4.2 Implementation results

Since strong randomness is required for masking to lead to its expected security improvements, we finally repeated our performance evaluations assuming a reasonable cost for producing each random byte. Concretely, this cost may be quite device-dependent. Hence, we first considered an optimistic 10 cycles for each of these generations (excluding the memory accesses), which roughly corresponds to the execution of two AES rounds for producing 16 bytes of pseudorandomness (excluding the key scheduling). As a complement, we considered a pessimistic 100 cycles per byte which is more in line with the cost required in high-security chips. Besides, and in order to optimize the randomness requirements, we also modified the addition chain for the inversion in order to minimize this additional criteria, as proposed in [6]. We then compared the schemes of Section 2 again, namely **RivP**, **KHL**, **GPQ** and **RocP**$\star$, as well as the MPC-based scheme using the multiplication of Algorithm 2 using a single secret per polynomial, and the best packed sharing scheme from the previous section (i.e. the most efficient solution for each security degree), considering compact codes. As illustrated in Figure 4, the cost of the random generation shifts the performance curves. But since all algorithms have a cost in randomness that is quadratic in the order of the masking scheme, this shift does not contradict the previous observations. One can just observe that the order for which packed secret sharing becomes a useful alternative is delayed depending on the cost of the randomness generation. For the rest, the gap between **RivP**, **KHL**, **GPQ** and MPC-based masking remains large, but has been significantly reduced thanks to our optimizations.

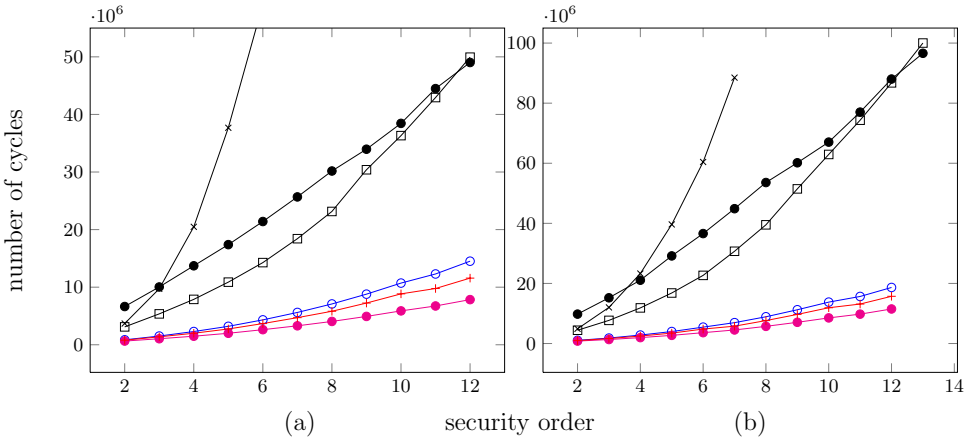

Fig. 1: Cycle counts for masked AES implementations with time to generate random. (a) 10 clock cycles per random byte, (b) 100 clock cycles per random byte. The curve —+— is for **KHL**, the curve —×— is for **RocP**$\star$, the curve —○— is for **RivP**, the curve —●— is for **GPQ**, the curve —□— is for the multiplication of Algorithm 2 with a single secret per polynomial and the curve —●— is for the best packed secret sharing.