Evaluation and Improvement of Generic-Emulating DPA Attacks

Weijia Wang, Yu Yu, Junrong Liu, Zheng Guo,
Francois-Xavier Standaert, Dawu Gu, Sen Xu, and Rong Fu

Outline

1. Background: generic-emulating DPA
2. Two new generic-emulating distinguishers
3. Improvement using cross-validation
4. Experimental results

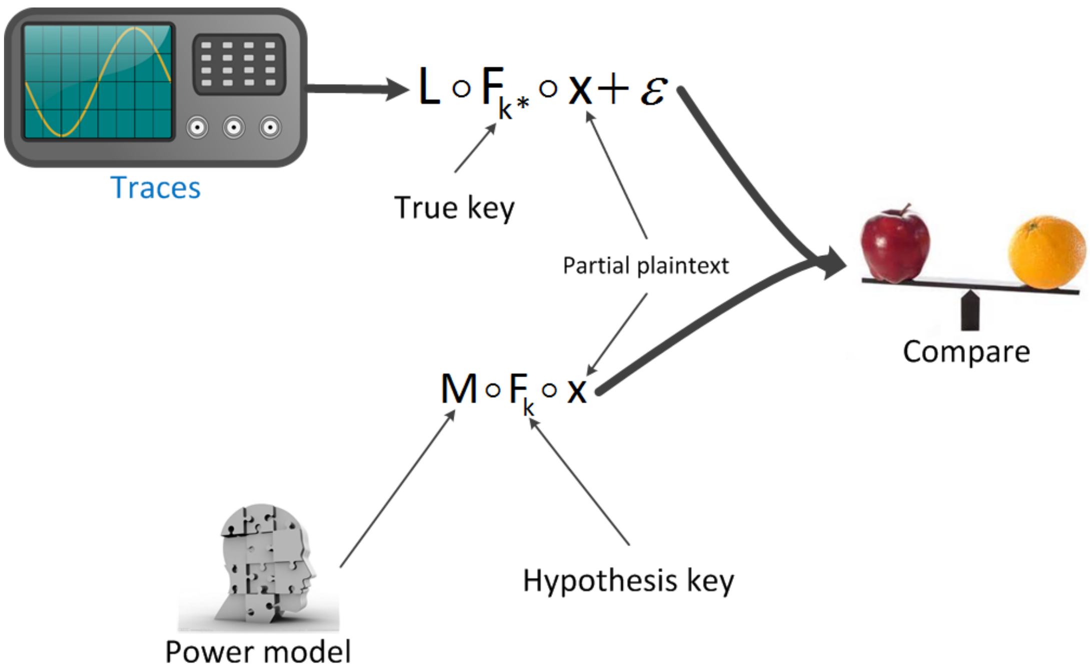
Outline

1. **Background: generic-emulating DPA**
2. Two new generic-emulating distinguishers
3. Improvement using cross-validation
4. Experimental Results

- **Schindler W, Lemke K, Paar C. A stochastic model for differential side channel cryptanalysis. CHES 2005.**
- **Doget J, Prouff E, Rivain M, et al. Univariate side channel attacks and leakage modeling. Journal of Cryptographic Engineering, 2011.**
- **Whitnall C, Oswald E, Standaert F X. The Myth of Generic DPA… and the Magic of Learning. CT-RSA 2014.**

1.1 Differential Power Analysis (DPA)

- Problems of DPA:
  - Choice of power model depends on the experiences of attacker
  - *The impact of power variability is becoming more and more significant, which makes common power models much less respected in practice.*



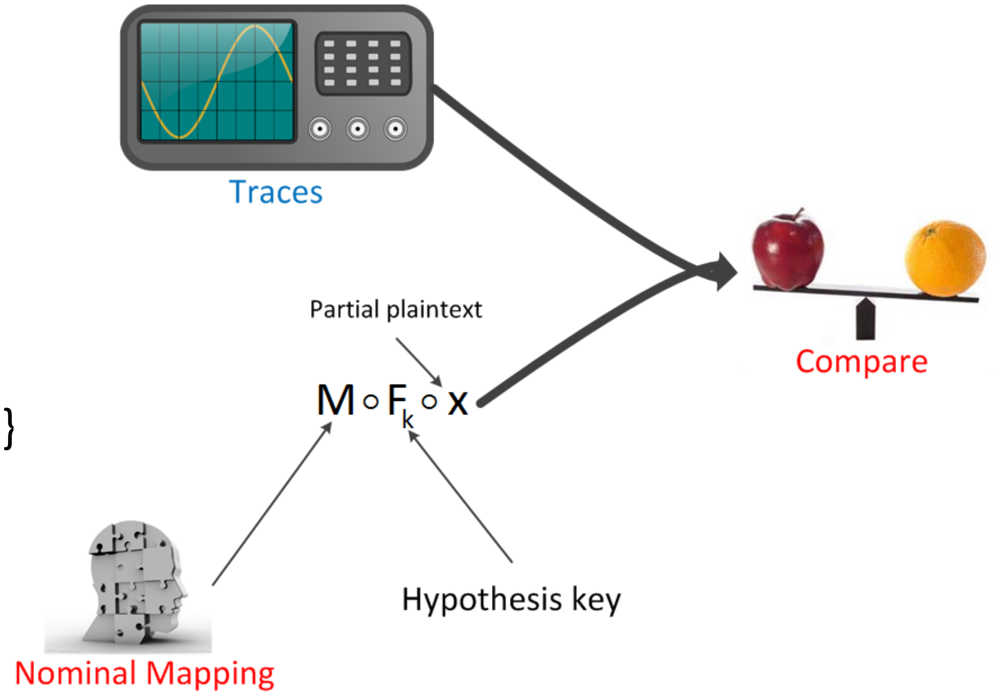
- Solution:
  - Generic DPA (e.g. MIA)

1.2 Generic DPA

- Generic DPA use the nominal mapping as power model.
  - We call the function $M(\cdot)$ as nominal mapping if we have:

$$\{z \mid M(z) = M(z')\} \approx \{z \mid L(z) = L(z')\}$$

- Limitation of generic DPA:
  - It doesn't work when the target function $F_k(x)$ is injective (AES sbox)



1.3 The Power Model using Algebra Normal Form

- Fact: any real valued leakage function can be represented in algebra normal form (ANF).
- For Example:

Let $z=(z_1,z_2,z_3)$ in $GF(2)^3$

For any leakage function $L(\cdot)$, we have:

$$L(z)=\alpha_0+\boxed{\alpha_1 z_1+\alpha_2 z_2+\alpha_3 z_3}+\boxed{\alpha_4 z_1 z_2+\alpha_5 z_1 z_3+\alpha_6 z_2 z_3}+\boxed{\alpha_7 z_1 z_2 z_3},$$

where α_0 to α_7 are the real number coefficients

terms of degree 1 terms of degree 2 terms of degree 3

1.3 The Power Model using Algebra Normal Form

- Fact: any real valued leakage function can be represented in algebra normal form (ANF).
- For Example:

