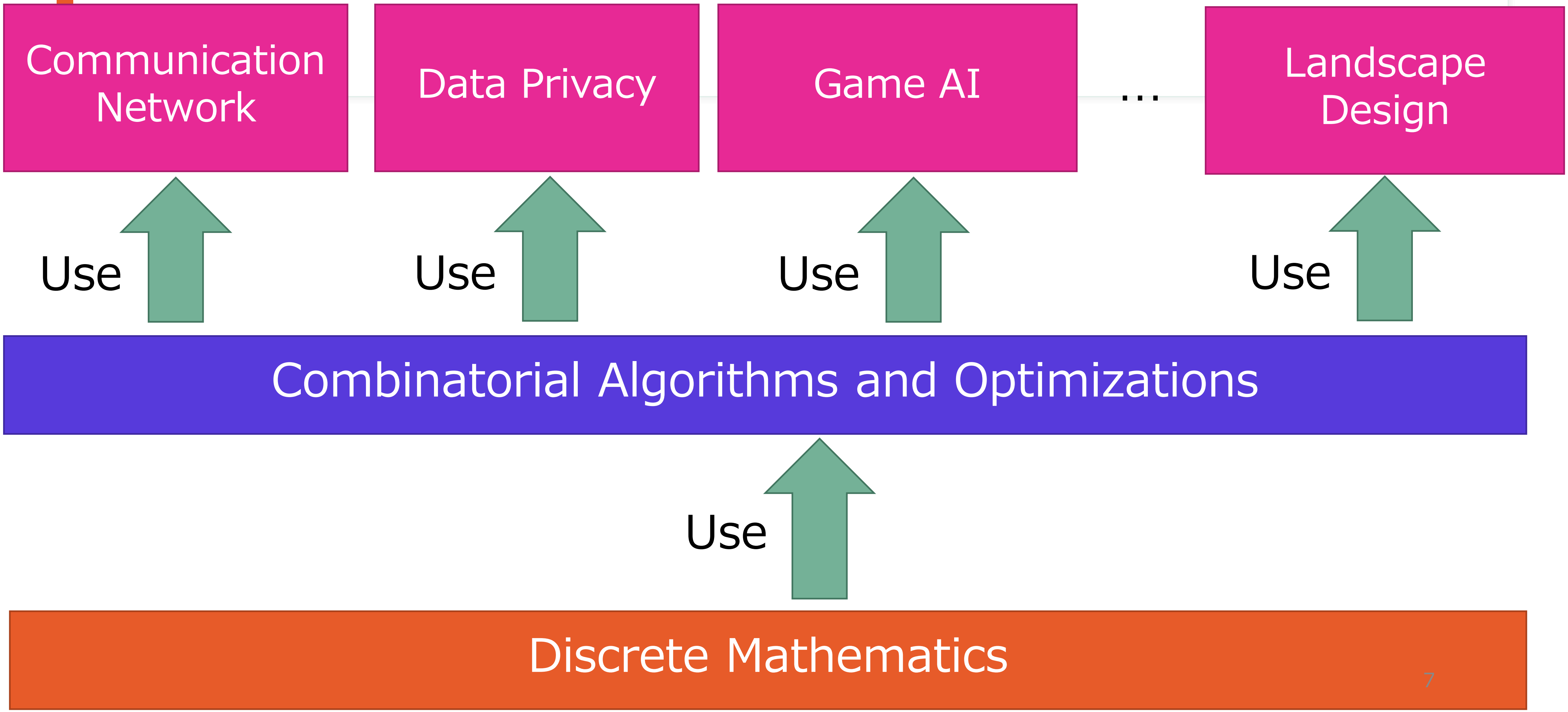
Variations of the subset sum and the maximum matching problem inspired by calculations in cryptography and wireless localization

