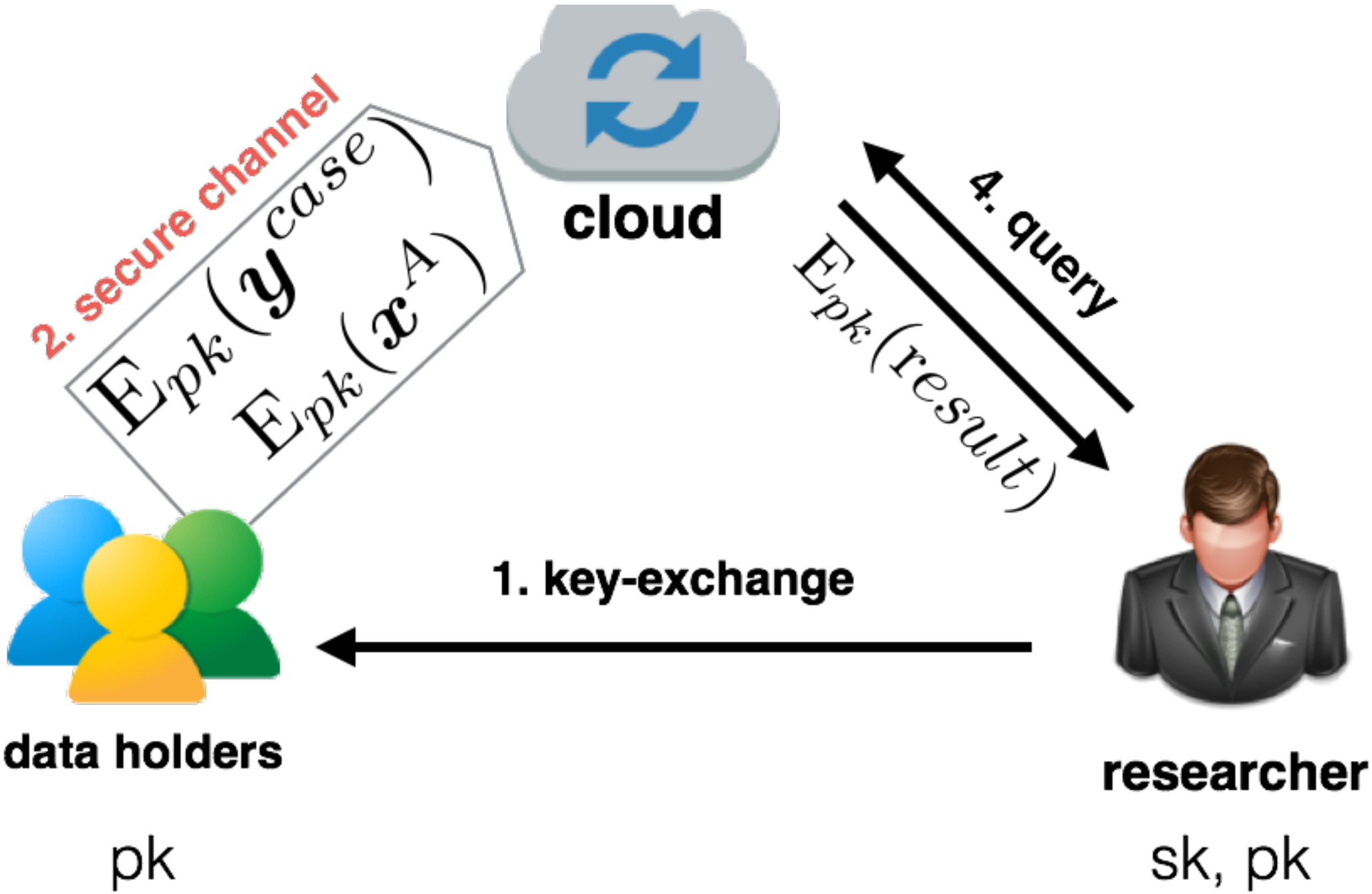
Efficient Outsourcing GWAS using FHE

Wenjie Lu*, Jun Sakuma*†
* Dept. of CS, University of Tsukuba, Japan
† JST CREST

Secure Outsourcing GWAS

3. Secure computation



Outline of our solution

[data holder]

$$[1,3,4] \rightarrow 1+3x+4x^2$$

Forward-backward packing
for scalar product computation

Encryption by
HELib[Halevi+14]

[Cloud]

Secure computation of r_2, a and d

	a	A	count
case	r_1	r_2	a
control	r_3	r_4	b
count	c	d	n

n — Public

Download encrypted E(a),E(d),E(r_2)

