Verifying Snapping Mechanism - Floating Point Implementation Version

In order to verify the differential privacy property of an implementation of the snapping mechanism [5], we follow the logic rules designed from [1] and the floating point error semantics from [7, 4, 2, 6].

1 Preliminary Definitions

Definition 1 (Laplace mechanism [3])
Let $\epsilon > 0$. The Laplace mechanism $\mathcal{L}_\epsilon$: $\mathbb{R} \to \mathsf{Distr}(\mathbb{R})$ is defined by $\mathcal{L}(t) = t + v$, where $v \in \mathbb{R}$ is drawn from the Laplace distribution $\mathsf{laplce}(\frac{1}{\epsilon})$.

2 Syntax

Following are the syntax of the system. The circled operators are rounded operation in floating point computation.

Expr.	e	::=	$c \mid x \mid f(x) \mid e_1 \oplus e_2 \mid e_1 \otimes e_2 \mid e_1 \ominus e_2 \mid e_1 \oslash e_2 \mid ⓛⓝ(e) \mid x \xleftarrow{\$} \mu$
Value	v	::=	$c \mid r$
Distribution	μ	::=	$\mathsf{laplce} \mid \mathsf{unif} \mid \mathtt{bernoulli}$
Error	err	::=	(e_1, e_2)
Condition	Φ	::=	$\mathtt{true} \mid \mathtt{false} \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2$

Definition 2 ($\mathsf{Snap}(a) : A \to \mathsf{Distr}(B)$)
The ideal Snapping mechanism $\mathsf{Snap}(a)$ is defined as:

$$u \xleftarrow{\$} \mu; y = ⓛⓝ(u) \oslash \epsilon; s \xleftarrow{\$} \{-1, 1\}; z = s \otimes y; x = f(a); w = x \oplus z; w' = \lfloor w \rceil_\Lambda; r = \mathsf{clamp}_B(w')$$

where f is the query function over input $a \in A$, ϵ is the privacy budget, B is the clamping bound and Λ is the rounding argument satisfying $\lambda = 2^k$ where 2^k is the smallest power of 2 greater or equal to the $\frac{1}{\epsilon}$.

3 Semantics

The big step semantics with floating point computation error are shown in Figure. 1.

The big step semantics with relative floating point computation error are shown in Figure. 2.

$$\frac{(e_1, err, \Phi) \Downarrow (v_1, err_1, \Phi_1) \qquad (e_2, err, \Phi) \Downarrow (v_2, err_2, \Phi_2)}{(e_1 \oplus e_2, err, \Phi) \Downarrow (v_1 + v_2, err_1 \uplus err_2 \uplus err, \Phi_1 \wedge \Phi_2 \wedge \Phi)} \text{ PLUS} \qquad \cdots$$

Figure 1: Semantics with Absolutes Floating Point Error

$$\frac{c = \mathtt{fl}(r)}{r \Downarrow c, (c(1+\eta), c(1-\eta))} \text{ VAL} \qquad \frac{e_1 \Downarrow v_1 \qquad e_2 \Downarrow v_2}{e_1 \oplus e_2 \Downarrow (v_1 + v_2)(1+\eta)} \text{ PLUS} \qquad \frac{e_1 \Downarrow v_1 \qquad e_2 \Downarrow v_2}{e_1 \otimes e_2 \Downarrow (v_1 \times v_2)(1+\eta)} \text{ TIMES}$$

$$\frac{e_1 \Downarrow v_1 \qquad e_2 \Downarrow v_2}{e_1 \ominus e_2 \Downarrow (v_1 - v_2)(1+\eta)} \text{ SUB}$$

Figure 2: Semantics with Relative Floating Point Error

4 Soundness Theorems

Theorem 1 (The Snap mechanism is ϵ−differentially private)
Consider $\mathsf{Snap}(a)$ defined as before, if $\mathsf{Snap}(a) = x$ given database a and privacy parameter ϵ, then its actual privacy loss is bounded by $\epsilon + 12x\epsilon\eta + 2\eta$

Proof. Given $\mathsf{Snap}(a) = x$ and parameter ϵ, we consider a' be the adjacent database of a satisfying $|f(a) - f(a')| \leq 1$. Without loss of generalization, we assume $f(a) + 1 = f(a')$ ($\diamond$). The proof is developed by cases of the output of $\mathsf{Snap}(a)$ mechanism.

case $\boldsymbol{x = -B}$

Let b be the largest number rounded by Λ that is smaller than B. Based on the proof of the ideal version, the derivation of this case given $\mathsf{Snap}(a) = \mathsf{Snap}(a') = x$ is shown as following:

$$\frac{\dfrac{u \in \left(0, ⓔ^{\epsilon \otimes (-b \ominus \frac{\Lambda}{2} \ominus f(a))}\right) \sim u' \in \left(0, ⓔ^{\epsilon \otimes (-b \ominus \frac{\Lambda}{2} \ominus f(a'))}\right)}{\cdots}}{\mathsf{Snap}(a) = x \sim \mathsf{Snap}(a') = x}$$

Following the semantics in Figure 2, we have following evaluation results:

$$u \in \left[0, (\underline{v}, \bar{v})\right) \sim u' \in \left[0, (\underline{v}', \bar{v}')\right),$$