Let $z = (z_1, z_2, z_3)$ in $GF(2)^3$

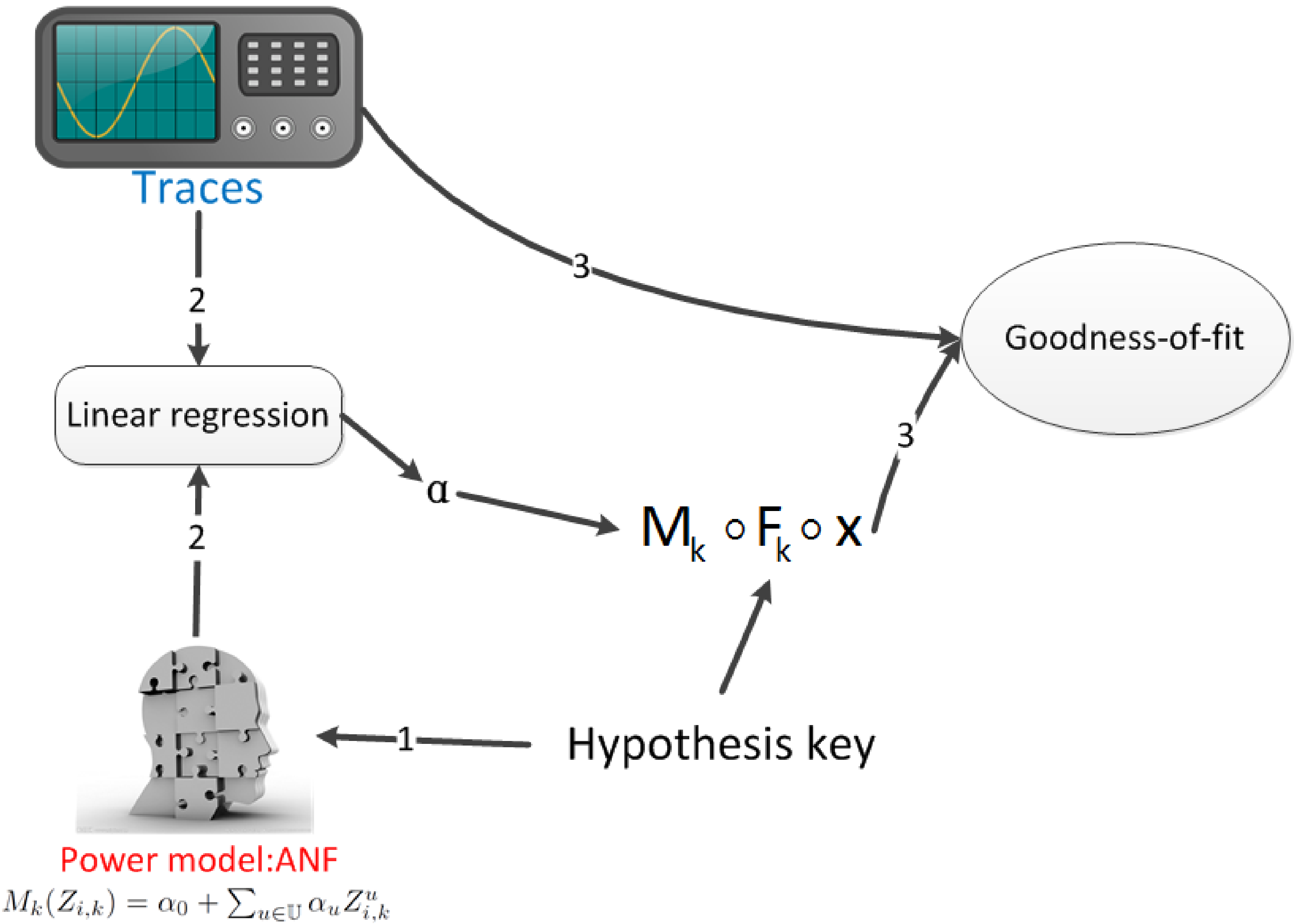
For any leakage function $L(\cdot)$, we have:

$$L(z) = \alpha_0 + \underbrace{\boxed{\alpha_1 z_1 + \alpha_2 z_2 + \alpha_3 z_3}}_{\text{terms of degree 1}} + \underbrace{\boxed{\alpha_4 z_1 z_2 + \alpha_5 z_1 z_3 + \alpha_6 z_2 z_3}}_{\text{terms of degree 2}} + \underbrace{\boxed{\alpha_7 z_1 z_2 z_3}}_{\text{terms of degree 3}},$$

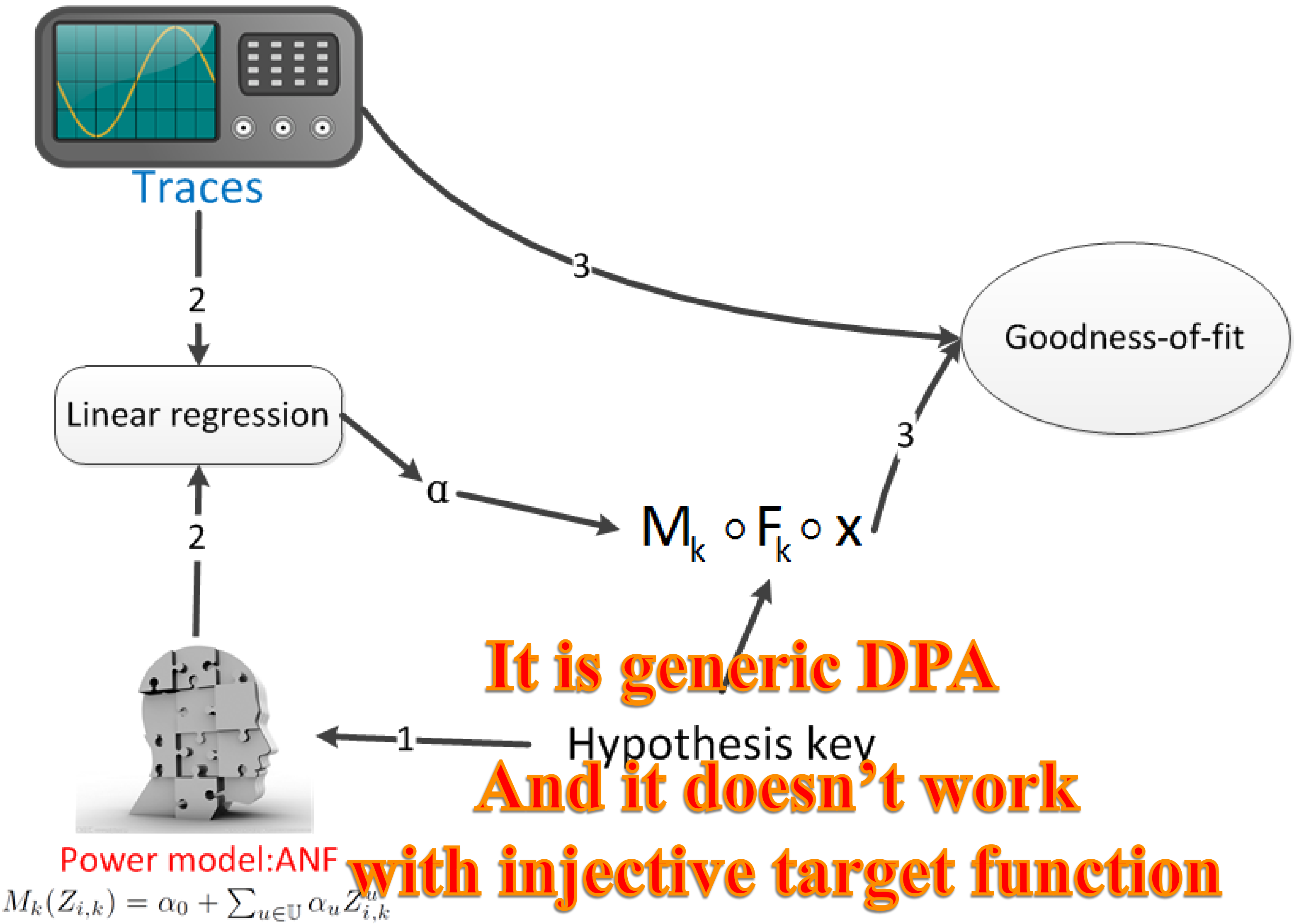
where α_0 to α_7 are the real number coefficients