Decrypt them and construct two tables

Expectation	a	A
case	$e_1 = \frac{c \cdot a}{n}$	$e_2 = \frac{d \cdot a}{n}$
control	$e_3 = \frac{c \cdot b}{n}$	$e_4 = \frac{d \cdot b}{n}$

	a	A	count
case	r_1	r_2	a
control	r_3	r_4	b
count	c	d	n

n — Public

[Researcher]

$$\chi^2 = \sum_i \frac{(r_i - e_i)^2}{e_i}$$

Locally compute the
chi-square statistic

Halevi, Shai, and Victor Shoup. "Algorithms in helib." Advances in Cryptology–CRYPTO 2014. Springer Berlin Heidelberg, 2014. 554-571.

Notations

Allele of M subjects $\boldsymbol{x} = \{AA, Aa, aa\}^M$

$$\boldsymbol{y}_i^{case} = \begin{cases} 1 \text{ subject } i \text{ is in the case group} \\ 0 \text{ otherwise} \end{cases}$$

Vector containing 1 onlly: $\mathbf{1}$

Scalar Product of vector x and y: $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$

Our Encoding for SNPs

$$\bar{\boldsymbol{x}}_i = \begin{cases} 2 \text{ if } \boldsymbol{x}_i = AA \\ 1 \text{ if } \boldsymbol{x}_i = Aa \\ 0 \text{ otherwise} \end{cases}$$

Then we have

$$\langle \bar{\boldsymbol{x}}, \boldsymbol{y}^{\text{case}} \rangle = r_2$$

$$\langle \mathbf{1}, \boldsymbol{y}^{\text{case}} \rangle = a$$

$$\langle \bar{\boldsymbol{x}}, \mathbf{1} \rangle = d$$

	a	A	count
case	r_1	r_2	a
control	r_3	r_4	b
count	c	d	n

Public

How to compute scalar product securely and efficiently?

Fully Homomorphic Encryption (FHE)

- Brakerski– Gentry–Vaikuntanathan (BGV)[Brakerski +2012] scheme, implemented by HELib[Halevi+2014]

- The plaintext-space of the BGV scheme is a polynomial ring:

$$R_t := \mathbb{Z}_t[x]/(x^m + 1)$$

- Supports leveled homomorphic multiplication

$$\mathrm{Dec}(\mathrm{Enc}(a) \otimes \mathrm{Enc}(b)) = a \times b$$

Brakerski, Zvika, Craig Gentry, and Vinod Vaikuntanathan. "(Leveled) fully homomorphic encryption without boot strapping." Proceedings of the 3rd Innovations in Theoretical Computer Science Conference. ACM, 2012.

Packing Technique for Efficient Scalar Product

- The plaintext-space of the FHE scheme:

$$R_t := \mathbb{Z}_t[x]/(x^m + 1)$$

  is a polynomial ring.

- A vector of integers can be embedded into coefficients of the polynomial such as

$$[1, 3, 4] \rightarrow 1 + 3x + 4x^2$$

- The whole vector can be encrypted as one ciphertext such as

$$\mathrm{Enc}([1 + 3x + 4x^2])$$

Packing Technique for Efficient Scalar Product

[Yasuda et al. 2011]

Two integer vectors

$$\boldsymbol{v} := [v_0, v_1, \cdots, v_\ell] \quad \boldsymbol{u} := [u_0, u_1, \cdots, u_\ell] \quad (\ell < m)$$

Make two polynomials

$$\mathrm{ForwardPack}(\boldsymbol{v}) \rightarrow V(x) = v_0 + v_1 x + v_2 x^2 + \cdots + v_\ell x^\ell$$

$$\mathrm{BackwardPack}(\boldsymbol{u}) \rightarrow U(x) = u_\ell + u_{\ell-1} x + u_{\ell-2} x^2 + \cdots + u_0 x^\ell$$

The multiplication of V(x), U(x) yields a scalar product

$$V(x)U(x) = \cdots + \langle \boldsymbol{u}, \boldsymbol{v} \rangle x^\ell + \cdots$$

Scalar product can be securely and efficiently computed as

$$\mathrm{Enc}(V(x)) \otimes \mathrm{Enc}(U(x))$$

Yasuda, Masaya, et al. "Secure pattern matching using somewhat homomorphic encryption." Proceedings of the 2013 ACM workshop on Cloud computing security workshop. ACM, 2013.