Vorapong Suppakitpaisarn

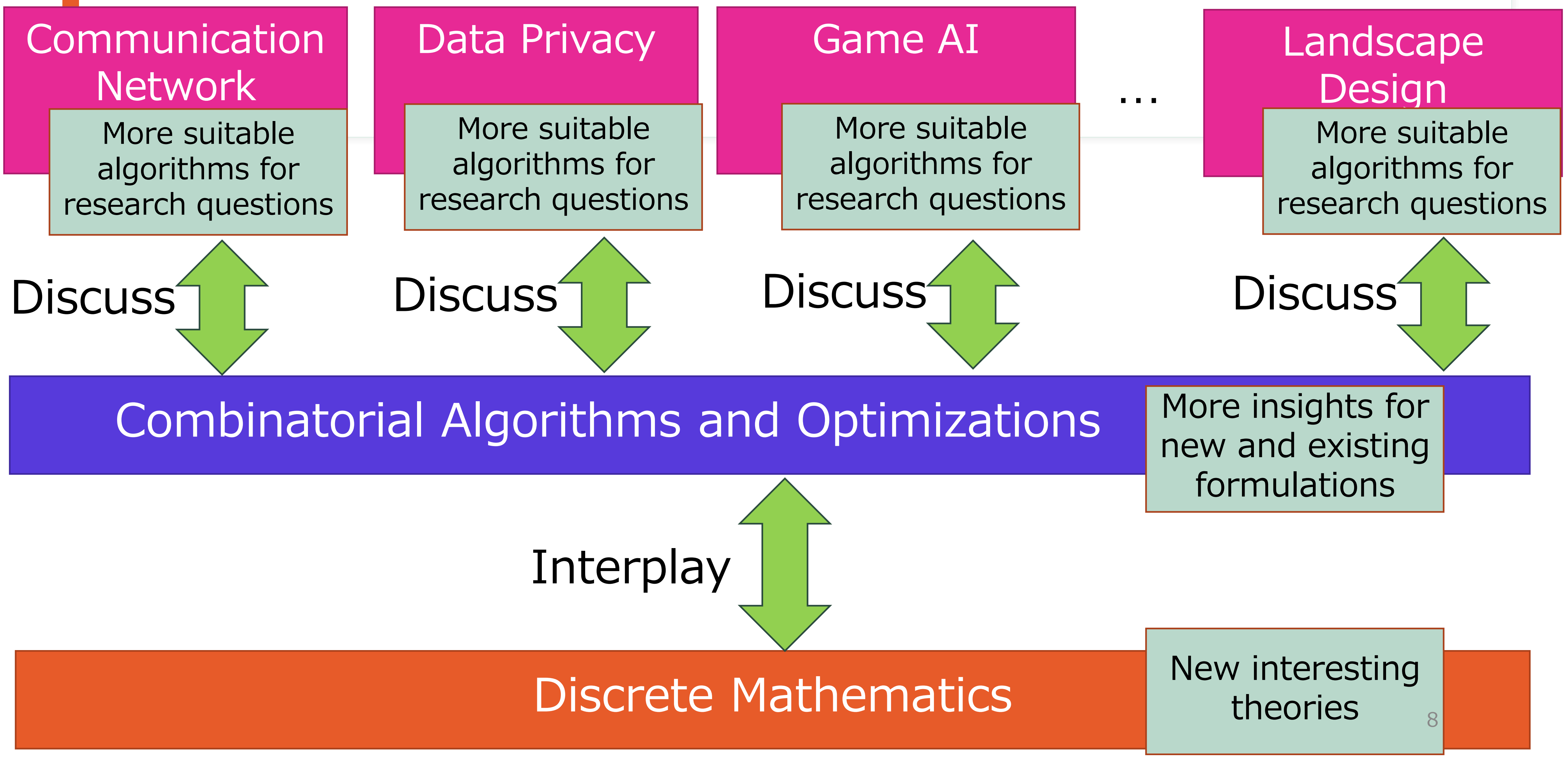
International Center for IST & Department of Computer Science

The University of Tokyo

Combinatorial Optimizations and Other CS Areas



Combinatorial Optimizations and Other CS Areas



Case 1

Parinya Chalermsook
Aalto U.

Hiroshi Imai
UTokyo

Daniel Krenn
U. Salzburg

Stefan Wagner
Uppsala U.

