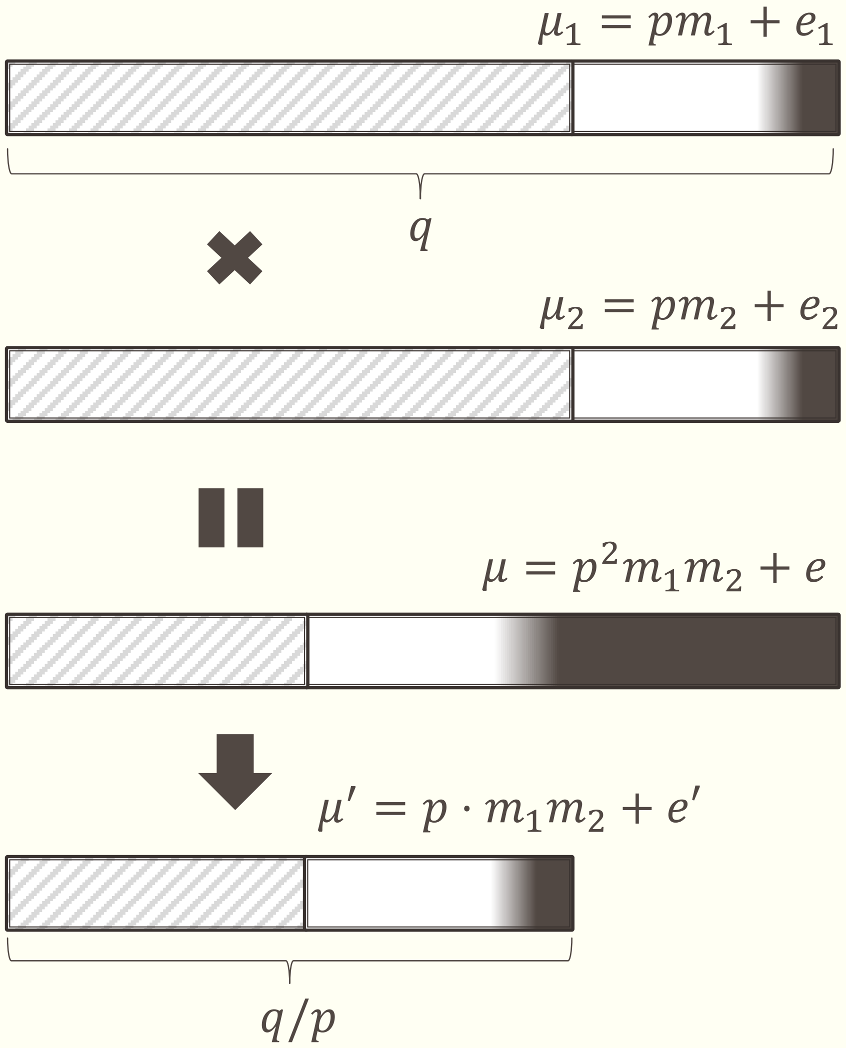
- Homomorphic Operations

  Input: $\mu_1 \approx pm_1,\ \mu_2 \approx pm_2$

  Addition: $\mu_1 + \mu_2 \approx p \cdot (m_1 + m_2)$

  Multiplication: $\mu = \mu_1 \mu_2 \approx p^2 \cdot m_1 m_2$

  Rounding: $\mu' \approx p^{-1} \cdot \mu \approx p \cdot m_1 m_2$



HEAAN

- A New Message Encoding

  $\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m), \quad \langle \mathrm{ct}, \mathrm{sk} \rangle = pm + e \pmod{q}$

  Consider e as part of approximation error

- Homomorphic Operations

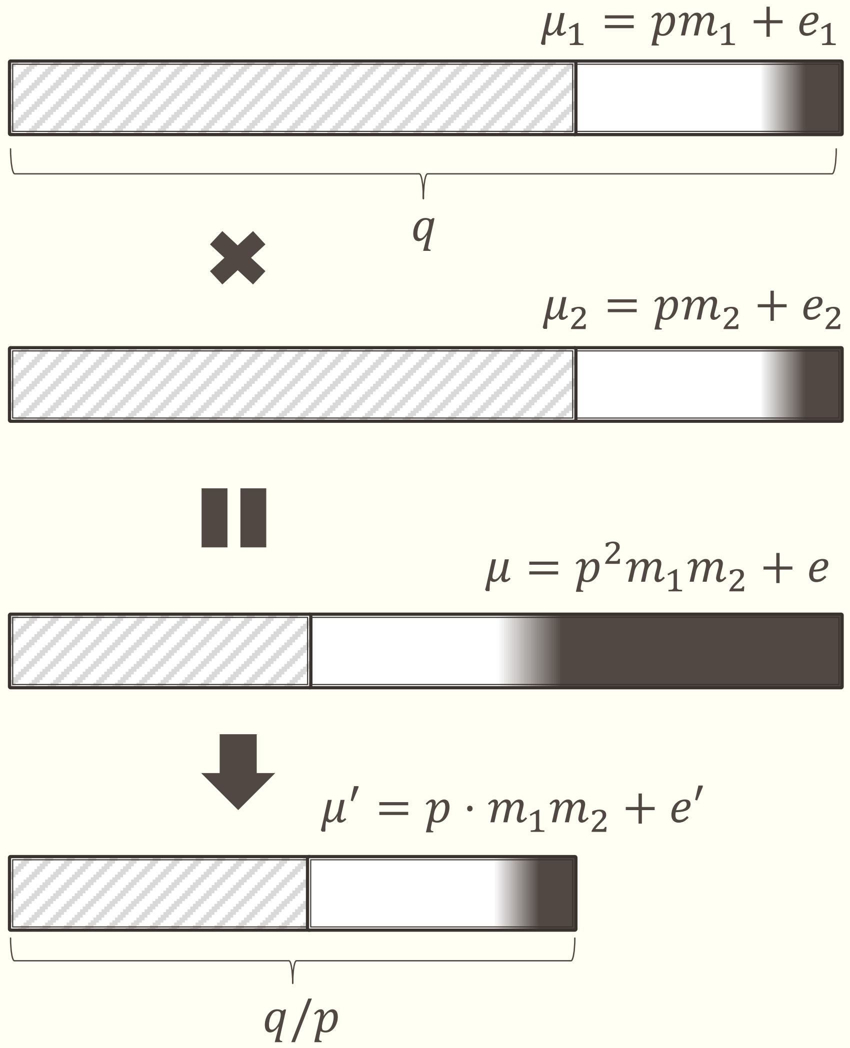
  Input: $\mu_1 \approx pm_1,\ \mu_2 \approx pm_2$

  Addition: $\mu_1 + \mu_2 \approx p \cdot (m_1 + m_2)$

  Multiplication: $\mu = \mu_1\mu_2 \approx p^2 \cdot m_1 m_2$

  Rounding: $\mu' \approx p^{-1} \cdot \mu \approx p \cdot m_1 m_2$

- Support for the (approximate) fixed-point arithmetic !
- **Leveled** HE : $q = p^L$



Packed Ciphertext

- Construction over the ring $R = \mathbb{Z}[X]/(X^n + 1)$ and $R_q = R \pmod q$

Packed Ciphertext

- Construction over the ring $R = \mathbb{Z}[X]/(X^n + 1)$ and $R_q = R \pmod q$
- Packing Technique:
  - A single ciphertext can encrypt a vector of plaintext values $z = (z_1, z_2, \ldots, z_\ell)$
  - Parallel computation in a SIMD manner $z \otimes w = (z_1 w_1, z_2 w_2, \ldots, z_\ell w_\ell)$

Packed Ciphertext

- Construction over the ring $R = \mathbb{Z}[X]/(X^n + 1)$ and $R_q = R \pmod q$
- Packing Technique:
  - A single ciphertext can encrypt a vector of plaintext values $z = (z_1, z_2, \ldots, z_\ell)$
  - Parallel computation in a SIMD manner $z \otimes w = (z_1 w_1, z_2 w_2, \ldots, z_\ell w_\ell)$
- RLWE-based HEAAN
  - A ciphertext can encrypt a polynomial $m(X) \in R$
  - Observation: $X^n + 1 = (X - \zeta_1)(X - \zeta_1^{-1})(X - \zeta_2)(X - \zeta_2^{-1}) \ldots (X - \zeta_{n/2})(X - \zeta_{n/2}^{-1})$