- Therefore, we can construct the nominal mapping power model using ANF

1.4 Liner Regression(LR)-based DPA



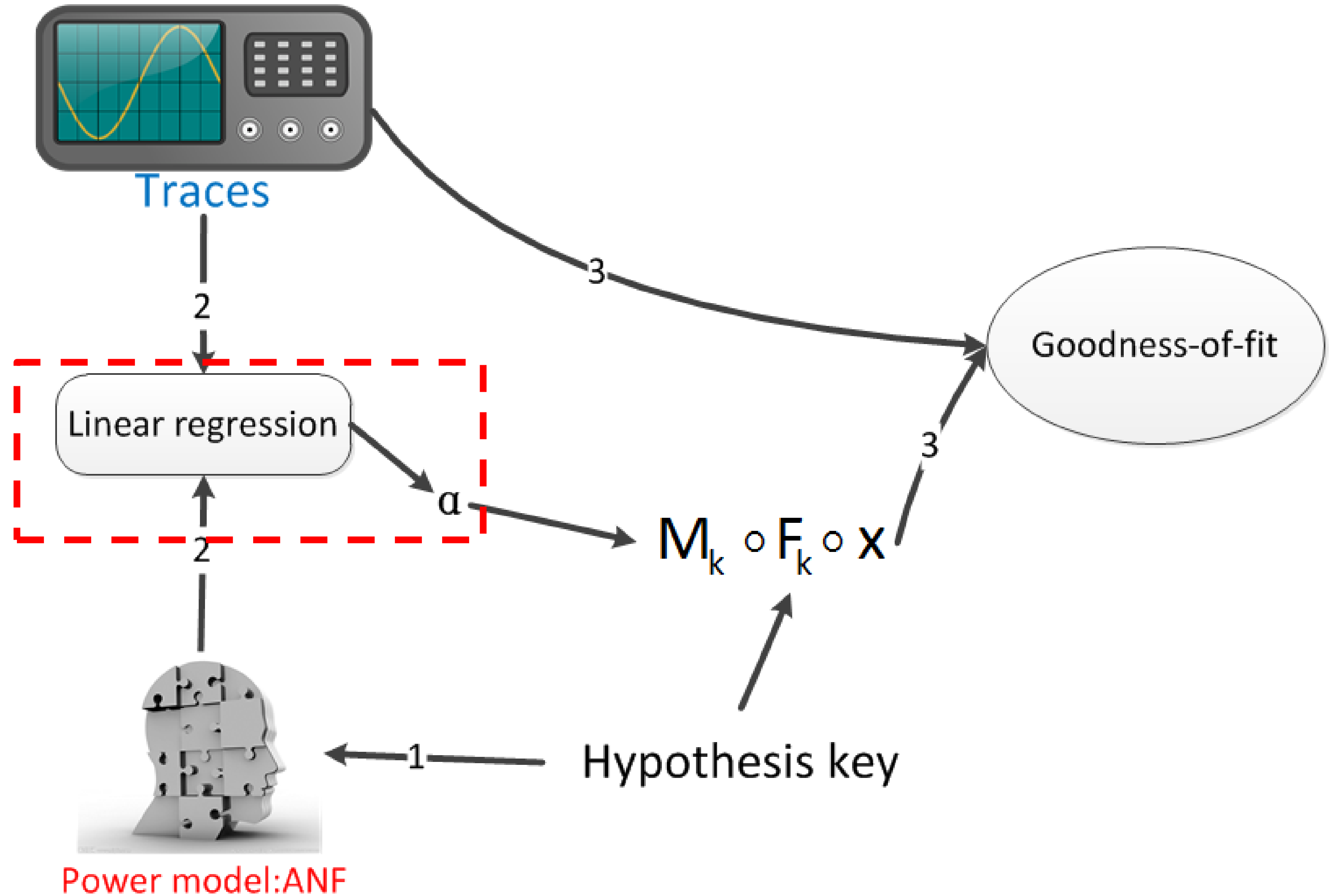
1.4 Liner Regression(LR)-based DPA



It is generic DPA

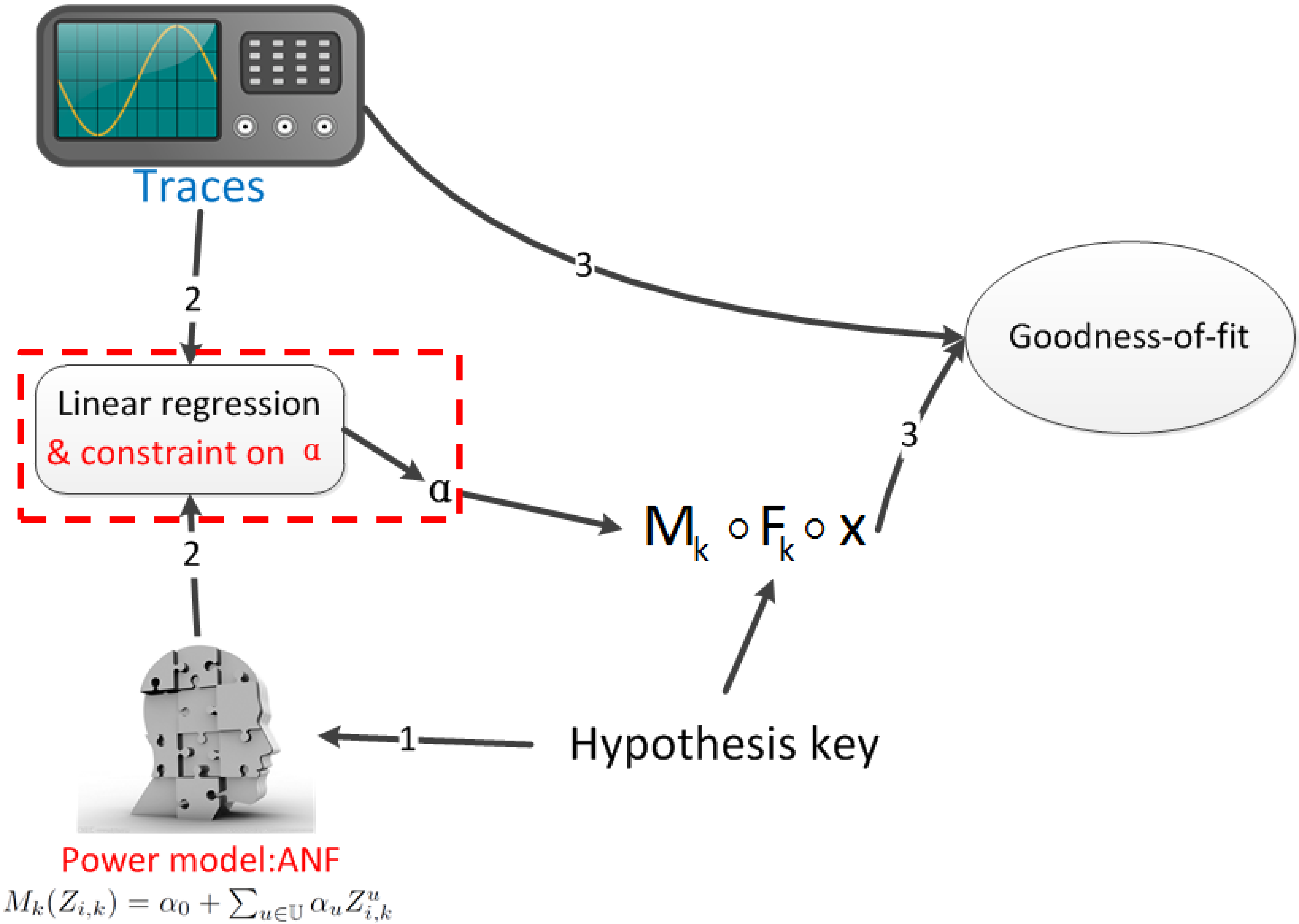
And it doesn't work with injective target function