Hamming weight of 2-smooth numbers

Interplay

Special cases of minimum subset sum problem

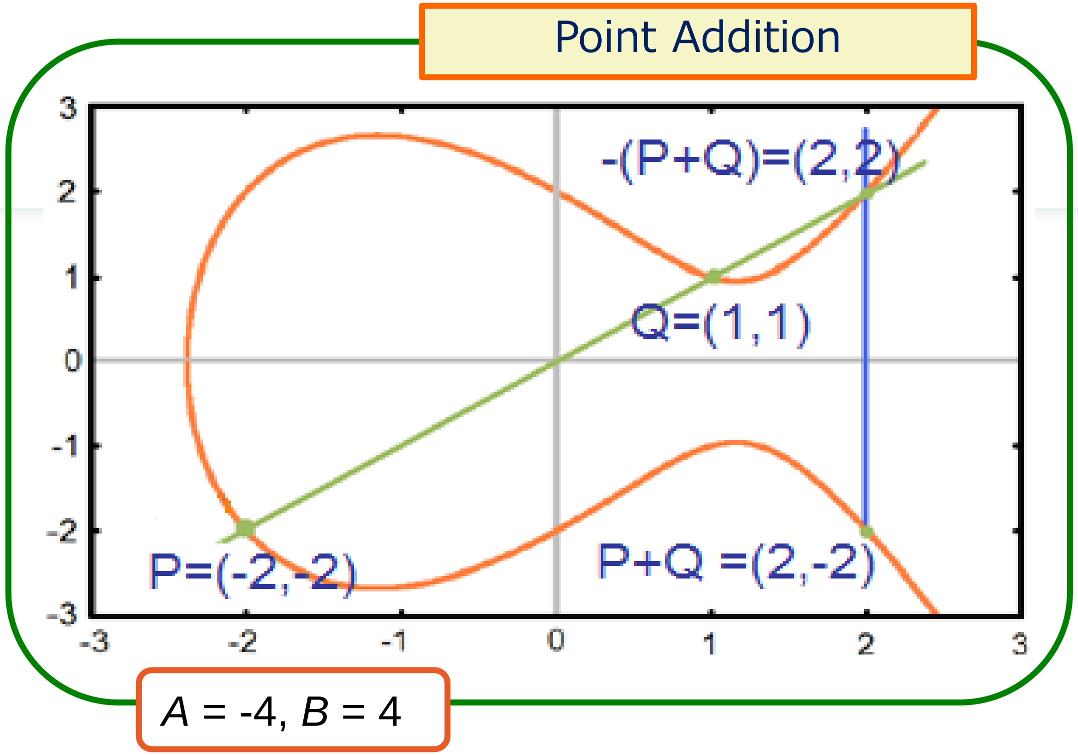
Discuss

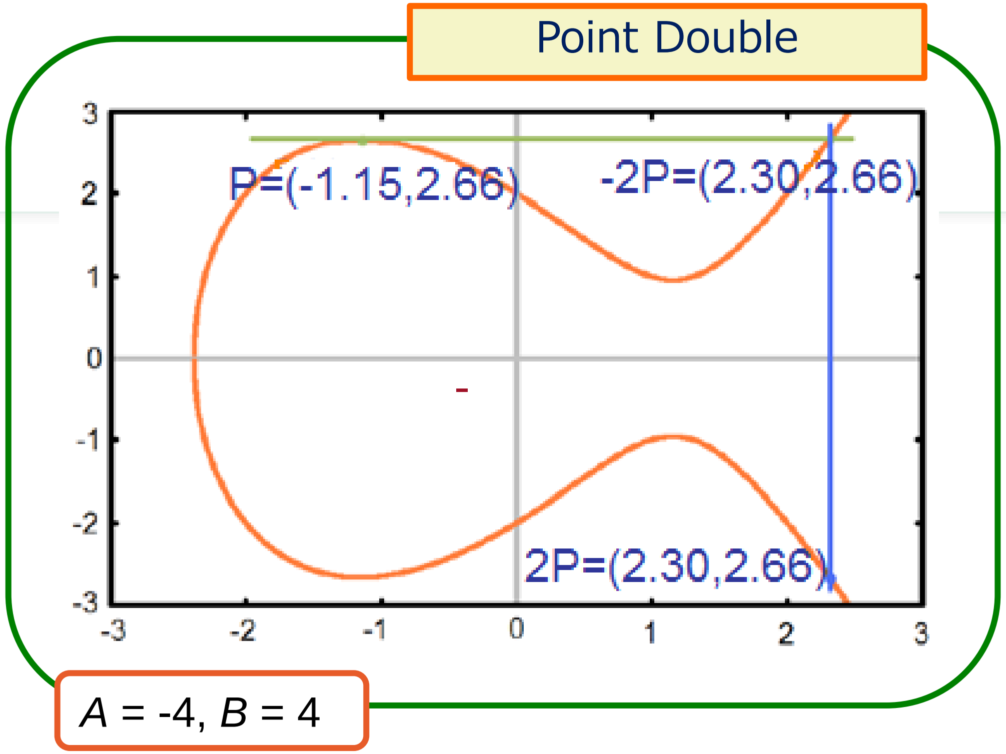
Implementation of cryptographic algorithm