Packed Ciphertext

- Construction over the ring $R = \mathbb{Z}[X]/(X^n + 1)$ and $R_q = R \pmod q$
- Packing Technique:
  - A single ciphertext can encrypt a vector of plaintext values $z = (z_1, z_2, \dots, z_\ell)$
  - Parallel computation in a SIMD manner $z \otimes w = (z_1 w_1, z_2 w_2, \dots, z_\ell w_\ell)$
- RLWE-based HEAAN
  - A ciphertext can encrypt a polynomial $m(X) \in R$
  - Observation: $X^n + 1 = (X - \zeta_1)(X - \zeta_1^{-1})(X - \zeta_2)(X - \zeta_2^{-1}) \dots (X - \zeta_{n/2})(X - \zeta_{n/2}^{-1})$
  - Decoding/Encoding function

$$R = \mathbb{Z}[X]/(X^n + 1) \subseteq \mathbb{R}[X]/(X^n + 1) \ \rightarrow \ \mathbb{C}^{n/2}$$

$$m(X) \ \mapsto \ z = \left(z_1, \dots, z_{n/2}\right), \ z_i = \mu(\zeta_i)$$

Packed Ciphertext

- Construction over the ring $R = \mathbb{Z}[X]/(X^n + 1)$ and $R_q = R \pmod q$
- Packing Technique:
  - A single ciphertext can encrypt a vector of plaintext values $z = (z_1, z_2, \ldots, z_\ell)$
  - Parallel computation in a SIMD manner $z \otimes w = (z_1 w_1, z_2 w_2, \ldots, z_\ell w_\ell)$
- RLWE-based HEAAN
  - A ciphertext can encrypt a polynomial $m(X) \in R$
  - Observation: $X^n + 1 = (X - \zeta_1)(X - \zeta_1^{-1})(X - \zeta_2)(X - \zeta_2^{-1}) \ldots (X - \zeta_{n/2})(X - \zeta_{n/2}^{-1})$
  - Decoding/Encoding function

$$R = \mathbb{Z}[X]/(X^n + 1) \subseteq \mathbb{R}[X]/(X^n + 1) \;\; \rightarrow \;\; \mathbb{C}^{n/2}$$

$$m(X) \;\; \mapsto \;\; z = \left(z_1, \ldots, z_{n/2}\right), \; z_i = \mu(\zeta_i)$$

  - Example: $n = 4, \; \zeta_1 = \exp(\pi i/4), \; \zeta_2 = \exp(5\pi i/4)$

$$z = (1 - 2i, 3 + 4i) \;\; \mapsto m(X) = 2 - 2\sqrt{2}\, X + X^2 - \sqrt{2}\, X^3$$

$$\mapsto \mu(X) = 2000 - 2828X + 1000X^2 - 1414\, X^3$$

$$\mu(\zeta_1) \approx 1000.15 - 1999.55\, i, \;\; \mu(\zeta_2) \approx 2999.85 + 3999.55\, i$$

Summary

- HEAAN natively support for the (approximate) fixed point arithmetic

- Ciphertext modulus $\log q = L \log p$ grows linearly
- Useful when evaluating analytic functions approximately:
  - Polynomial
  - Multiplicative Inverse
  - Trigonometric Functions
  - Exponential Function (Logistic Function, Sigmoid Function)
  - …
- Packing technique based on DFT
  - SIMD operation
  - Rotation on plaintext slots

$$z = \left(z_1, \dots, z_{n/2}\right) \mapsto z' = \left(z_2, \dots, z_{n/2}, z_1\right)$$

Table of Contents

Bootstrapping of HEAAN

- Bootstrapping
  - Ciphertexts of a leveled HE have a limited lifespan

Bootstrapping of HEAAN

- Bootstrapping
  - Ciphertexts of a leveled HE have a limited lifespan
  - Refresh a ciphertext $\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m)$ by **evaluating the decryption circuit homomorphically**

$$\mathrm{Dec}_{\mathrm{sk}}(\mathrm{ct}) = m \iff F_{\mathrm{ct}}(\mathrm{sk}) = m \text{ where } F_{\mathrm{ct}}(*) = \mathrm{Dec}_{*}(\mathrm{ct})$$

Bootstrapping of HEAAN

- Bootstrapping
  - Ciphertexts of a leveled HE have a limited lifespan
  - Refresh a ciphertext $\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m)$ by **evaluating the decryption circuit homomorphically**

$$\mathrm{Dec}_{\mathrm{sk}}(\mathrm{ct}) = m \iff F_{\mathrm{ct}}(\mathrm{sk}) = m \text{ where } F_{\mathrm{ct}}(*) = \mathrm{Dec}_{*}(\mathrm{ct})$$

  - Bootstrapping key $\mathrm{BK} = \mathrm{Enc}_{\mathrm{sk}}(\mathrm{sk})$

$$F_{\mathrm{ct}}(\mathrm{BK}) = F_{\mathrm{ct}}\big(\mathrm{Enc}_{\mathrm{sk}}(\mathrm{sk})\big) = \mathrm{Enc}_{\mathrm{sk}}\big(F_{\mathrm{ct}}(\mathrm{sk})\big) = \mathrm{Enc}_{\mathrm{sk}}(m)$$

Bootstrapping of HEAAN

- Bootstrapping
  - Ciphertexts of a leveled HE have a limited lifespan
  - Refresh a ciphertext $\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m)$ by **evaluating the decryption circuit homomorphically**

$$\mathrm{Dec}_{\mathrm{sk}}(\mathrm{ct}) = m \iff F_{\mathrm{ct}}(\mathrm{sk}) = m \text{ where } F_{\mathrm{ct}}(*) = \mathrm{Dec}_{*}(\mathrm{ct})$$

  - Bootstrapping key $\mathrm{BK} = \mathrm{Enc}_{\mathrm{sk}}(\mathrm{sk})$

$$F_{\mathrm{ct}}(\mathrm{BK}) = F_{\mathrm{ct}}\big(\mathrm{Enc}_{\mathrm{sk}}(\mathrm{sk})\big) = \mathrm{Enc}_{\mathrm{sk}}\big(F_{\mathrm{ct}}(\mathrm{sk})\big) = \mathrm{Enc}_{\mathrm{sk}}(m)$$

- HEAAN
  - Homomorphic operations introduce errors

$$F_{\mathrm{ct}}(\mathrm{BK}) = F_{\mathrm{ct}}\big(\mathrm{Enc}_{\mathrm{sk}}(\mathrm{sk})\big) = \mathrm{Enc}_{\mathrm{sk}}(F_{\mathrm{ct}}(\mathrm{sk}) + e) = \mathrm{Enc}_{\mathrm{sk}}(m + e)$$

  - It is ok to have an additional error

Bootstrapping of HEAAN

- Bootstrapping
  - Ciphertexts of a leveled HE have a limited lifespan
  - Refresh a ciphertext $\mathrm{ct} = \mathrm{Enc}_{\mathrm{sk}}(m)$ by **evaluating the decryption circuit homomorphically**

$$\mathrm{Dec}_{\mathrm{sk}}(\mathrm{ct}) = m \iff F_{\mathrm{ct}}(\mathrm{sk}) = m \ \text{ where } \ F_{\mathrm{ct}}(*) = \mathrm{Dec}_{*}(\mathrm{ct})$$

  - Bootstrapping key $\mathrm{BK} = \mathrm{Enc}_{\mathrm{sk}}(\mathrm{sk})$

$$F_{\mathrm{ct}}(\mathrm{BK}) = F_{\mathrm{ct}}\big(\mathrm{Enc}_{\mathrm{sk}}(\mathrm{sk})\big) = \mathrm{Enc}_{\mathrm{sk}}\big(F_{\mathrm{ct}}(\mathrm{sk})\big) = \mathrm{Enc}_{\mathrm{sk}}(m)$$

- HEAAN
  - Homomorphic operations introduce errors

$$F_{\mathrm{ct}}(\mathrm{BK}) = F_{\mathrm{ct}}\big(\mathrm{Enc}_{\mathrm{sk}}(\mathrm{sk})\big) = \mathrm{Enc}_{\mathrm{sk}}(F_{\mathrm{ct}}(\mathrm{sk}) + e) = \mathrm{Enc}_{\mathrm{sk}}(m + e)$$

  - It is ok to have an additional error
  - How to evaluate the decryption circuit (efficiently)?

