A Simple *Non-Interactive* Pairwise Key Establishment Scheme in Sensor Networks

Chia-Mu Yu[1,2], Chun-Shien Lu[1], and Sy-Yen Kuo[2]

[1]*Academia Sinica*

[2]*National Taiwan University*

Overview

- Sensor network
  - A lot of resource-constrained sensor nodes
  - Adversarial environment
- Goal of this paper
  - Establishing key between any pair of nodes
- Node compromise

Key Establishment in Sensor Network

- ***Numerous*** proposals
  - First work is [CCS'02]
  - Then, [S&P'03], [CCS'03], [INFOCOM'04], [INFOCOM'05], [INFOCOM'06], [MobiHoc'07], etc
  - 872 results by Google Scholar
- Why one more scheme?
  - ***Interactive*** becomes ***non-interactive***

Why ***interactive*** matters?

- Energy
- Communication dominates the energy consumption in sensor networks[MobiCom'01]
- reducing #comm. prolongs network lifetime
- Security
- DoS attack which sends bogus key-sharing message to distant node

What We Want?

- **Desired properties**
  1. Resilience to the Adversary's Intervention
     - Eavesdropping, node compromise, and PHY/MAC attacks

What We Want?

- **Desired properties**
  1. Resilience to the Adversary's Intervention
     - Eavesdropping, node compromise, and PHY/MAC attacks
  2. Directed and Guaranteed Key Establishment
     - Node compromise

What We Want?

- **Desired properties**
  1. Resilience to the Adversary's Intervention
     - Eavesdropping, node compromise, and PHY/MAC attacks
  2. Directed and Guaranteed Key Establishment
     - Mobile sink, node compromise
  3. Resilience to Network Configurations
     - Mobility, heterogeneity, hardware, etc

What We Want?

- **Desired properties**
  4. Efficiency
     - Computation, communication, and storage

What We Want?

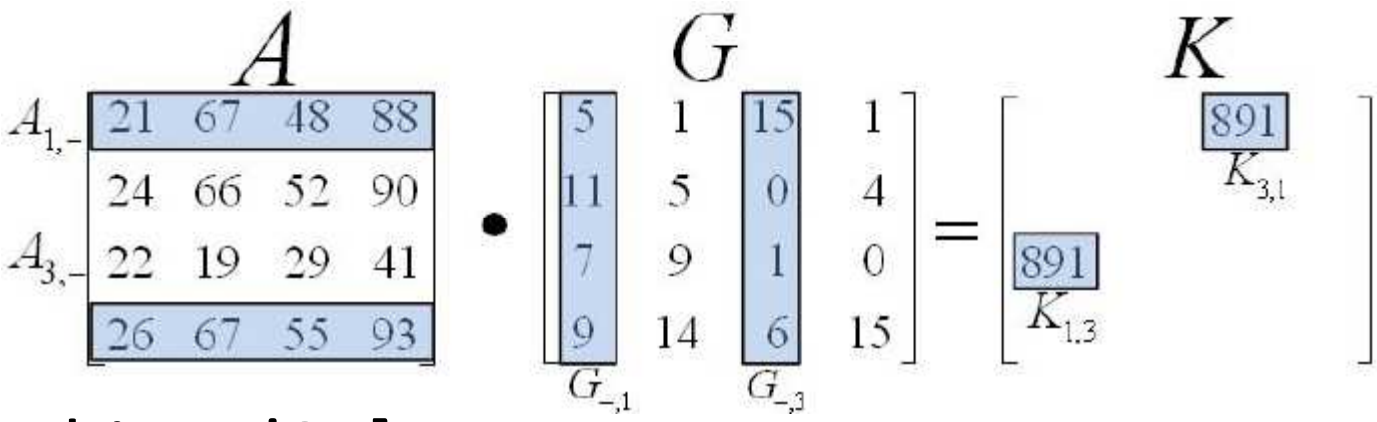
- **Desired properties**
  4. Efficiency
     - Computation, communication, and storage
  5. Resilience to Dynamic Node Deployment
     - Re-deploy some sensor nodes

What We Want?