Topics considered in several crypto conferences

Elliptic Curve

Weierstrass Equation: $y^2 = x^3 + Ax + B$





Scalar Multiplication: $aP = \underbrace{P + \cdots + P}_{a \text{ times}}$

Elliptic Curve Key Exchan… …l $a, b \approx 2^{256}$

Other persons
knows *P*, *aP*,
bP, but not *abP*

A L I C E

1. Generate ***P*** $\in E(\mathbb{Z}_p)$
2. Generate positive integer *a*
3. Receive Q = ***bP***
4. Compute aQ = ***abP***

P

aP

bP

Key

B O B

1. Receive ***P***
2. Receive S = ***aP***
3. Generate positive integer *b*
4. Compute bS = ***abP***

For other persons

Given *P*, *aP*, and *bP*,
Compute *abP*.

Diffie-Hellman Problem

No algorithm in $o(a^{1/2})$ ≈20,000 years

For Alice

Given *a*, *P*
Compute *aP*.

Scalar Multiplication

There is an algorithm with time $O(\log a)$.

Scalable algorithm for calculating aP

P

$P + P = 2P$

$2P + 2P = 4P$

$57P = 32P + 16P + 8P + P$

$O(\log a)$ additions

$8P = 4P + 4P$

$16P = 8P + 8P$

$32P = 16P + 16P$

Question: How to find a set of points in memory which gives smallest number of additions?

Subset Sum Problem!!

Giving a set $\mathcal{S}$ and an integer r, find a set $S' \subseteq \mathcal{S}$ with smallest size $\sum_{s \in S'} s = r$.

NP-hard problem

But, we can solve the problem in this case.

$\mathcal{S} = \{1,2,4,8,\ldots,2^n\}$ and an integer $r < 2^n$, find a set $S' \subseteq \mathcal{S}$ with smallest size $\sum_{s\in S'} s = r$.

There is always one solution. A greedy algorithm should work!

Ex: $r = 57$, optimal solution $= \{32,16,8,1\}$

$\mathcal{S}_1 = \{1,2,4,8,\ldots,2^n\}, \mathcal{S}_{-1} = \{-1,-2,\ldots,-2^n\}$

$\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_{-1}$ and an integer $r < 2^n$, find a set $S' \subseteq \mathcal{S}$ with smallest size $\sum_{s\in S'} s = r$. Can be solved in polynomial time! [folklore, c.f. Reitwiesner 1960]

Ex: $r = 57$, possible solution: $\{32,16,8,1\}$, $\underline{\{64,-8,1\}}$

Optimal Solution

Performance Analysis

Given a set $\mathcal{S}$ and an integer $r < 2^n$, find a set $S' \subseteq \mathcal{S}$ with smallest size $\sum_{s \in S'} s = r$.

Let the optimal solution be $S^*(r)$.
We want to know more about the set size $|S^*(r)|$ to avoid security attacks.