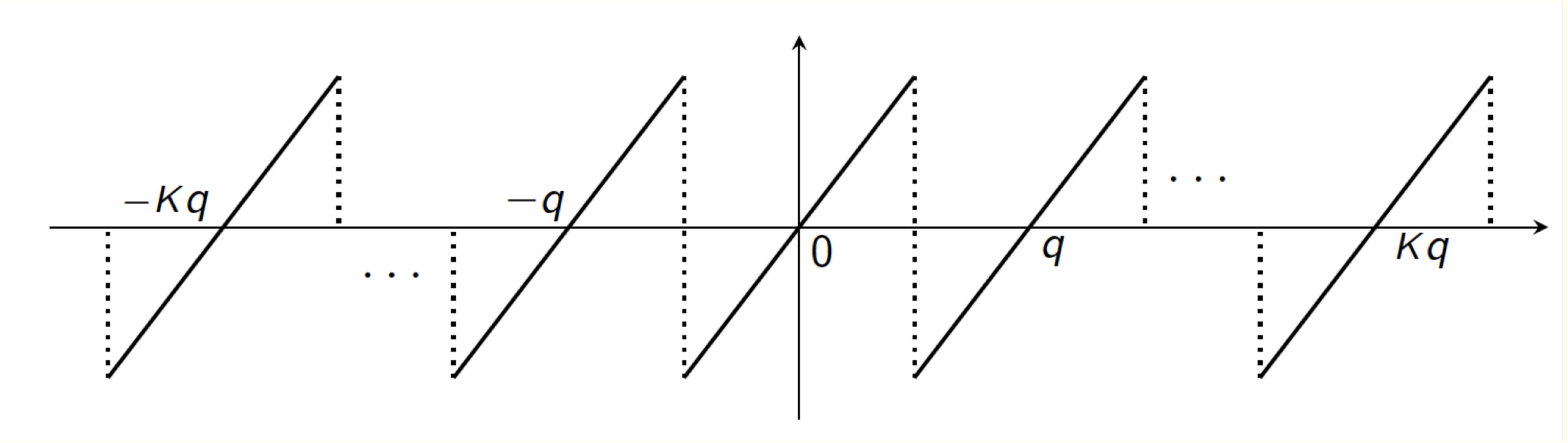
$$\mathrm{Dec}_{\mathrm{sk}}(\mathrm{ct}) = \langle \mathrm{ct}, \mathrm{sk} \rangle \pmod q$$

Approximate Decryption

$$\mathrm{Dec}_{\mathrm{sk}}(\mathrm{ct}) \;\mapsto\; t = \langle \mathrm{ct}, \mathrm{sk} \rangle \;\mapsto\; [t]_q = \mu,$$

$$t = qI + \mu \text{ for some } |I| < K$$

- Naïve solution: polynomial interpolation on $[-Kq, Kq]$
- Huge depth, complexity & inaccurate result

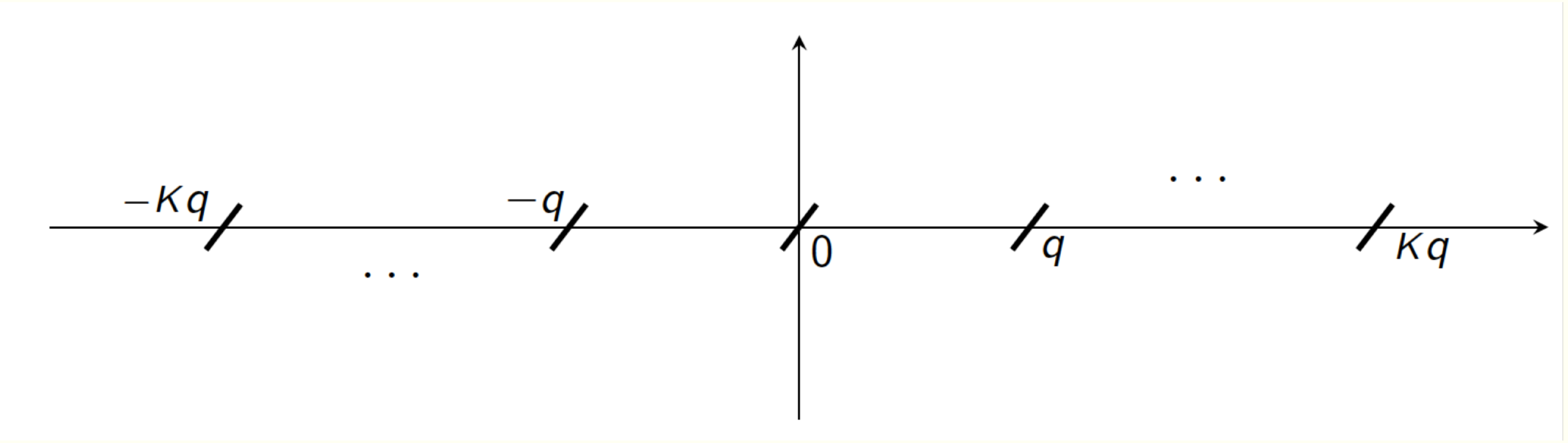


Approximate Decryption

$$\mathrm{Dec}_{\mathrm{sk}}(\mathrm{ct}) \;\mapsto\; t = \langle \mathrm{ct}, \mathrm{sk} \rangle \;\mapsto\; [t]_q = \mu,$$

$$t = qI + \mu \text{ for some } |I| < K$$

- Idea 1: Restriction of domain $|\mu| \ll q$

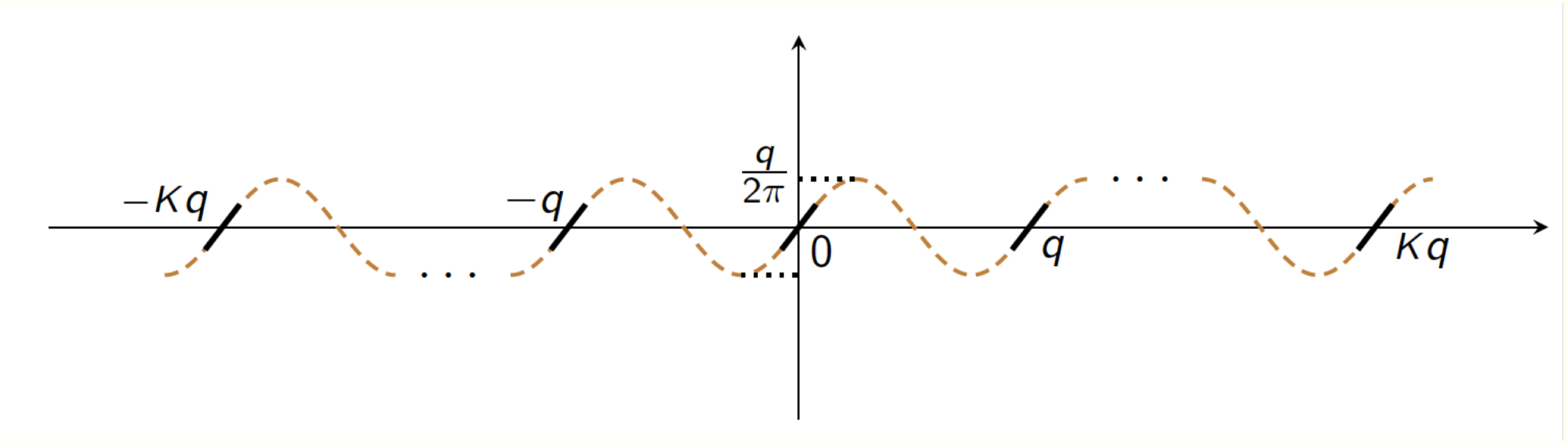


Approximate Decryption

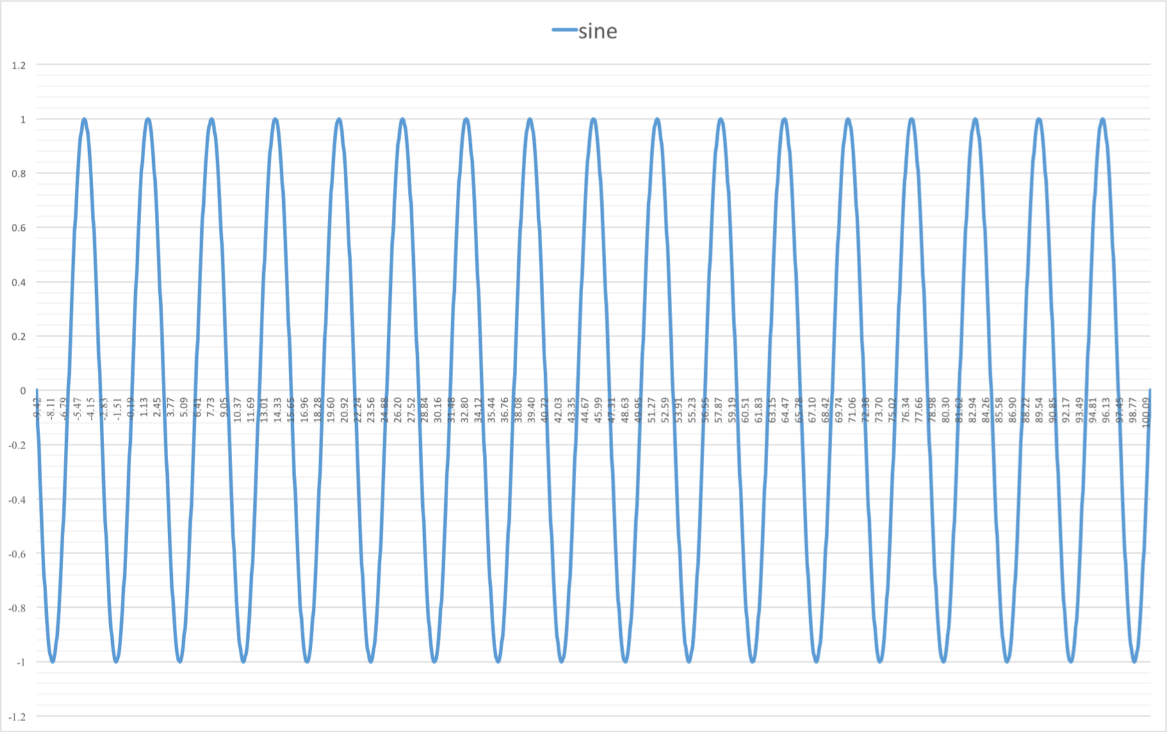
$$\mathrm{Dec}_{\mathrm{sk}}(\mathrm{ct}) \;\mapsto\; t = \langle \mathrm{ct}, \mathrm{sk} \rangle \;\mapsto\; [t]_q = \mu,$$

$$t = qI + \mu \text{ for some } |I| < K$$

- Idea 1: Restriction of domain $|\mu| \ll q$
- Idea 2: Sine approximation $\mu \approx \frac{q}{2\pi} \sin\theta$ for $\theta = \frac{2\pi}{q} t$