Additive Property I

Prevention of information leakage by randomization

$$V(x)U(x) = \cdots + \langle \boldsymbol{u}, \boldsymbol{v} \rangle \, x^{\ell} + \cdots$$

information leak

Random Polynomial $R(x) = r_0 + \cdots + r_{\ell-1}x^{\ell-1} + r_{\ell+1}x^{\ell+1} + \cdots$

r_i is randomly chosen from $\mathbb{Z}_t$

prevent from information leak by randomization $V(x)U(x) + R(x)$

Outsourcing the computation of Contingency Table

	a	A	count
case	r_1	r_2	a
control	r_3	r_4	b
count	c	d	n

$$\langle \bar{\boldsymbol{x}}, \boldsymbol{y}^{\text{case}} \rangle = r_2$$

$$\langle \mathbf{1}, \boldsymbol{y}^{\text{case}} \rangle = a$$

$$\langle \bar{\boldsymbol{x}}, \mathbf{1} \rangle = d$$

data holders

Two ciphertexts!

$$\text{Enc}(\text{FPack}(\bar{\boldsymbol{x}}))$$

$$\text{Enc}(\text{BPack}(\boldsymbol{y}^{case}))$$

cloud

Three homomorphic multiplications!

cloud

$$\text{Enc}(\text{FPack}(\bar{\boldsymbol{x}})) \otimes \text{Enc}(\text{BPack}(\boldsymbol{y}^{case}))$$

$$\text{Enc}(\text{FPack}(\bar{\boldsymbol{x}})) \otimes \text{Enc}(\text{BPack}(\mathbf{1}))$$

$$\text{Enc}(\text{FPack}(\mathbf{1})) \otimes \text{Enc}(\text{BPack}(\boldsymbol{y}^{case}))$$

Scheme Parameters

- Parameters of the encryption scheme: plaintext-space parameter t = 20003; polynomial degree m = 4096; levels L = 3

- Security analysis of our scheme parameters[Gentry+2012]

$$m > \frac{(L(\log m + 23) - 8.5)(\kappa + 110)}{7.2}$$

κ-bit security is guaranteed.

In our settings, κ>= 128

Gentry, Craig, Shai Halevi, and Nigel P. Smart. "Homomorphic evaluation of the AES circ uit." Advances in Cryptology–CRYPTO 2012. Springer Berlin Heidelberg, 2012. 850-867.

Experiments

- Outsourcing the computation of the contingency table of *one* SNPs
- the number of subjects varies from 100 to 10,000
- CPU 2.3GHz, RAM 16GB
- FHE implementation: Helib [https://github.com/shaih/HElib]