Worst Case: $\mathrm{W}(n) = \max_{0 \le r < 2^n} |S^*(r)|$ Average Case: $\mathrm{A}(n) = \frac{1}{2^n} \sum_{0 \le r < 2^n} |S^*(r)|$

When $\mathcal{S} = \{1,2,\dots,2^n\}$, $\mathrm{W}(n) = n + o(n)$ and $\mathrm{A}(n) = 0.5n + o(n)$

e.g. $S^*(15) = \{8,4,2,1\}$ [folklore]

When $\mathcal{S} = \{1,2,\dots,2^n\} \cup \{-1,-2,\dots,-2^n\}$, $\mathrm{W}(n) = n/2$ and $\mathrm{A}(n) = n/3$

[Egecioglu and Koc, 1994]

Other $\mathcal{S}$ $\mathcal{S}_p = \{p, 2p, \dots, 2^n p\}$,

When $\mathcal{S} = \mathcal{S}_1$, $\mathrm{W}(n) = n$ and $\mathrm{A}(n) = n/2$

When $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_{-1}$, $\mathrm{W}(n) = n/2$ and $\mathrm{A}(n) = n/3$ [Egecioglu and Koc, 1994]

When $\mathcal{S} = \mathcal{S}_3 \cup \mathcal{S}_1 \cup \mathcal{S}_{-1} \cup \mathcal{S}_{-3}$, $\mathrm{W}(n) = n/3$ and $\mathrm{A}(n) = n/4$

When $\mathcal{S} = \bigcup_{-7 \le h \le 7 : h \text{ is odd}} \mathcal{S}_h$, $\mathrm{W}(n) = n/4$ and $\mathrm{A}(n) = n/5$

When $\mathcal{S} = \bigcup_{-2^q+1 \le h \le 2^q-1 : h \text{ is odd}} \mathcal{S}_h$, $\mathrm{W}(n) = \frac{n}{q+1}$ and $\mathrm{A}(n) = \frac{n}{q+2}$

[Muir 2004]

Other collection $\mathcal{S}$

$$\mathcal{S}_p = \{p, 2p, \dots, 2^n p\}$$

We can calculate $\mathrm{W}(n)$ for $\mathcal{S} = \cup_{-2p+1 \le h \le 2p-1} \mathcal{S}_h$ for $2p-1$ that cannot be written in the form of $2^q - 1$. It is equal to $\mathrm{W}(n)$ for $\mathcal{S} = \cup_{-2^q+1 \le h \le 2^q-1} \mathcal{S}_h$ for largest $2^q - 1$ smaller than $2p-1$.

Thus, we complete the table!

We can also find $\mathrm{A}(n)$ and $\mathrm{W}(n)$ for many collection $\mathcal{S}$ cryptographer is using.

Set $\mathcal{S}$	$\mathbf{A}(n)$	$\mathbf{W}(n)$
$\mathcal{S}_1 \cup \mathcal{S}_{-1}$	$\frac{1}{3}n$ [Egecioglu and Koc, 1994]	$\frac{1}{2}n$ [Egecioglu and Koc, 1994]
$\mathcal{S}_3 \cup \mathcal{S}_1 \cup \mathcal{S}_{-1} \cup \mathcal{S}_{-3}$	$\frac{1}{4}n$ [Muir, 2004]	$\frac{1}{3}n$ [Muir, 2004]
$\mathcal{S}_5 \cup \cdots \cup \mathcal{S}_{-5}$	$\frac{2}{9}n$ [Moller, 2005]	$\frac{1}{3}n$ [Sup and Imai, 2014]
$\mathcal{S}_7 \cup \cdots \cup \mathcal{S}_{-7}$	$\frac{1}{5}n$ [Muir, 2004]	$\frac{1}{4}n$ [Muir, 2004]
$\mathcal{S}_9 \cup \cdots \cup \mathcal{S}_{-9}$	$\frac{4}{21}n$ [Moller, 2005]	$\frac{1}{4}n$ [Sup and Imai, 2014]
$\mathcal{S}_{11} \cup \cdots \cup \mathcal{S}_{-11}$	$\frac{4}{22}n$ [Moller, 2005]	$\frac{1}{4}n$ [Sup and Imai, 2014]

Double-Base Number System (DBNS)

$\mathcal{S} = \{2^i : 0 \leq i \leq n\}$ and an integer $r < 2^n$, find a set $S' \subseteq \mathcal{S}$ with smallest size $\sum_{s \in S'} s = r$. Polynomial-time solvable

$\mathcal{D} = \{2^i 3^j : 0 \leq i, j \leq n\}$ and an integer $r < 2^n$, find a set $S' \subseteq \mathcal{D}$ with smallest size $\sum_{s \in S'} s = r$. **Open Problem**

Two bases (2,3) → Double-base Number System

Ex: $r = 41$, possible solution: $\{36,4,1\}$, $\{32,9\}$

Optimal Solution

Greedy Algorithm for DBNS

[Dimitrov, Imbert, and Mishra 2008]

1: $r' \leftarrow r$

2: Let s be the largest number in $\mathcal{D}$ that is smaller than r'.