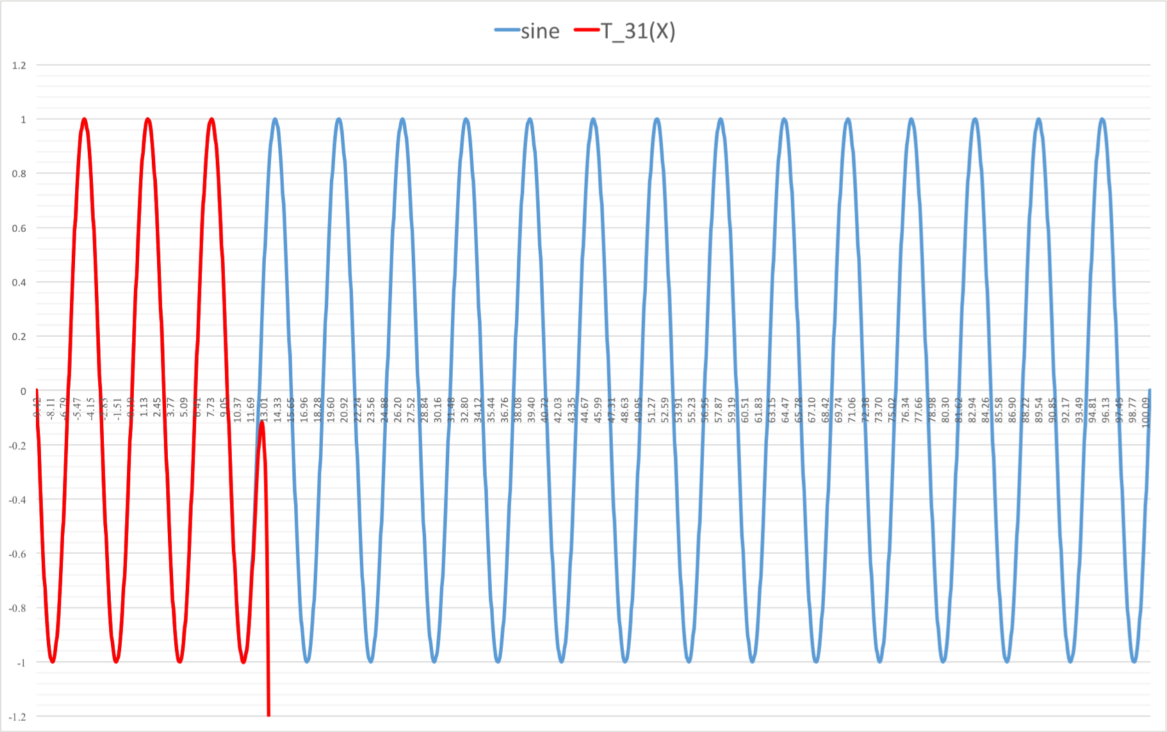


Sine Evaluation



Sine Evaluation

- Direct Taylor approximation
  - huge depth & complexity, low precision

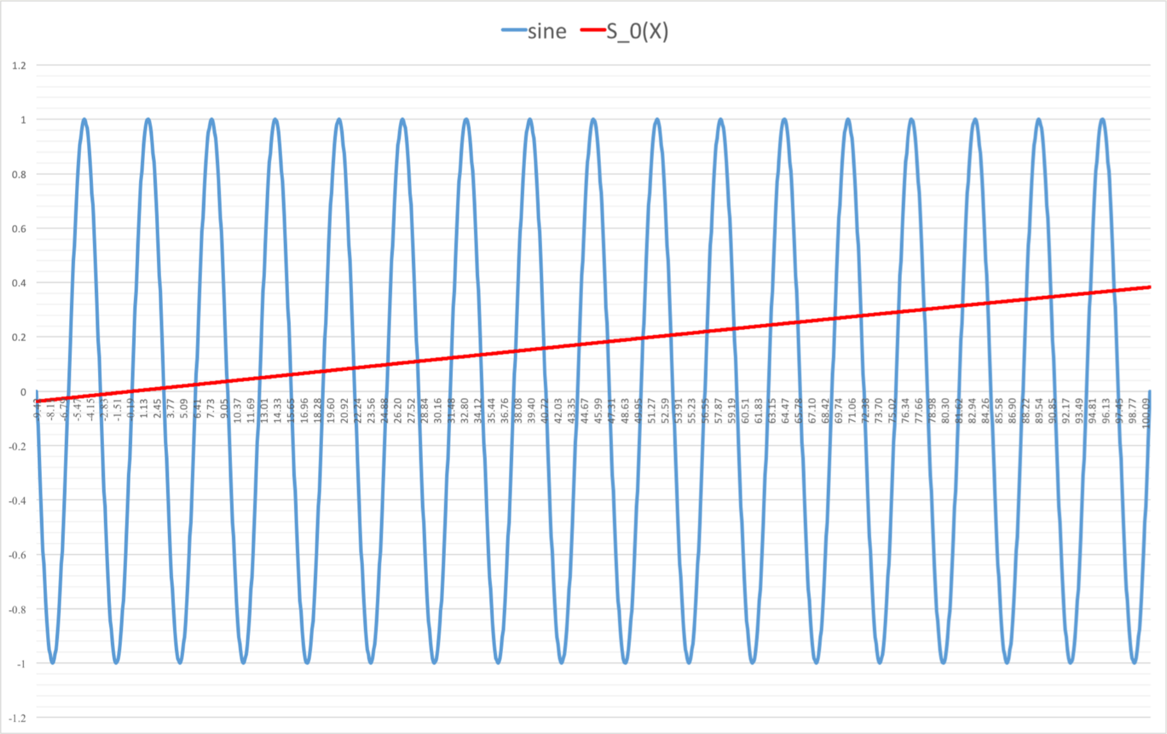


Sine Evaluation

- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!} (\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!} (\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$



Sine Evaluation

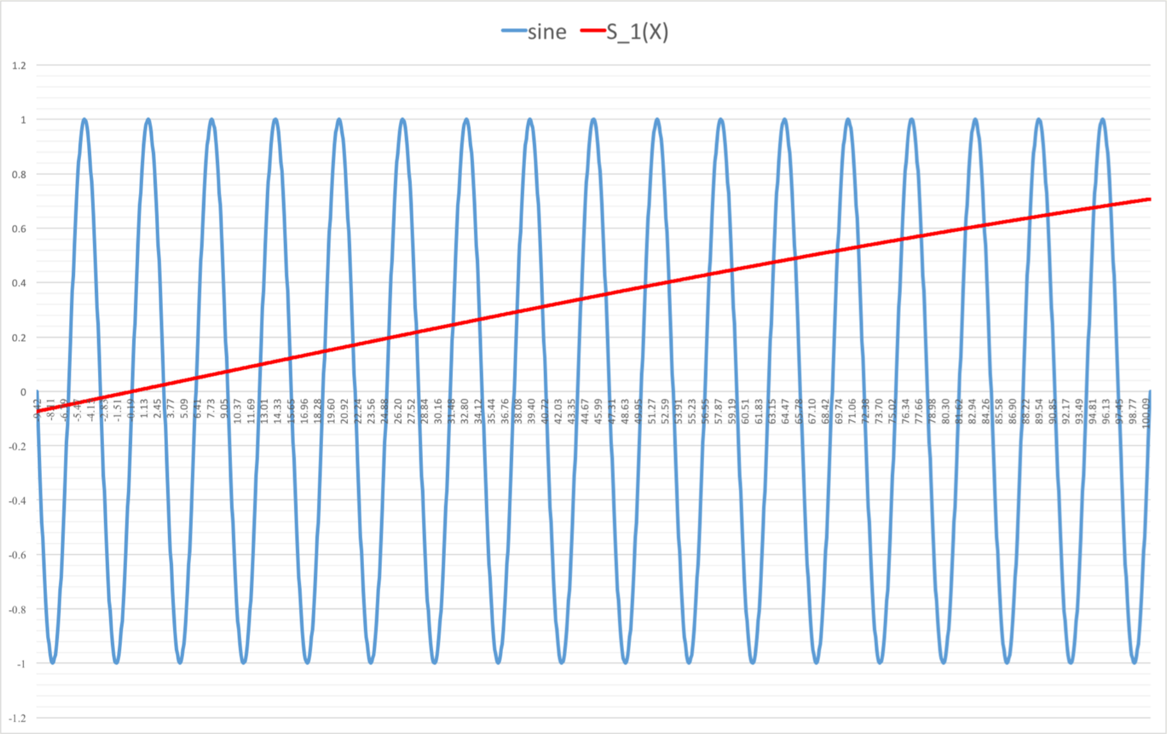
- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!} (\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!} (\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$