1.5 Generic-emulating DPA

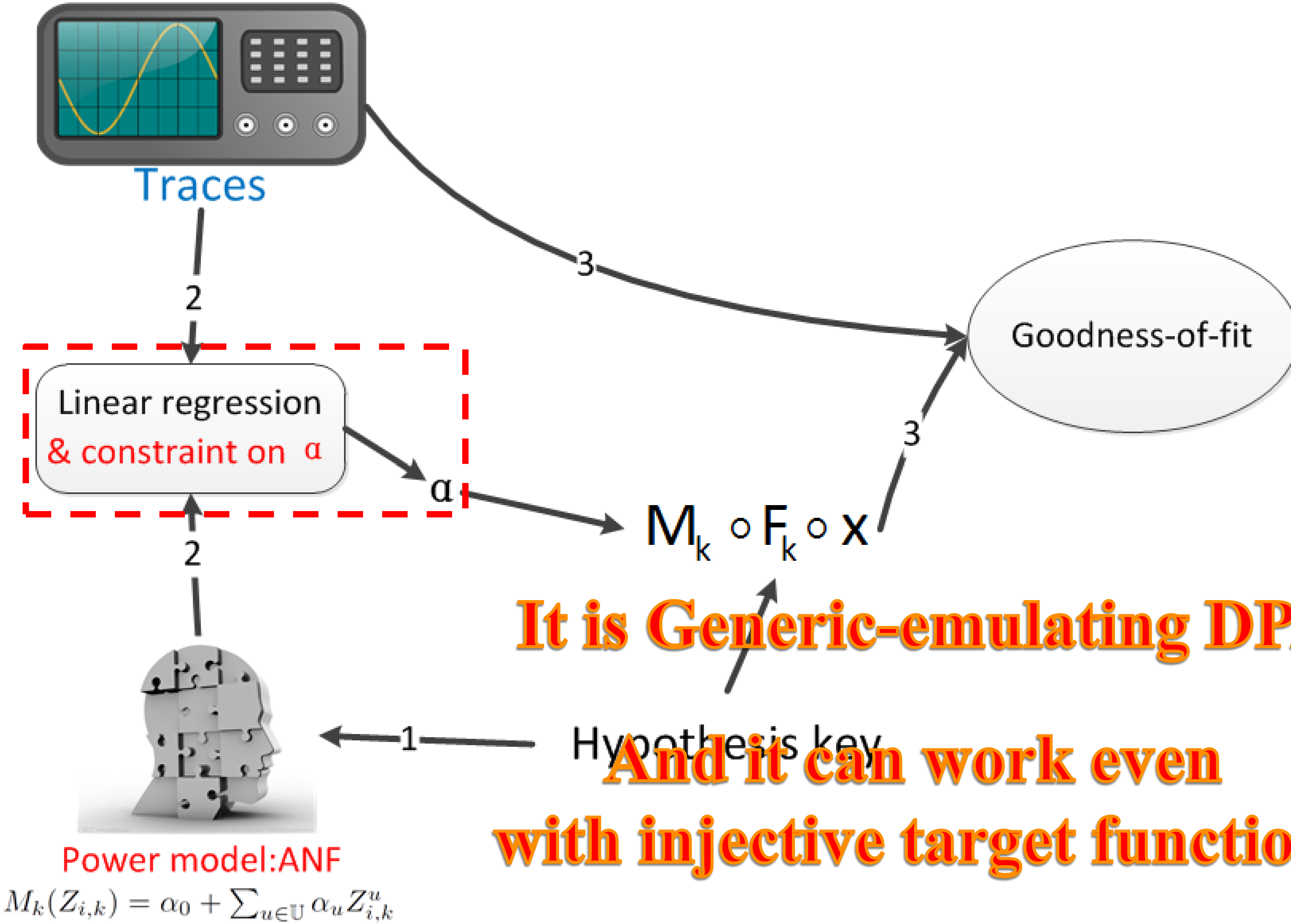


$$M_k(Z_{i,k}) = \alpha_0 + \sum_{u \in \mathbb{U}} \alpha_u Z_{i,k}^u$$

1.5 Generic-emulating DPA



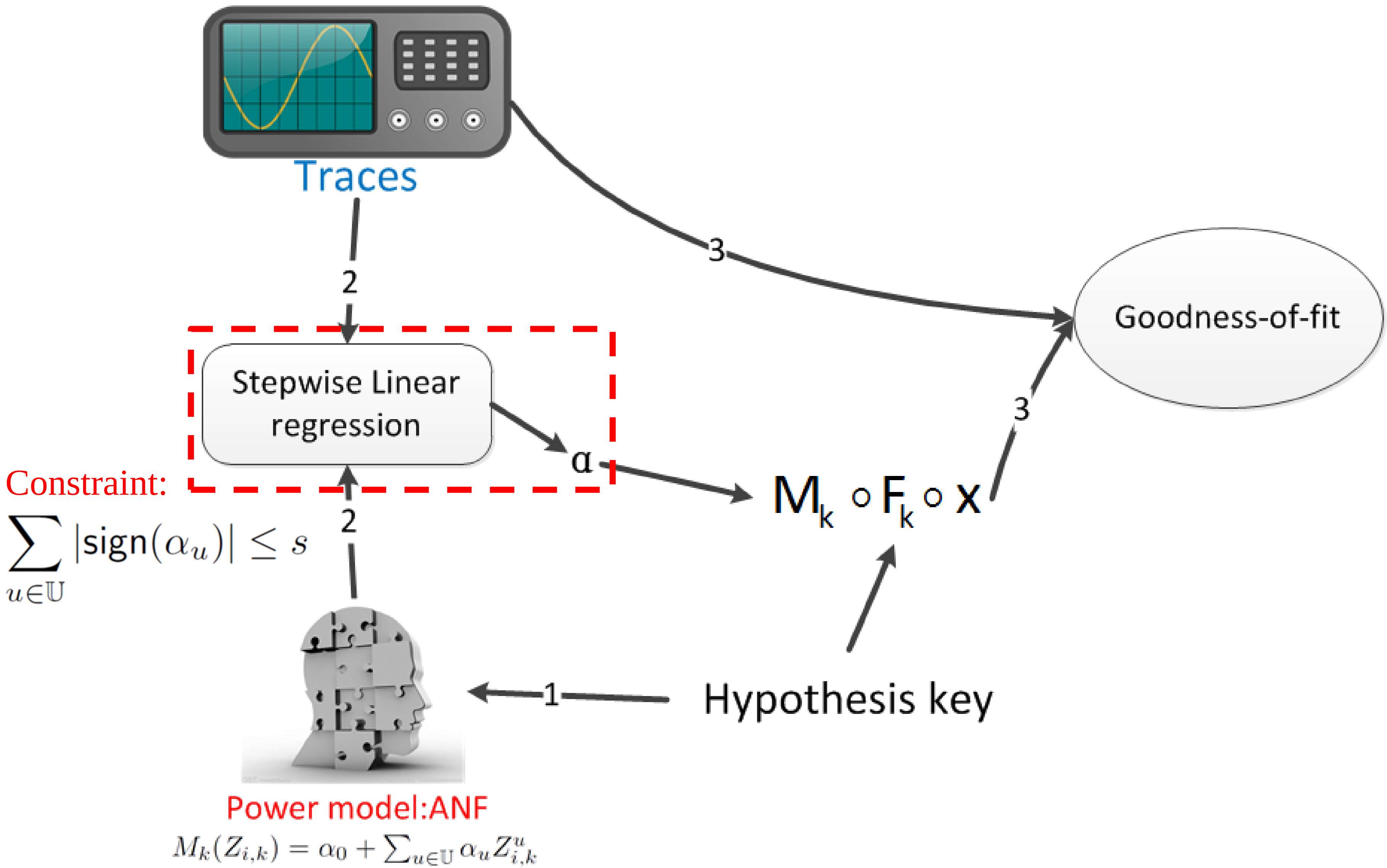
1.5 Generic-emulating DPA



It is Generic-emulating DPA

And it can work even with injective target function

1.6 Stepwise Linear Regression (SLR)-based DPA



1.6 SLR-based DPA

- The coefficients in the leakage function are sparse
- Formal description:

$$\hat{\boldsymbol{\alpha}}^{SLR} \stackrel{\mathsf{def}}{=} \underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^{N} (T_i - M_k(Z_{i,k}))^2$$

$$\text{subject to} \sum_{u \in \mathbb{U}} |\mathsf{sign}(\alpha_u)| \leq s$$

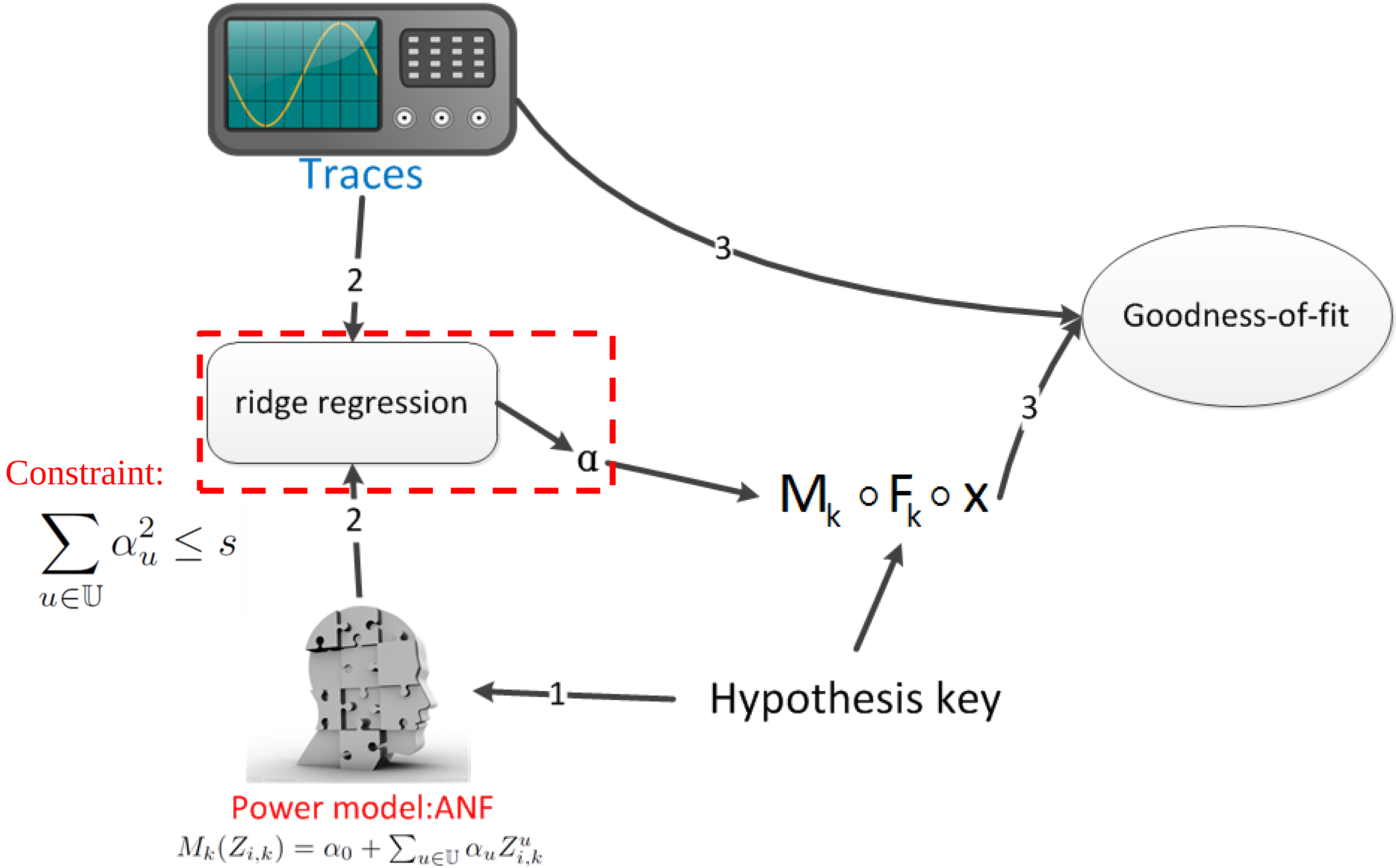
Outline

1. Background: generic-emulating DPA
2. **Two new generic-emulating distinguishers**
3. Improvement using cross-validation
4. Experimental Results