3: $r' \leftarrow r' - s$, $S \leftarrow S \cup \{s\}$

4: If $r' \neq 0$: go to step 2

Let the greedy solution be $S^g(r)$.

Worst Case: $\mathrm{W}^g(n) = \max_{0 \le r < 2^n} |S^g(r)|$

Average Case: $\mathrm{A}^g(n) = \frac{1}{2^n} \sum_{0 \le r < 2^n} |S^g(r)|$

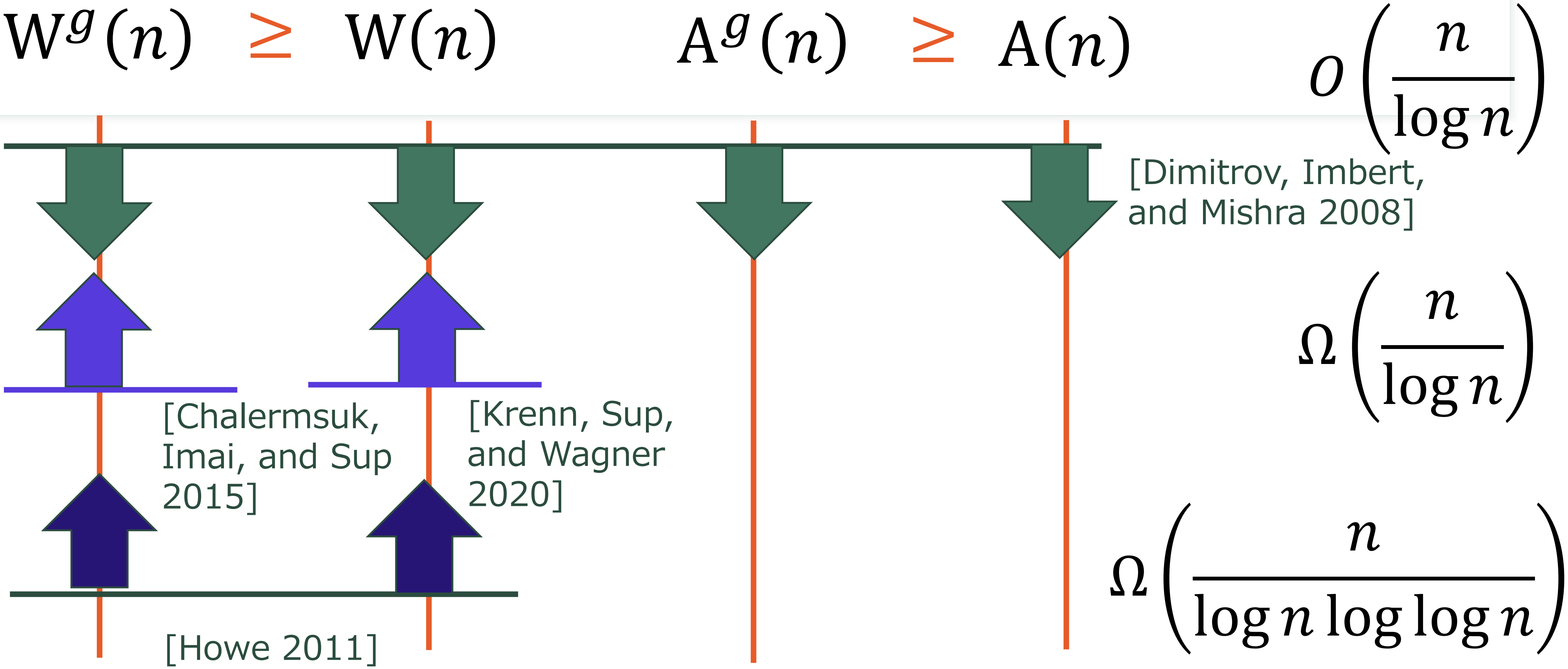
$\mathrm{W}^g(n) \in O(n/\log n)$!!!

