Small Generic Hardcore Subsets for the Discrete Logarithm: Short Secret DL-Keys

C.P. Schnorr
Fachbereich Mathematik/Informatik
Universität Frankfurt, Germany
schnorr@cs.uni-frankfurt.de

Abstract for rump and poster session

Let \( G \) be a group of prime order \( q \) with generator \( g \). We study hardcore subsets \( H \subset G \) of the discrete logarithm (DL) \( \log_g \) in the model of generic algorithms. In this model we count group operations such as multiplication, division while computations with non-group data are for free. It is known from Nechaev (1994) and Shoup (1997) that generic DL-algorithms for the entire group \( G \) must perform \( \Omega(\sqrt{q}) \) generic steps.

Main results. Let \( m = \#H \) denote the size of \( H \). We show that the generic DL-complexity is at least \( m^2 + o(m) \) for almost all \( H \) of size \( m \leq \sqrt{q} \). On the other hand \( \lceil m^2 \rceil + 1 \) generic steps are always sufficient. Thus the generic DL-complexity is \( \frac{m^2}{q} + o(m) \) for almost all subsets \( H \subset G \) of size \( m \leq \sqrt{q} \). For \( m = \sqrt{q} \) the generic DL-complexity is \( \frac{1}{2\sqrt{q}} + o(\sqrt{q}) \), i.e., about \( \frac{1}{2\sqrt{q}} \) times the generic DL-complexity \( \sqrt{2q} \) for the entire group \( G \). Interestingly, our generic lower bounds hold for arbitrary multivariate exponentiations and not just for multiplications/division.

Short secret keys. Our main result justifies to generate secret keys of DL-cryptosystems from random seeds with \( \frac{1}{2} \log_2 q \) bits. For this expand a random integer \( x' \in_R [0, \sqrt{q}] \) of \( \frac{1}{2} \log_2 q \) bits using a strong hash function \( SH \) into a pseudo-random integer \( SH(x') \in_R [0, q] \). The corresponding pair \( x', g^{SH(x')} \) is a DL-key pair that is — for generic attacks — nearly as strong as pairs \( x, g^x \) for truly random \( x \in_R [0, q] \). This is because the generic DL-complexity is for almost all subsets \( H \subset G \) of size \( \sqrt{q} \) about \( \frac{1}{2\sqrt{q}} \) times the generic DL-complexity for \( G \). Clearly, a strong hash function \( SH \) yields a set of pseudo-random public keys \( SH[0, \sqrt{q}] \subset [0, q] \) of size \( \Omega(\sqrt{q}) \) since otherwise collisions \( SH(x') = SH(x'') \) can be constructed using less than \( \Omega(\sqrt{q}) \) function evaluations \( [0, \sqrt{q}] \ni x \mapsto SH(x) \). Moreover, it is reasonable to assume that the set \( SH[0, \sqrt{q}] \) does not fall into the exceptional class of subsets \( H \subset G \) where \( \log_g \) is easy in the generic model. Generating secret keys from short random seeds can be practical if a strong hash function \( SH \) is at hand anyway. Now, there is a theoretical justification that seeds of length \( \frac{1}{2} \log_2 q \) are nearly of the highest security level while shorter seeds are
less secure.

Moreover, as the generic DL-complexity is \( \frac{m^2}{2} + o(m) \) for almost all subsets \( H \subset G \) of size \( m \), it is sufficient to generate secret DL-keys from seeds \( x' \) ranging over a set of size \( m \) that is so large that \( \frac{m^2}{2} \) generic steps are infeasible — at present \( m \geq 2^{80} \) is sufficient.

*Fast pseudo-random exponentiation.* An intriguing challenge along this line is to replace \( SH \) in the short secret key representation by a pseudo-random function \( F \) that speeds up the exponentiation \( x' \mapsto g^{F(x')} \). We will study this problem in another submission.