[[where $\underline{v}, \bar{v}, \underline{v}' and \bar{v}'$ have following values:]]

case $\boldsymbol{x \in (-B, \lfloor f(a) \rceil_\Lambda)}$

The derivation of this case is shown as following:

$$\frac{\dfrac{\dfrac{u \in \left[ⓔ^{\epsilon \otimes (x \ominus \frac{\Lambda}{2} \ominus f(a))}, ⓔ^{\epsilon \otimes (x \oplus \frac{\Lambda}{2} \ominus f(a))}\right) \sim u' \in \left[ⓔ^{\epsilon \otimes (x \ominus \frac{\Lambda}{2} \ominus f(a'))}, ⓔ^{\epsilon \otimes (x \oplus \frac{\Lambda}{2} \ominus f(a'))}\right)}{\cdots}}{\mathsf{Snap}''(a) \in [x \ominus \frac{\Lambda}{2} \ominus f(a), x \oplus \frac{\Lambda}{2} \ominus f(a)) \sim_{\_} \mathsf{Snap}''(a') \in [x \ominus \frac{\Lambda}{2} \ominus f(a'), x \oplus \frac{\Lambda}{2} \ominus f(a'))}}{\dfrac{\mathsf{Snap}'(a) \in [x \ominus \frac{\Lambda}{2}, x \oplus \frac{\Lambda}{2}) \sim_{\_} \mathsf{Snap}'(a') \in [x \ominus \frac{\Lambda}{2}, x \oplus \frac{\Lambda}{2})}{\mathsf{Snap}(a) = x \sim \mathsf{Snap}(a') = x}}$$

Following the semantics in Figure 2, we have following evaluation results:

$$u \in \big[(\underline{v}_1, \bar{v}_1), (\underline{v}_2, \bar{v}_2)\big) \sim u' \in \big[(\underline{v}'_1, \bar{v}'_1), (\underline{v}'_2, \bar{v}'_2)\big),$$

[[where $\underline{v}_1, \bar{v}_1, \underline{v}_2, \bar{v}_2, \underline{v}'_1, \bar{v}'_1, \underline{v}'_2 and \bar{v}'_2$ have following values:

$$u \in \big[((1-\eta)e^{\epsilon(x-\frac{\Lambda}{2}-f(a))(1-\eta)^2}, (1+\eta)e^{\epsilon(x-\frac{\Lambda}{2}-f(a))(1+\eta)^2}), ((1-\eta)e^{\epsilon(x+\frac{\Lambda}{2}-f(a))(1-\eta)^2}, (1+\eta)e^{\epsilon(x+\frac{\Lambda}{2}-f(a))(1+\eta)^2})\big)$$
$$\sim u' \in \big[((1-\eta)e^{\epsilon(x-\frac{\Lambda}{2}-f(a'))(1-\eta)^2}, (1+\eta)e^{\epsilon(x-\frac{\Lambda}{2}-f(a'))(1+\eta)^2}), ((1-\eta)e^{\epsilon(x+\frac{\Lambda}{2}-f(a'))(1-\eta)^2}, (1+\eta)e^{\epsilon(x+\frac{\Lambda}{2}-f(a'))(1+\eta)^2})\big)$$

Given that the probability is equivalent to the length of the range, we have the ratio between u and u' is bounded by:

$$\frac{u}{u'} \leq \frac{\bar{v}_2 - \underline{v}_1}{\underline{v}'_2 - \bar{v}'_1} \leq \epsilon + 12x\epsilon\eta + 2\eta$$

By the AxUnif rule, we have the actual privacy loss is bounded by the same value.]]

case $\boldsymbol{x = \lfloor f(a) \rceil_\Lambda}$

case $\boldsymbol{x \in (\lfloor f(a) \rceil_\Lambda, \lfloor f(a') \rceil_\Lambda)}$

case $\boldsymbol{x = \lfloor f(a') \rceil_\Lambda}$

case $\boldsymbol{x \in (\lfloor f(a') \rceil_\Lambda, B)}$

case $\boldsymbol{x = B}$

$\square$

References

[1] Gilles Barthe, Marco Gaboardi, Benjamin Grégoire, Justin Hsu, and Pierre-Yves Strub. Proving differential privacy via probabilistic couplings. In *LICS 2016*.

[2] H. Becker, N. Zyuzin, R. Monat, E. Darulova, M. O. Myreen, and A. Fox. A verified certificate checker for finite-precision error bounds in coq and hol4. In *2018 Formal Methods in Computer Aided Design (FMCAD)*, 2018.

[3] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating Noise to Sensitivity in Private Data Analysis. In *TCC*, 2016.

[4] Matthieu Martel. Semantics of roundoff error propagation in finite precision calculations. *Higher-Order and Symbolic Computation*, 2006.

[5] Ilya Mironov. On significance of the least significant bits for differential privacy. In *CCS 2012*, 2012.

[6] Mariano Moscato, Laura Titolo, Aaron Dutle, and César A. Muñoz. Automatic estimation of verified floating-point round-off errors via static analysis. In Stefano Tonetta, Erwin Schoitsch, and Friedemann Bitsch, editors, *Computer Safety, Reliability, and Security*, 2017.

[7] Tahina Ramananandro, Paul Mountcastle, Benoundefinedt Meister, and Richard Lethin. A unified coq framework for verifying c programs with floating-point computations. In *Proceedings of the 5th ACM SIGPLAN Conference on Certified Programs and Proofs (CPP)*. Association for Computing Machinery, 2016.