- **Desired properties**
  4. Efficiency
     - Computation, communication, and storage
  5. Resilience to Dynamic Node Deployment
     - Re-deploy some sensor nodes
  6. Transparency
     - Power-saving protocols

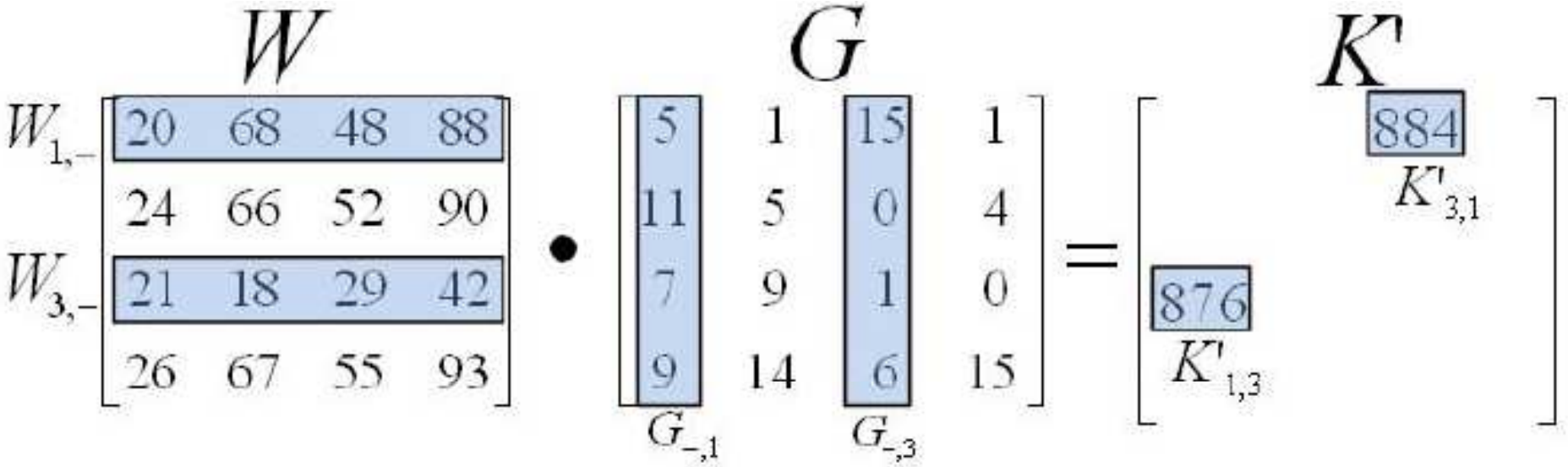
Related Work

- [EuroCrypt'84] $A = (D \cdot G)^T$ and $K = A \cdot G$



- [MobiHoc'07]
  - RPB scheme
  - Random perturbation

Idea



$$[20 \quad 68 \quad 48 \quad 88] = W_{1,-} = A_{1,-} + \phi_1 = [21 \quad 67 \quad 48 \quad 88] + [-1 \quad 1 \quad 0 \quad 0]$$

$$[21 \quad 18 \quad 29 \quad 42] = W_{3,-} = A_{3,-} + \phi_3 = [22 \quad 19 \quad 29 \quad 41] + [-1 \quad -1 \quad 0 \quad 1]$$

ϕ_1 and ϕ_3 are the random noise for $W_{1,-}$ and $W_{3,-}$, respectively

$K'_{1,3} \neq K'_{3,1}$ but $X_{1,3} = (11011)_2 = f_{10,5}(876)$

$X_{3,1} = (11011)_2 = f_{10,5}(884)$

If represented by 10 bits, their first 5 bits will be the same

Idea

- Perturbed result should be under control

$$W = \begin{matrix} W_{1,-} \\ \\ W_{3,-} \\ \\ \end{matrix} \begin{bmatrix} 20 & 68 & 48 & 88 \\ 24 & 66 & 52 & 90 \\ 21 & 18 & 29 & 42 \\ 26 & 67 & 55 & 93 \end{bmatrix}, \quad G = \begin{bmatrix} 5 & 1 & 15 & 1 \\ 11 & 5 & 0 & 4 \\ 7 & 9 & 1 & 0 \\ 9 & 14 & 6 & 15 \end{bmatrix}, \quad K' = \begin{bmatrix} & & 884 & \\ & & K'_{3,1} & \\ 876 & & & \\ K'_{1,3} & & & \end{bmatrix}$$

$G_{-,1}$ $G_{-,3}$

$$[20 \quad 68 \quad 48 \quad 88] = W_{1,-} = A_{1,-} + \phi_1 = [21 \quad 67 \quad 48 \quad 88] + [-1 \quad 1 \quad 0 \quad 0]$$

$$[21 \quad 18 \quad 29 \quad 42] = W_{3,-} = A_{3,-} + \phi_3 = [22 \quad 19 \quad 29 \quad 41] + [-1 \quad -1 \quad 0 \quad 1]$$