- Idea 2: Use double-angle formula

$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta),\ S_{k+1}(\theta) = 2S_k(\theta) \cdot C_k(\theta).$$



Sine Evaluation

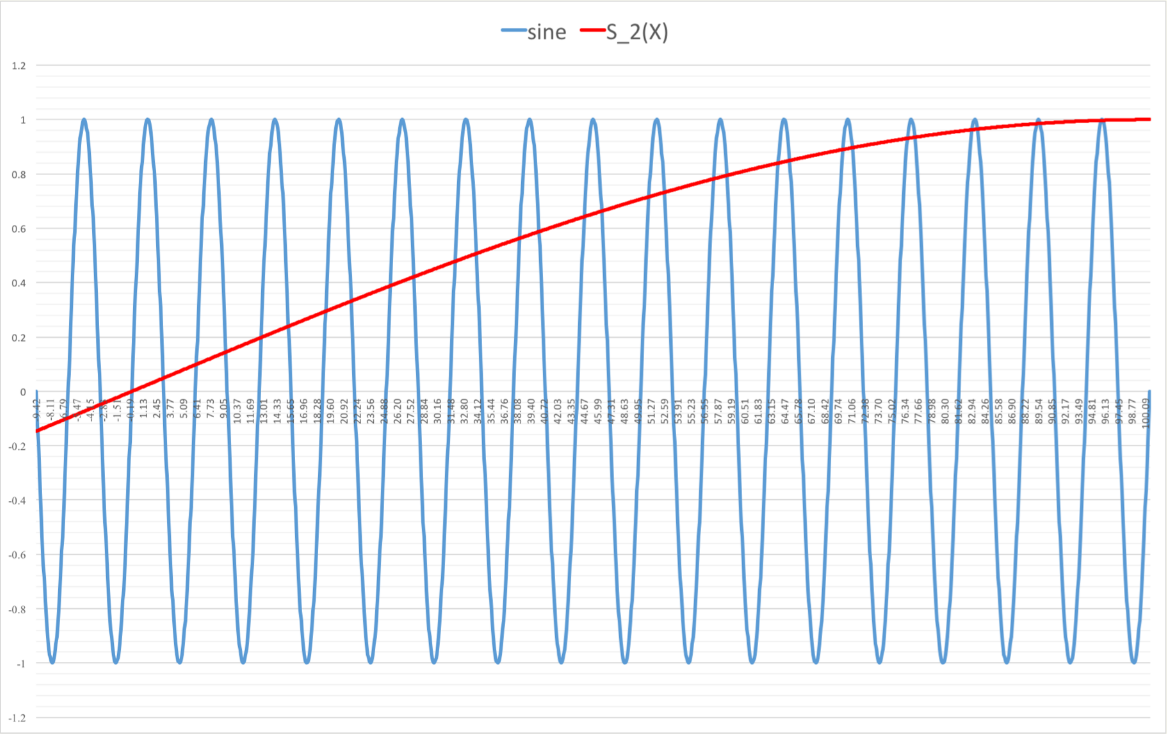
- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!} (\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!} (\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$

- Idea 2: Use double-angle formula

$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta),\ S_{k+1}(\theta) = 2S_k(\theta) \cdot C_k(\theta).$$



Sine Evaluation

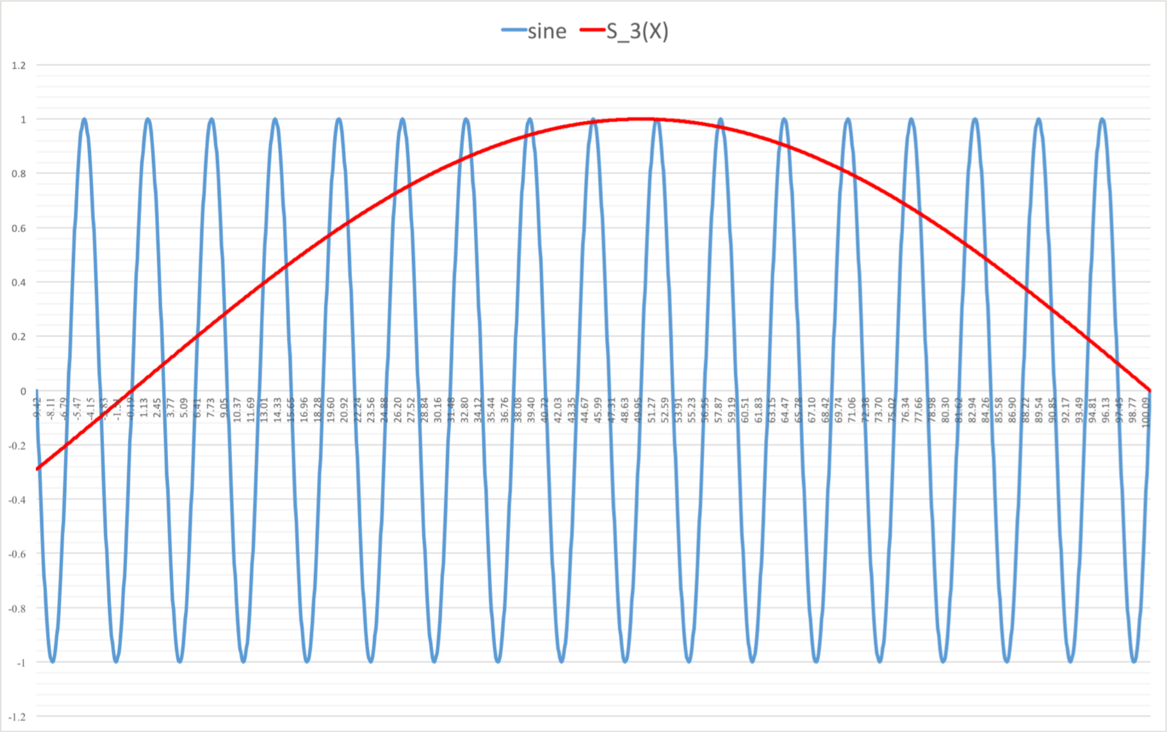
- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!}(\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!}(\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$

- Idea 2: Use double-angle formula

$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta),\; S_{k+1}(\theta) = 2S_k(\theta) \cdot C_k(\theta).$$



Sine Evaluation

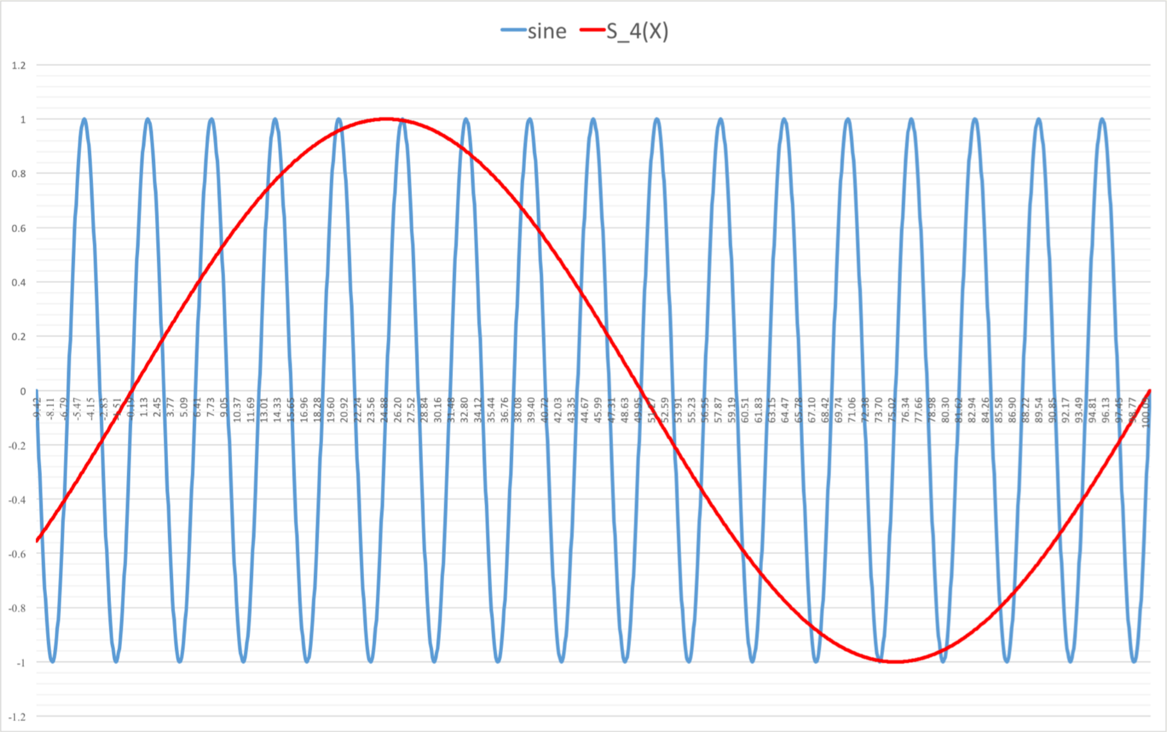
- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!} (\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!} (\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$