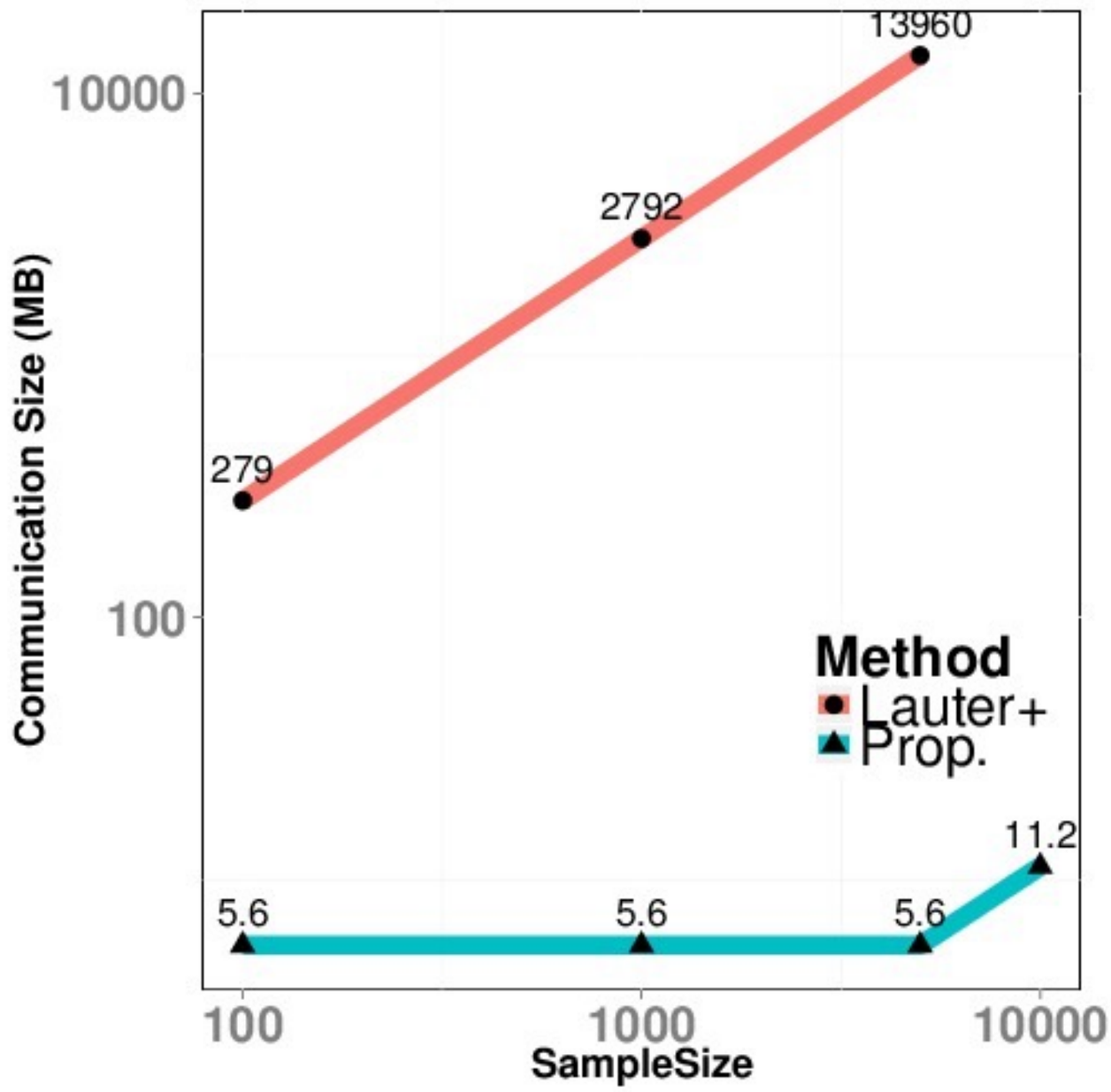
Experimental Results: Communication Size

Red Line: Lauter et al's encoding

Green Line: proposal encoding

X-axis: the number of subjects

Y-axis: communication size (MB)



Lauter, Kristin, Adriana López-Alt, and Michael Naehrig. "Private computation on encrypted genomic data." 14th Privacy Enhancing Technologies Symposium, Workshop on Genome Privacy 2014

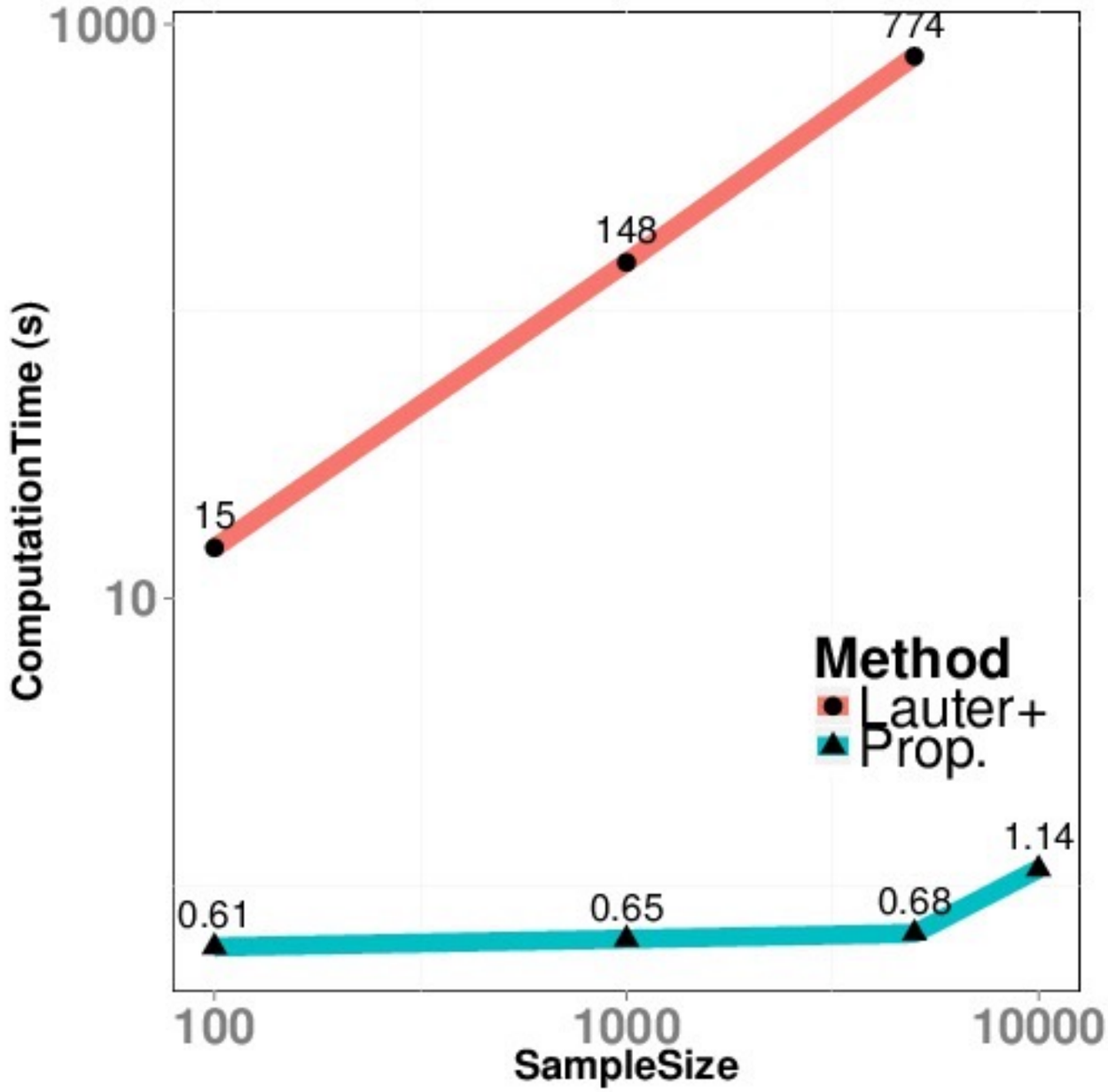
Experimental Results: Computation Time (cloud side)