Motivation

- Two drawbacks in SLR-based DPA
  - Unstable outcomes in the high-noise regime
    - the insignificant coefficients are discarded, which makes the unstable outcomes
  - Less-satisfactory performance especially on real smart cards

2.1 Ridge-based Distinguishers



2.1 Ridge-based Distinguishers

- Ridge-based distinguisher shrinks coefficients by explicitly imposing an overall constraint on their size:

$$\hat{\boldsymbol{\alpha}}^{ridge} \stackrel{\mathsf{def}}{=} \operatorname*{argmin}_{\alpha} \sum_{i=1}^{N} \Big(T_i - M_k(Z_{i,k}) \Big)^2$$

$$\text{subject to } \sum_{u \in \mathbb{U}} \alpha_u^2 \leq s$$

2.1 Ridge-based Distinguishers

- Ridge-based distinguisher shrinks coefficients by explicitly imposing an overall constraint on their size:

$$\hat{\boldsymbol{\alpha}}^{ridge} \stackrel{\text{def}}{=} \underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^{N} \Big(T_i - M_k(Z_{i,k}) \Big)^2$$

$$\text{subject to } \sum_{u \in \mathbb{U}} \alpha_u^2 \leq s$$

- An equivalent formulation:

$$\hat{\boldsymbol{\alpha}}^{ridge} = \underset{\alpha}{\operatorname{argmin}} \left(\sum_{i=1}^{N} (T_i - M_k(Z_{i,k}))^2 + \lambda \sum_{u \in U} \alpha_u^2 \right)$$

2.1 Ridge-based Distinguishers

- The optimal solution is given by:

$$\hat{\boldsymbol{\alpha}}^{ridge} = (\boldsymbol{U}_k^\top \boldsymbol{U}_k + \lambda \boldsymbol{I})^{-1} \boldsymbol{U}_k^\top T$$

where $\boldsymbol{U}_k = (Z_{i,k}^u)_{i \in \{1,2,\ldots,N\}, u \in \mathbb{F}_2^m \setminus \{0\}}$

$$\hat{\boldsymbol{\alpha}}^{LR} = (\boldsymbol{U}_k^\top \boldsymbol{U}_k + \quad)^{-1} \boldsymbol{U}_k^\top T$$

shrink

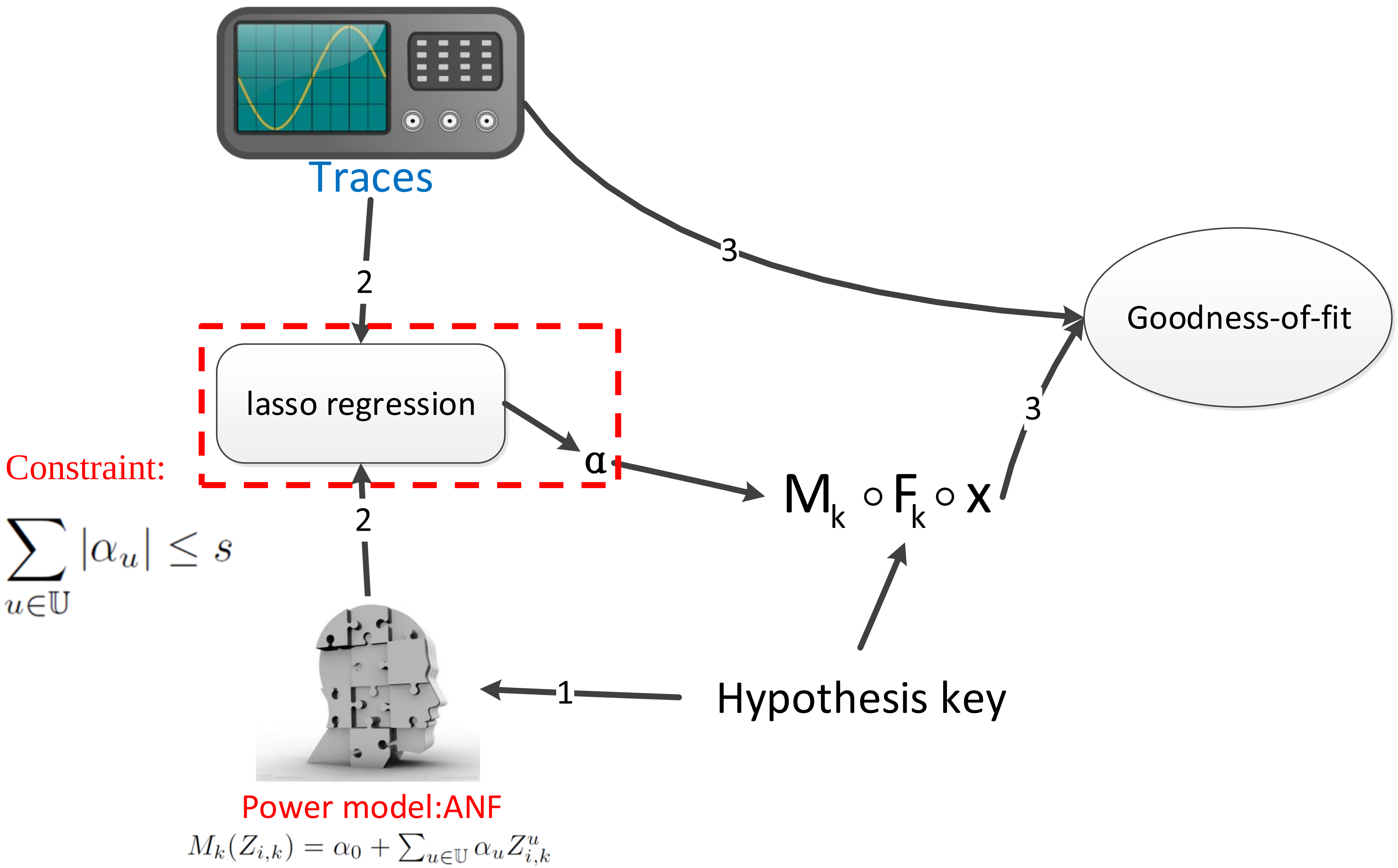
$$\hat{\boldsymbol{\alpha}}^{ridge} = (\boldsymbol{U}_k^\top \boldsymbol{U}_k + \lambda \boldsymbol{I})^{-1} \boldsymbol{U}_k^\top T$$

2.2 How The Coefficients Shrink in Ridge-based Distinguishers

Amount of shrinkage of terms' coefficients $\xleftrightarrow{\text{Proportional to}}$ Degrees of the terms

Consistent with leakage functions in practice

2.3 Lasso-based Distinguishers



2.3 Lasso-based Distinguishers

- The lasso-based distinguisher is similar to the ridge-based one excepted for a different constraint:

$$\hat{\boldsymbol{\alpha}}^{lasso} \overset{\text{def}}{=} \underset{\alpha}{\operatorname{argmin}} \sum_{i=1}^{N} \Big(T_i - M_k(Z_{i,k}) \Big)^2$$