- Idea 2: Use double-angle formula

$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta),\ S_{k+1}(\theta) = 2S_k(\theta) \cdot C_k(\theta).$$



Sine Evaluation

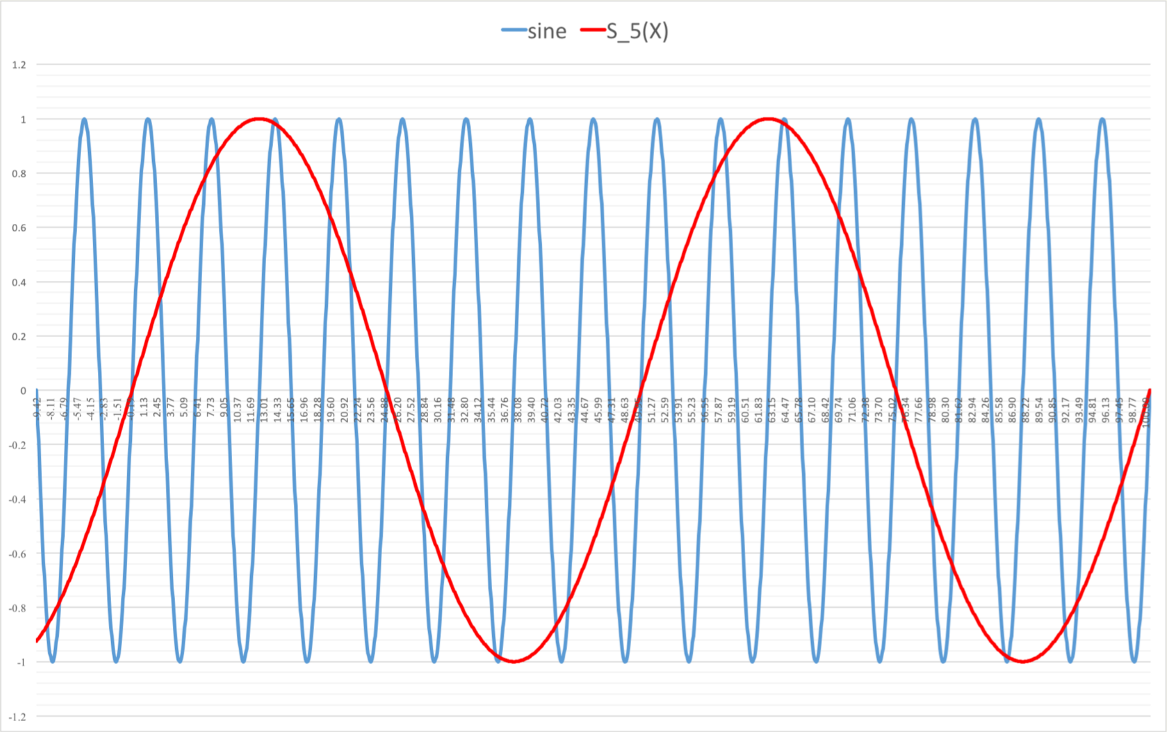
- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!}(\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!}(\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$

- Idea 2: Use double-angle formula

$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta),\ S_{k+1}(\theta) = 2S_k(\theta)\cdot C_k(\theta).$$



Sine Evaluation

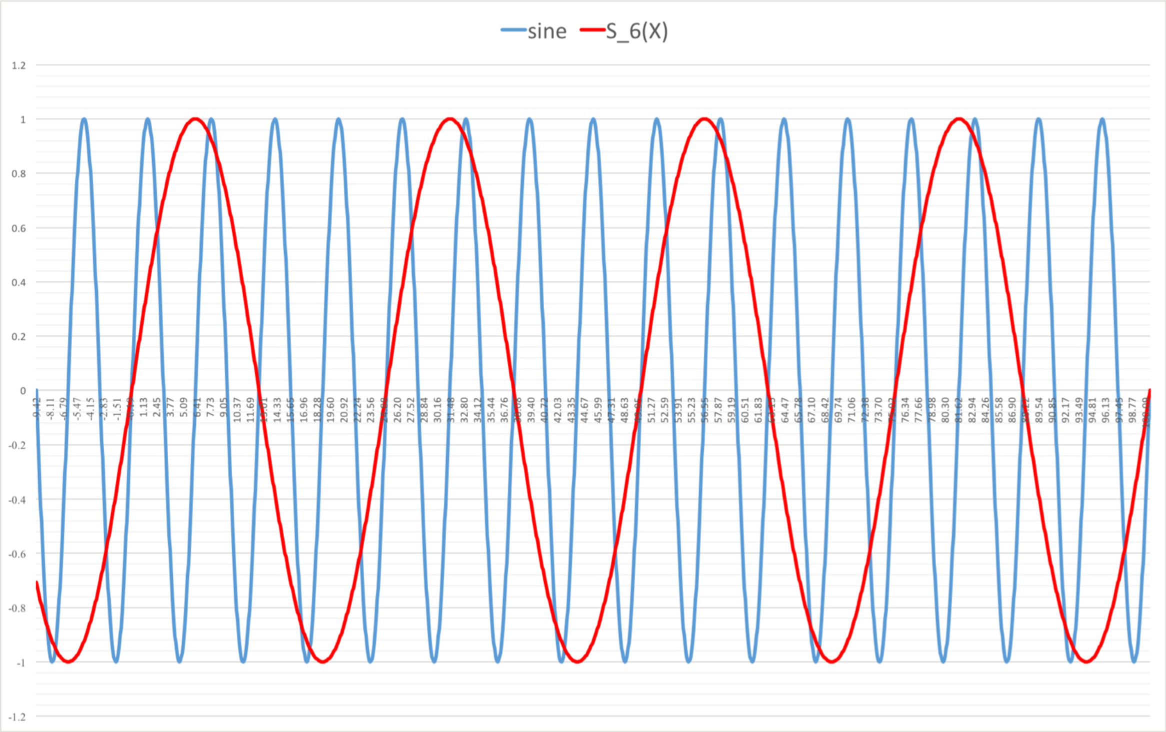
- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!}(\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!}(\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$

- Idea 2: Use double-angle formula

$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta),\ S_{k+1}(\theta) = 2S_k(\theta) \cdot C_k(\theta).$$



Sine Evaluation

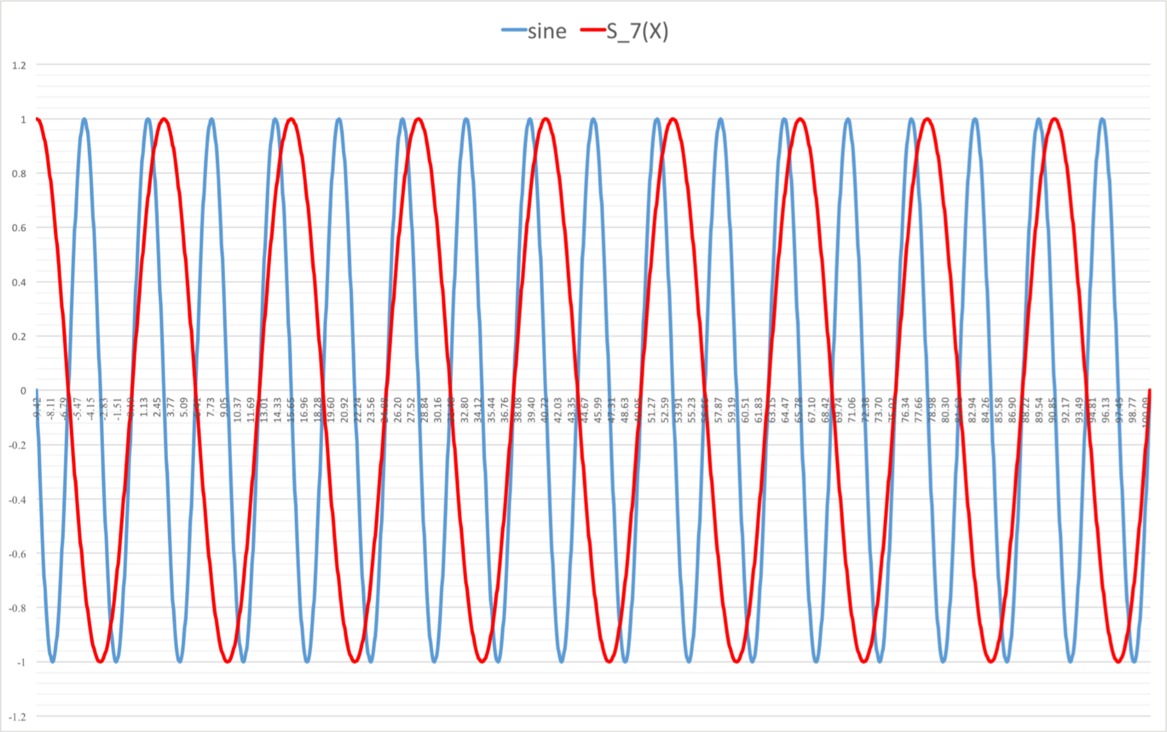
- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!} (\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!} (\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$