ϕ_1 and ϕ_3 are the random noise for $W_{1,-}$ and $W_{3,-}$, respectively

$K'_{1,3} \neq K'_{3,1}$ but $X_{1,3} = (11011)_2 = f_{10,5}(876)$

$X_{3,1} = (11011)_2 = f_{10,5}(884)$

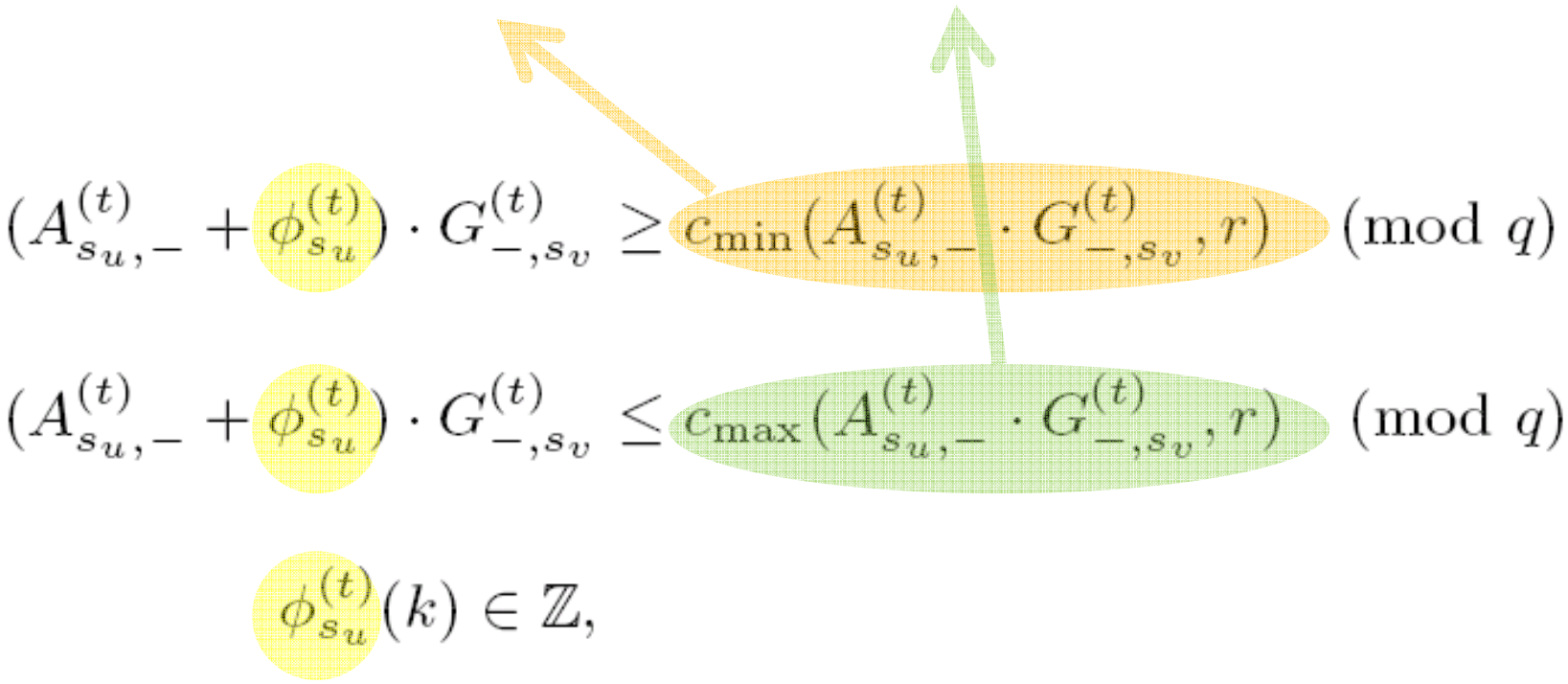
Perturbed result should be within the range

$[(11011\mathit{00000})_2, (11011\mathit{11111})_2]$

Constrained Random Perturbation

Perturbed result should be within the range

$[(11011\mathit{00000})_2, (11011\mathit{11111})_2]$



Added random perturbation

Constrained Random Perturbation

- Find constrained random perturbations
- LP Relaxation

$$(A^{(t)}_{s_u,-} + \phi^{(t)}_{s_u}) \cdot G^{(t)}_{-,s_v} \geq c_{\min}(A^{(t)}_{s_u,-} \cdot G^{(t)}_{-,s_v}, r) \pmod q$$

$$(A^{(t)}_{s_u,-} + \phi^{(t)}_{s_u}) \cdot G^{(t)}_{-,s_v} \leq c_{\max}(A^{(t)}_{s_u,-} \cdot G^{(t)}_{-,s_v}, r) \pmod q$$