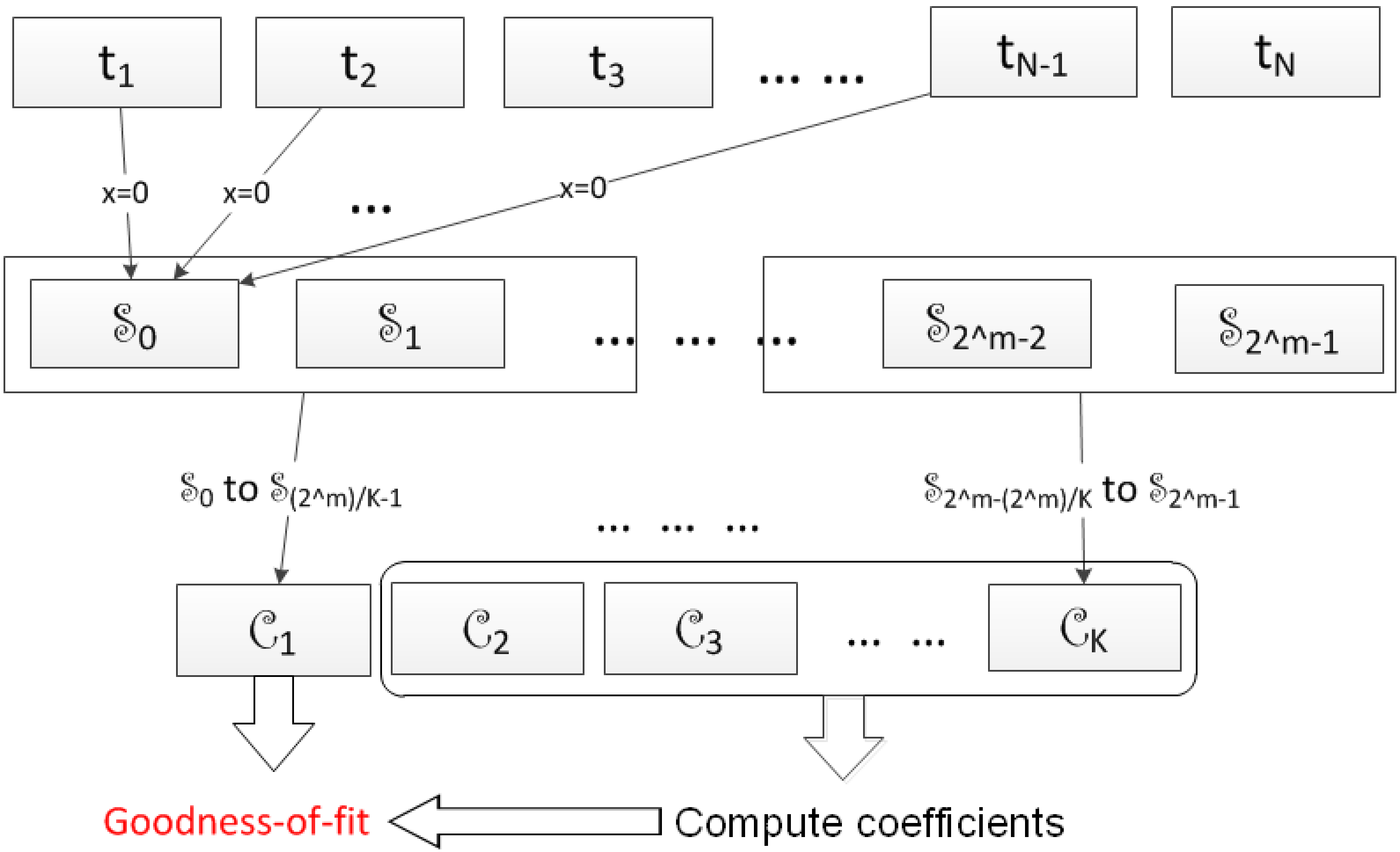
$$\text{subject to} \sum_{u \in \mathbb{U}} |\alpha_u| \leq s$$

- Finding the optimal solution for lasso-based distinguishers is essentially a quadratic programming problem

Outline

1. Background: generic-emulating DPA
2. Two new generic-emulating distinguishers
3. **Improvement using cross-validation**
4. Experimental results

3 Cross-validation



Outline

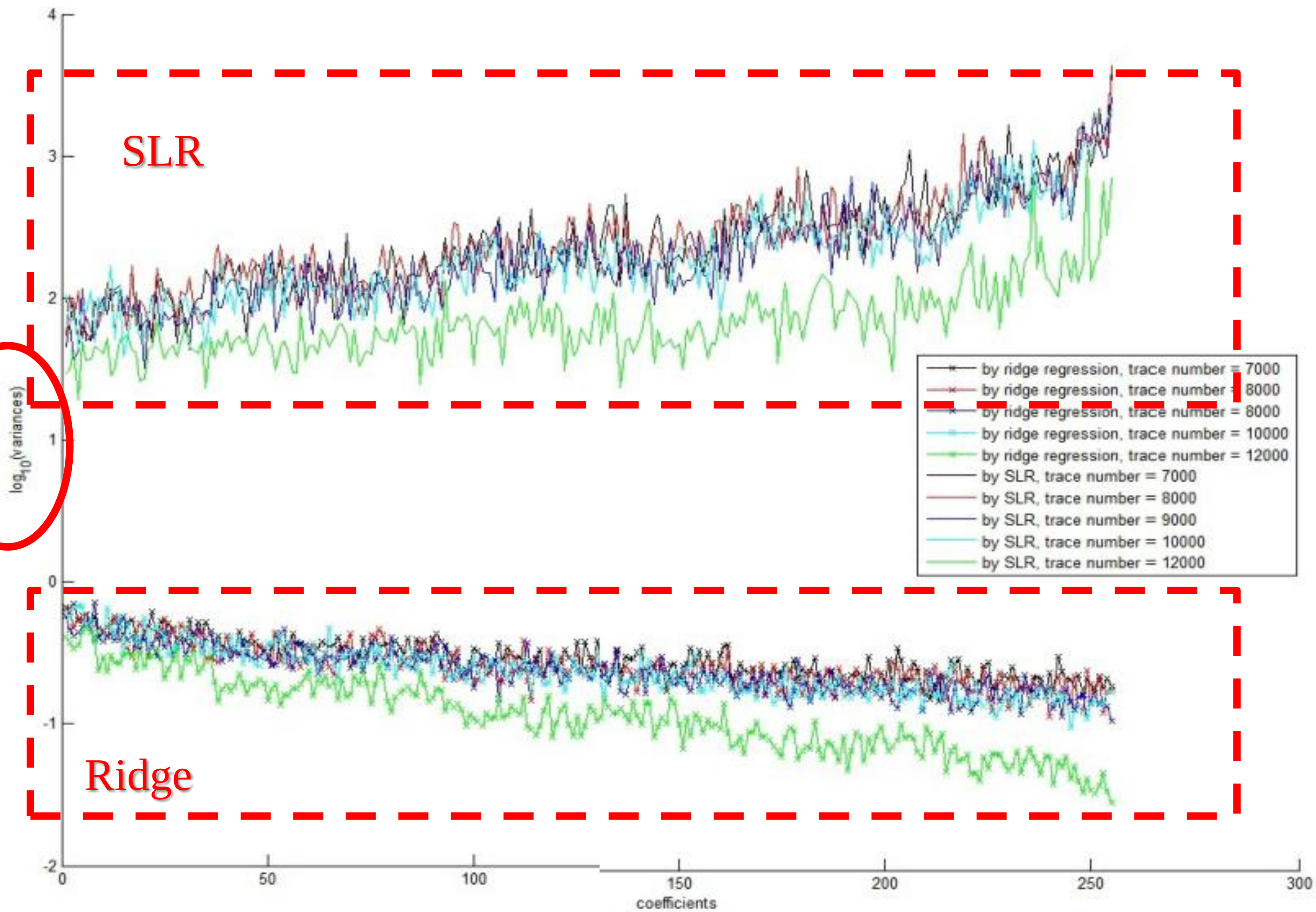
1. Background: generic-emulating DPA
2. Two new generic-emulating distinguishers
3. Improvement using cross-validation
4. Experimental results