We can reduce time complexity of cryptography using DBNS!!

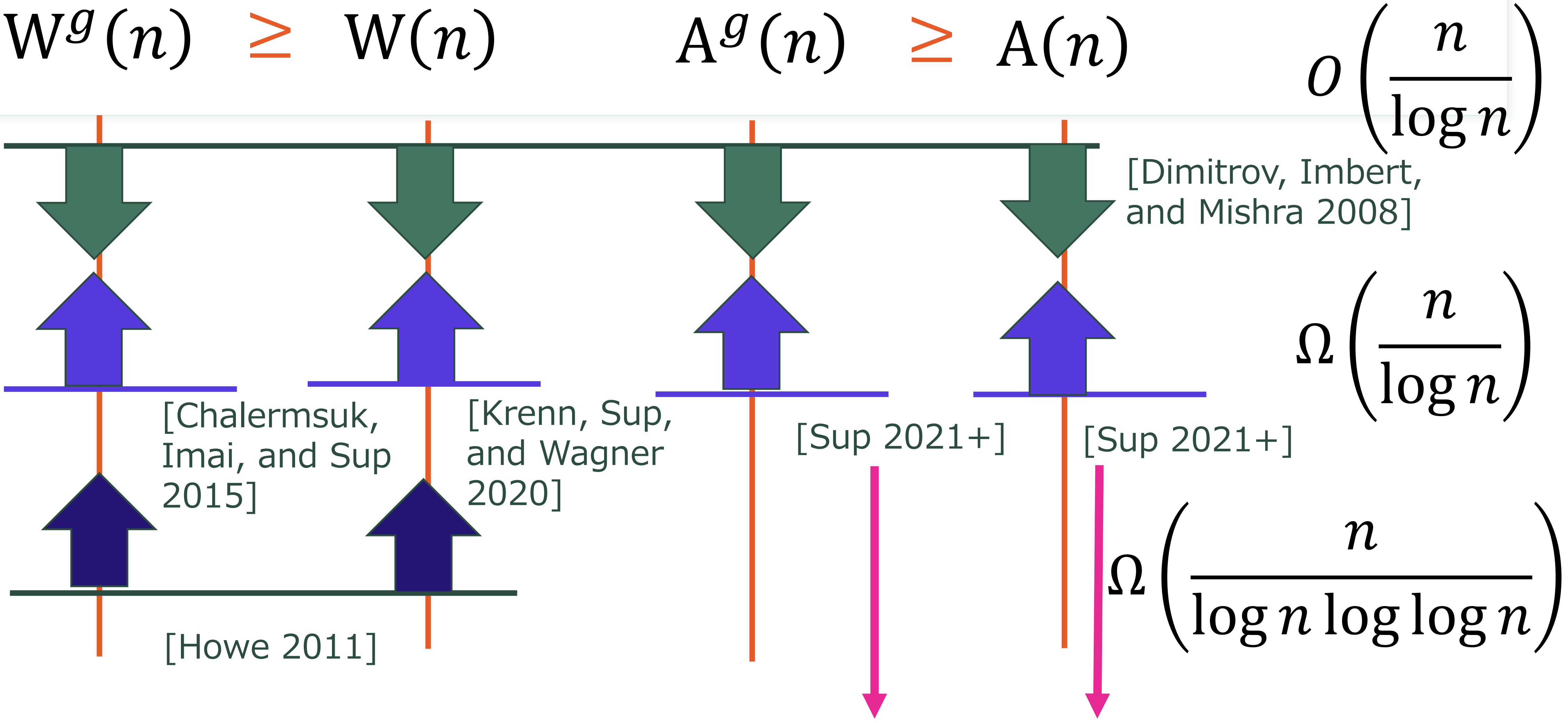
Ex: $r = 41$, possible solution: $\{36,4,1\}$, $\{32,9\}$

The greedy algorithm does not always give an optimal solution.

Asymptotic Values of DBNS



Asymptotic Values of DBNS



This presentation will focus here!