Red Line: Lauter et al's encoding

Green Line: proposal encoding

X-axis: the number of subjects

Y-axis: computation time (sec)



Merits of the packing technique

- Communication Efficiency: Allele of several thousands of subjects can be packed into a single ciphertext
- Computation Efficiency: Scalar product of two vectors needs only a single homomorphic multiplication

Scalability of our method

$$\boldsymbol{v} := [v_0, v_1, \cdots, v_\ell] \quad \boldsymbol{u} := [u_0, u_1, \cdots, u_\ell]$$

When $\ell \geq m$, which means the number of subjects is too large

1. Use larger parameter m, (may not be computationally efficient)

2. Partition v, u into smaller pieces

$$\boldsymbol{v} \to [\boldsymbol{v}_1 || \boldsymbol{v}_2 || \cdots || \boldsymbol{v}_k] \quad \boldsymbol{u} \to [\boldsymbol{u}_1 || \boldsymbol{u}_2 || \cdots || \boldsymbol{u}_k]$$

$$\langle \boldsymbol{v}, \boldsymbol{u} \rangle = \sum_{i=1}^{k} \langle \boldsymbol{v}_i, \boldsymbol{u}_i \rangle$$

Thank you!

An Existent Encoding for SNPs

[Lauter et al. 2014]

Encoding for Genotype:

$$\boldsymbol{x}_i = \begin{cases} AA & \rightarrow [\mathrm{Enc}(1), \mathrm{Enc}(0), \mathrm{Enc}(0)] \\ Aa & \rightarrow [\mathrm{Enc}(0), \mathrm{Enc}(1), \mathrm{Enc}(0)] \\ aa & \rightarrow [\mathrm{Enc}(0), \mathrm{Enc}(0), \mathrm{Enc}(1)] \end{cases}$$

Encoding for Phenotype:

$$\boldsymbol{y}_i^{case} = \begin{cases} 1 & \rightarrow [\mathrm{Enc}(1), \mathrm{Enc}(0)] \\ 0 & \rightarrow [\mathrm{Enc}(0), \mathrm{Enc}(1)] \end{cases}$$

The number of ciphertext of M subjects is 5M for one SNP.

Additive Property II

Data collection from multiple data holders

The genotype and phenotype data is hold separately by Alice and Bob

$$\bar{\boldsymbol{x}} = [\bar{\boldsymbol{x}}_{\mathrm{A}} || \bar{\boldsymbol{x}}_{\mathrm{B}}] \qquad \boldsymbol{y}^{\mathrm{case}} = [\boldsymbol{y}_{\mathrm{A}}^{\mathrm{case}} || \boldsymbol{y}_{\mathrm{B}}^{\mathrm{case}}]$$