4.1.1 SLR-based Distinguisher is Not Stable

High variance of coefficients

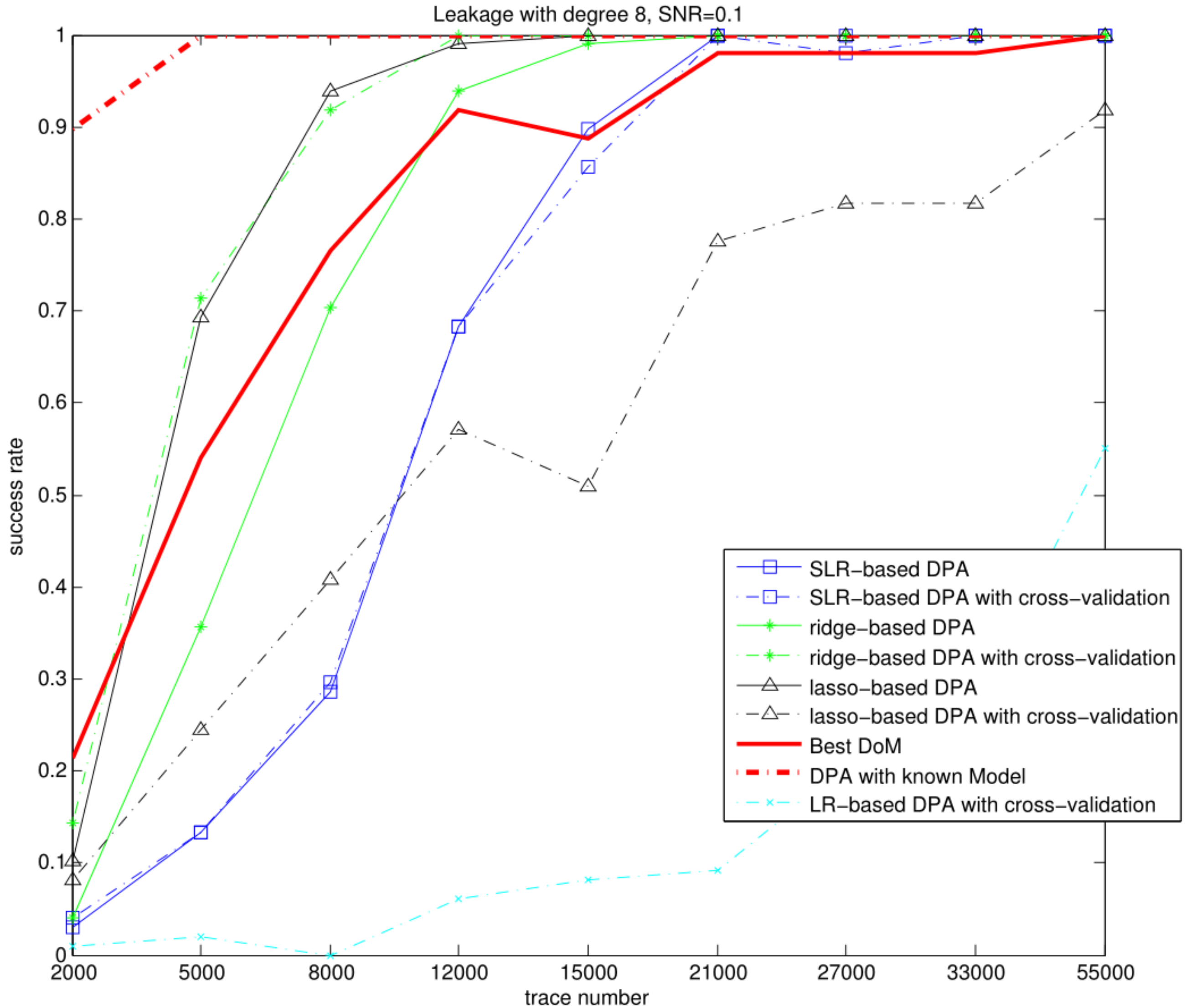
Unstable

Logarithmic



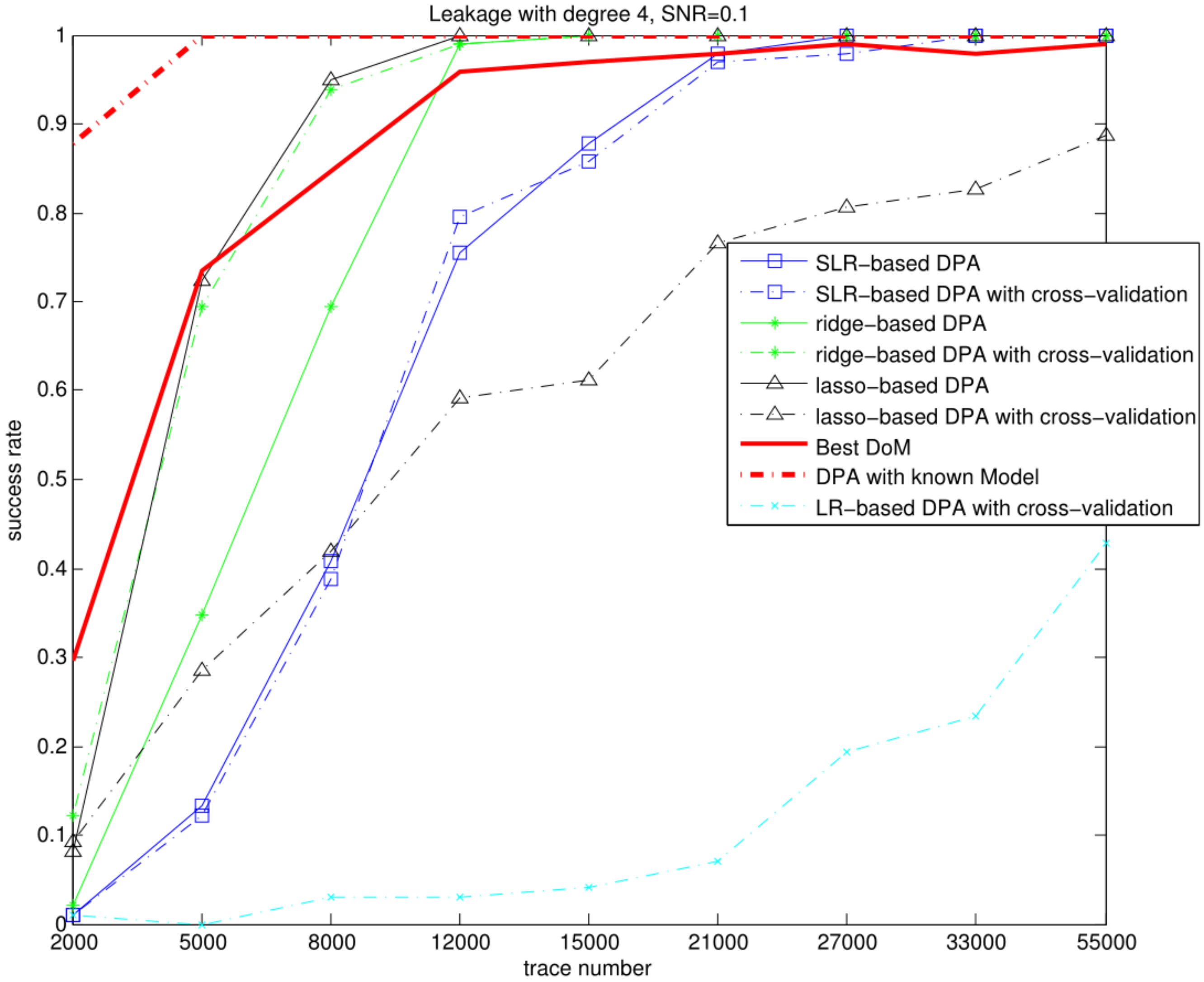
4.1.2 A Comparison of Various Attacks

- Leakage with degree 8
- Ridge-based DPA with C-V and lasso-based DPA are best
- New generic-emulating DPAs perform better than SLR-based One
- C-V improves the ridge-based DPA