- Idea 2: Use double-angle formula

$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta), \; S_{k+1}(\theta) = 2S_k(\theta) \cdot C_k(\theta).$$



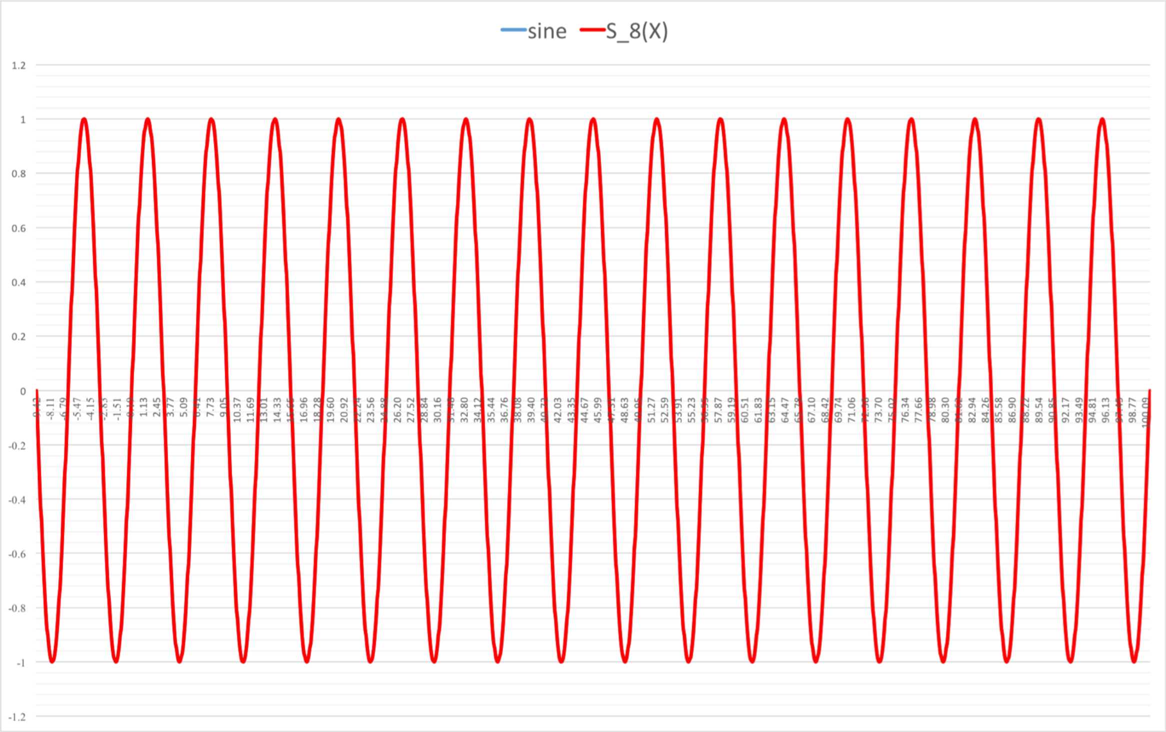
Sine Evaluation

- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!}(\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!}(\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$

- Idea 2: Use double-angle formula



$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta),\ S_{k+1}(\theta) = 2S_k(\theta) \cdot C_k(\theta).$$

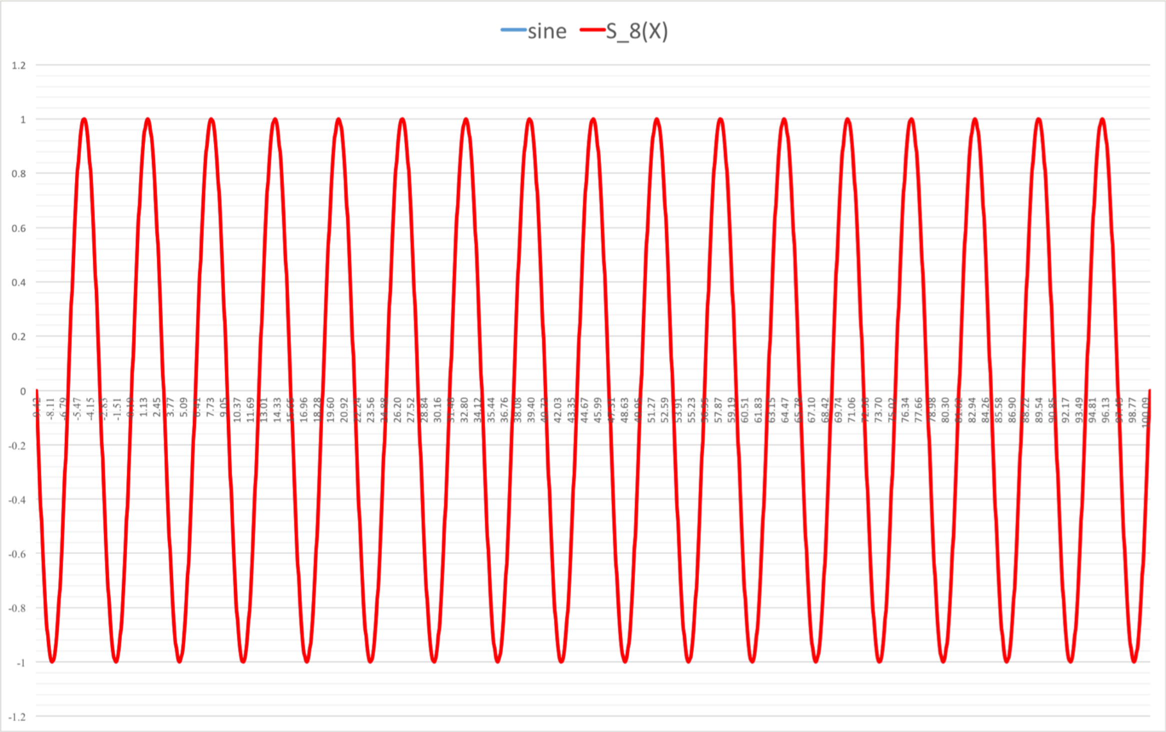
$$S_r(\theta) \approx \sin\theta$$

Sine Evaluation

- Direct Taylor approximation
  - huge depth & complexity, low precision
- Idea 1: Low-degree approximation of smooth functions

$$C_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k)!} (\theta/2^r)^{2k} \approx \cos(\theta/2^r),$$

$$S_0(\theta) = \sum_{k=0}^{d} \frac{(-1)^k}{(2k+1)!} (\theta/2^r)^{2k+1} \approx \sin(\theta/2^r).$$



- Idea 2: Use double-angle formula

$$C_{k+1}(\theta) = C_k^2(\theta) - S_k^2(\theta),\ S_{k+1}(\theta) = 2S_k(\theta) \cdot C_k(\theta).$$

$$S_r(\theta) \approx \sin\theta$$

- **Numerically stable & Linear complexity**

Slot-Coefficient Switching

- Ring-based HEAAN
  - Homomorphic operations on plaintext slots, not on coefficients
  - We need to perform the modulo reduction on coefficients

Slot-Coefficient Switching

- Ring-based HEAAN
  - Homomorphic operations on plaintext slots, not on coefficients
  - We need to perform the modulo reduction on coefficients
- Pre/post computation before/after sine evaluation

Coefficient	Plaintext Slots
$\mathbb{R}[X]/(X^n + 1)$	$\mathbb{C}^{n/2}$
$t(X) = q \cdot I(X) + \mu(X)$	
1. coeff to slot $\longrightarrow$	$(t_0, t_1, \dots t_{n-1})$