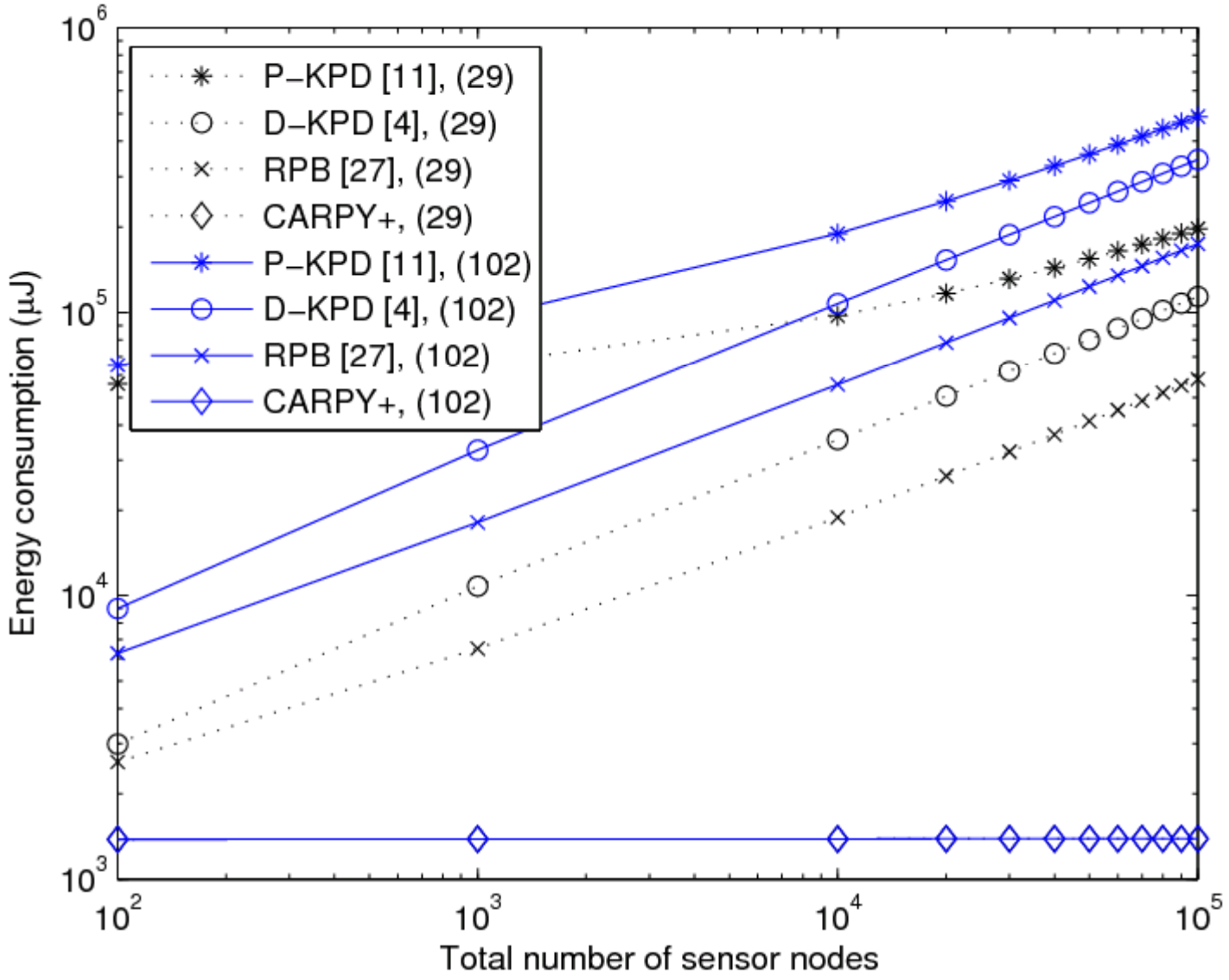
$$\phi^{(t)}_{s_u}(k) \in \mathbb{Z},$$

- Data type relaxation
  - Elements of G are float-pointed
  - randomly choose a perturbation vector, and test it

Advantage

- *All the advantages come from "non-interactive"*
  - Directed and Guaranteed Key Establishment
  - Resilience to Network Configurations
  - Efficiency
  - Resilience to Dynamic Node Deployment
  - Transparency
  - Resilience to the Adversary's Intervention
    - ***Security to node compromise***
    - ***Intuitively, compromise one more node, introduce one more uncertain variable***

Energy Consumption



Prototype

- IAR Embedded Workbench, instead of TinyOS
  - Native C, instead of nesC
- TelosB (MSP430F1611, CC2420)
- 690 bytes for CODE
- 8906 bytes for DATA
- ~0.15s for key generation
- ***Our program not optimized***
  - ***still working on it***

Conclusion

- A non-interactive key establishment scheme is proposed
- future work is reduce the storage cost

Thank you for your attention

For further information:
Chia-Mu Yu r91045@csie.ntu.edu.tw