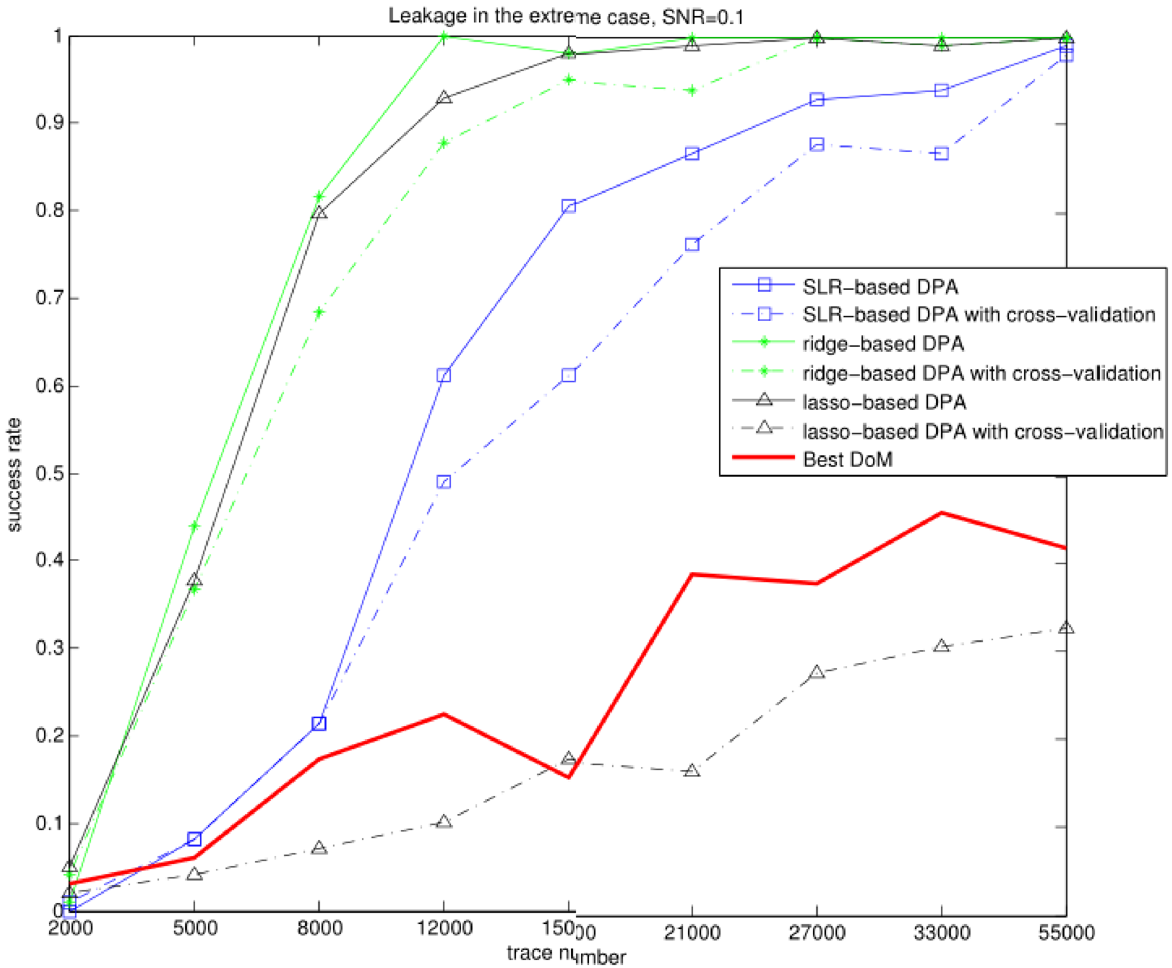
4.1.2 A Comparison of Various Attacks

- Leakage with degree 4
- The Best DoM becomes better in lower degree leakage



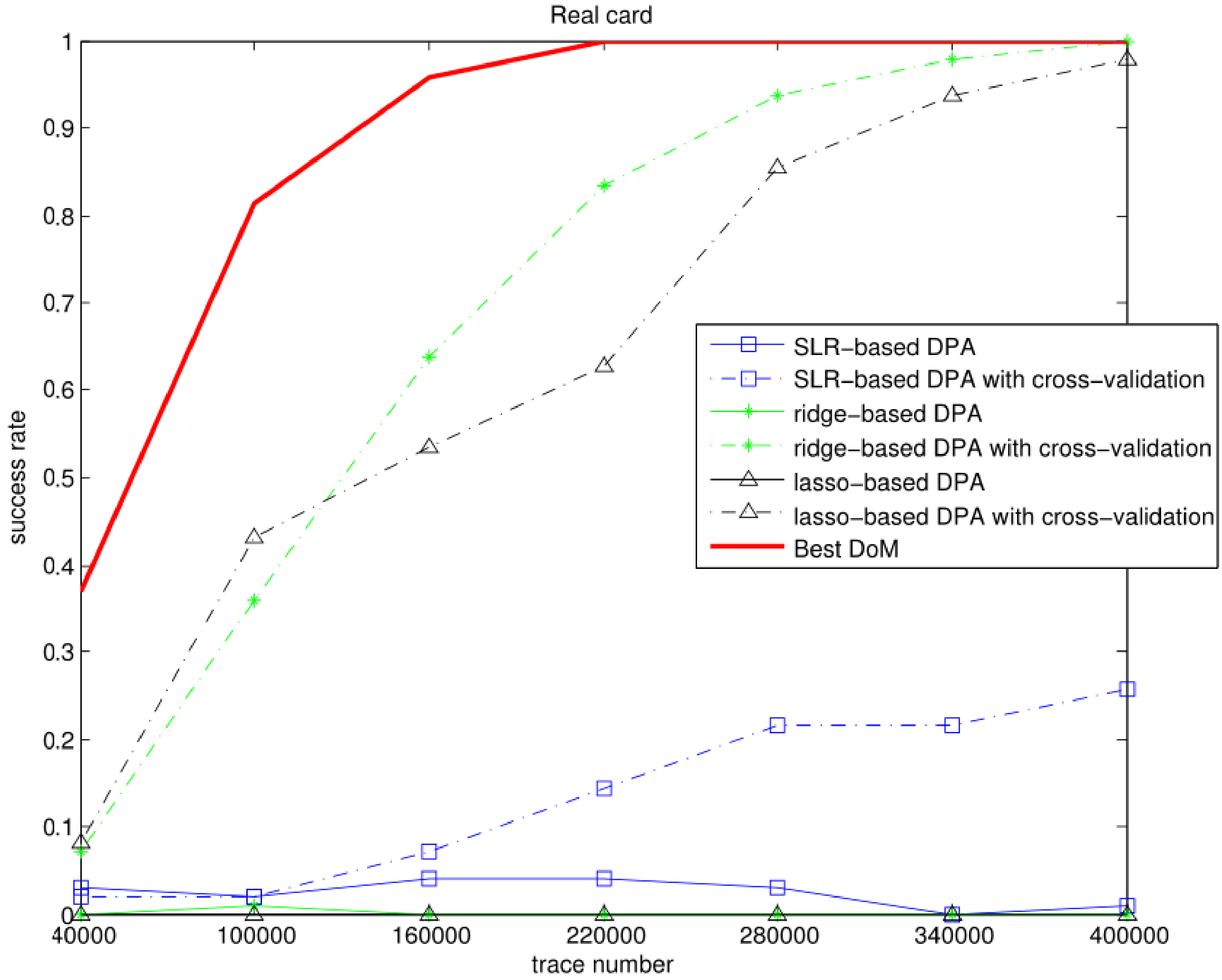
4.1.3 Attacks Against Some Artificial Leakage Function

- All low degree terms (<4) are discarded.
- Best DoM attack behaves poorly
- The generic-emulating DPAs are not affected.



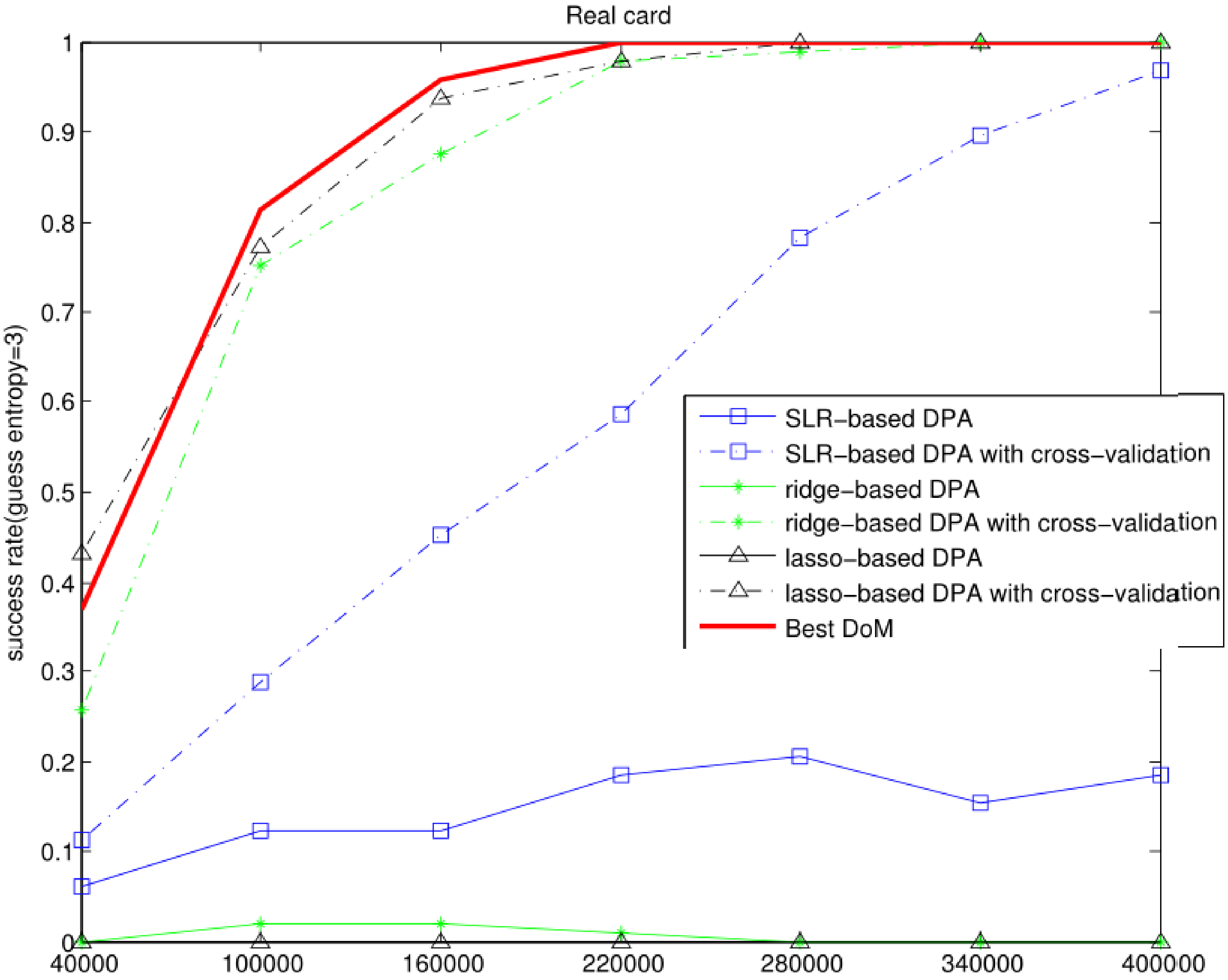
4.2 Experiments on Smart Cards

- Microscale ASIC implementation
- 1st order success rates
- C-V significantly improves the performance of generic-emulating DPAs



4.2 Experiments on Smart Cards

- 8th-order success rates for better alignment with the best DoM attack
- Ridge-based and lasso-based DPAs (both with C-V) are very close to best DoM.



Conclusion

- **Making generic-emulating DPA practicable**
  - Ridge-based and lasso-based distinguishers → more stable
  - Cross-validation →generic-emulating DPAs can be significantly improved

Thank you!