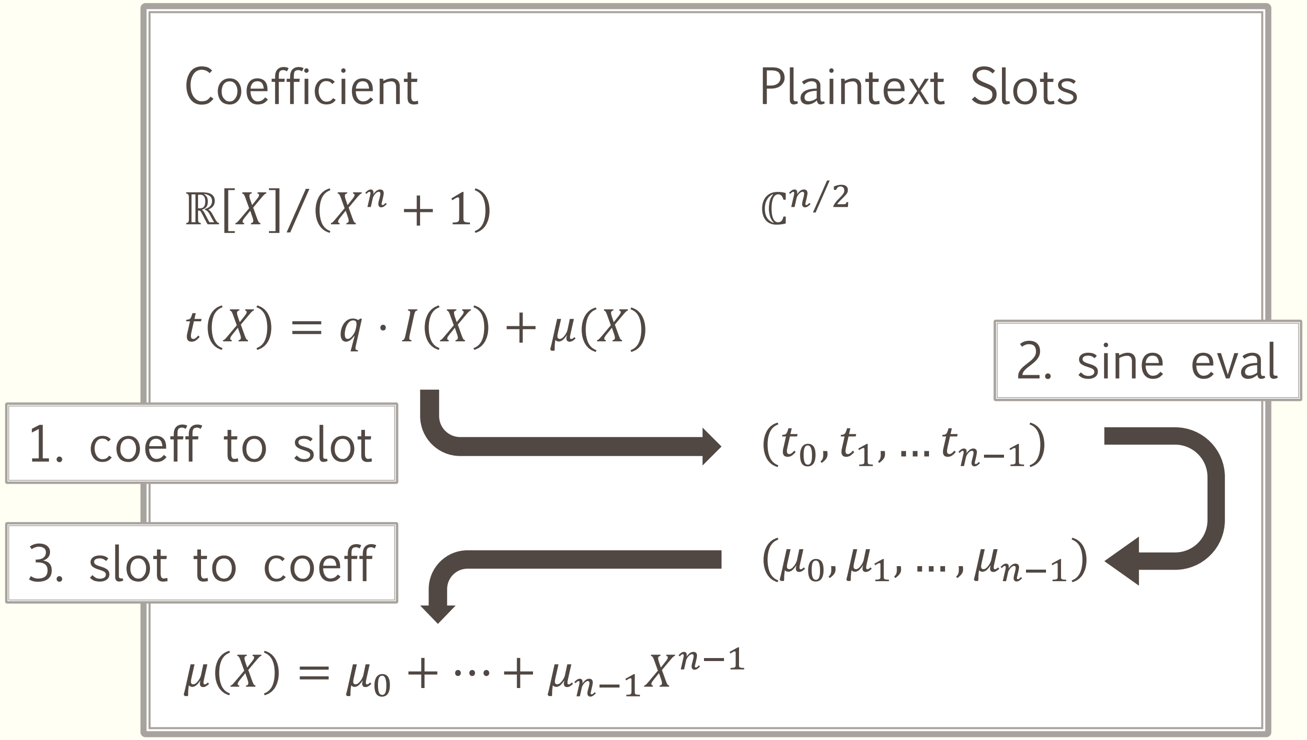
Slot-Coefficient Switching

- Ring-based HEAAN
  - Homomorphic operations on plaintext slots, not on coefficients
  - We need to perform the modulo reduction on coefficients
- Pre/post computation before/after sine evaluation

Coefficient	Plaintext Slots
$\mathbb{R}[X]/(X^n+1)$	$\mathbb{C}^{n/2}$
$t(X) = q \cdot I(X) + \mu(X)$	
1. coeff to slot $\rightarrow$	$(t_0, t_1, \dots t_{n-1})$
	2. sine eval
	$(\mu_0, \mu_1, \dots, \mu_{n-1})$

Slot-Coefficient Switching

- Ring-based HEAAN
  - Homomorphic operations on plaintext slots, not on coefficients
  - We need to perform the modulo reduction on coefficients
- Pre/post computation before/after sine evaluation



Slot-Coefficient Switching

- Ring-based HEAAN
  - Homomorphic operations on plaintext slots, not on coefficients
  - We need to perform the modulo reduction on coefficients
- Pre/post computation before/after sine evaluation
- Performance of Bootstrapping
  - Depth consumption : Sine evaluation
  - Complexity: Slot-Coefficient switchings (# of slots)
- Experimental Results
  - 127 + 12 = 139 s / 128 slots X 12 bits
  - 456 + 68 = 524 s / 128 slots X 24 bits

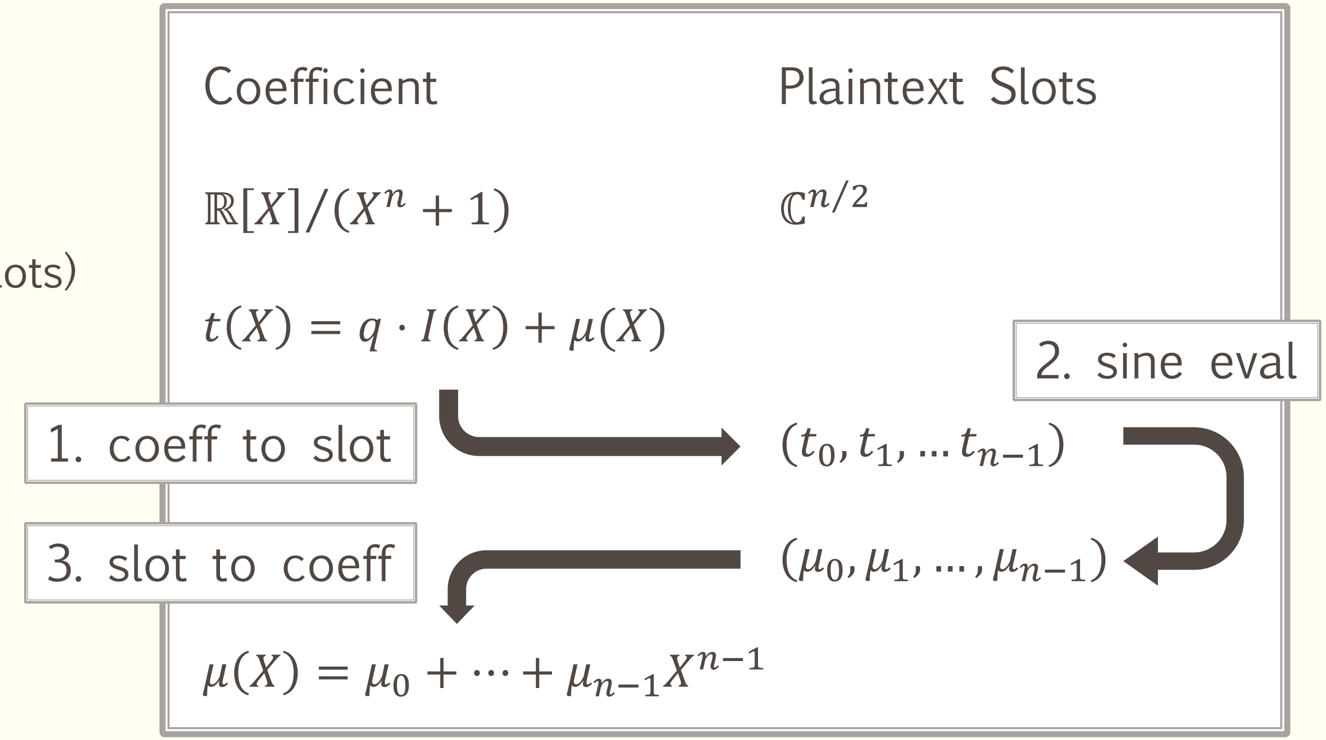


Table of Contents

- ~~Background~~
- ~~Construction~~
  - [CKKS, AC17] Homomorphic Encryption for Arithmetic of Approximate Numbers
- ~~Bootstrapping~~
  - [CHKKS, EC18] Bootstrapping for Approximate Homomorphic Encryption
- **Related Works**

Followed Work

- Improved Bootstrapping for Approximate Homomorphic Encryption
    - Joint work with Hao Chen and Ilaria Chillotti (submission to EC19)
    - FFT-like algorithms to optimize Slot-Coefficient switchings
    - Better evaluation of sine function based on Chebyshev approximation

Followed Work

- Improved Bootstrapping for Approximate Homomorphic Encryption
  - Joint work with Hao Chen and Ilaria Chillotti (submission to EC19)
  - FFT-like algorithms to optimize Slot-Coefficient switchings
  - Better evaluation of sine function based on Chebyshev approximation

- [JKLS, CCS18] Secure Outsourced Matrix Computation and Application to Neural Networks
  - Evaluation of an encrypted CNN model on the encrypted MNIST data
- [DSC+18] CHET, [BLW18] nGraph-HE : Automatic HE compilers for Deep Learning

Followed Work

- Improved Bootstrapping for Approximate Homomorphic Encryption
  - Joint work with Hao Chen and Ilaria Chillotti (submission to EC19)
  - FFT-like algorithms to optimize Slot-Coefficient switchings
  - Better evaluation of sine function based on Chebyshev approximation

- [JKLS, CCS18] Secure Outsourced Matrix Computation and Application to Neural Networks
  - Evaluation of an encrypted CNN model on the encrypted MNIST data

- [DSC+18] CHET, [BLW18] nGraph-HE : Automatic HE compilers for Deep Learning

- [CHKKS, SAC18] A Full RNS Variant of Approximate Homomorphic Encryption
  - Better performance without any high-precision arithmetic library
  - iDASH 2018

- [KS, ICISC18] Approximate Homomorphic Encryption over the Real Numbers