Research Directions on the Complexity of Boolean Circuits for Codes and Cryptography

Luís T. A. N. Brandão, Çağdaş Çalık, Morris Dworkin, Nathan Dykas, René Peralta, and Meltem Sönmez Turan

National Institute of Standards and Technology

The circuit complexity of Boolean functions is of fundamental interest for practical cryptography. For example, the number of non-linear gates in a Boolean circuit has a direct impact on the efficiency of high-level cryptographic primitives (such as zero-knowledge proofs and secure multi-party computation based on such a circuit), and on the size of corresponding side-channel resistant circuits. Also, the number of linear gates directly impacts the cost (time and space) of hardware implementations.

In this talk, we will describe some of the recent results that the circuit complexity team at the NIST Cryptographic Technology Group achieved in three research directions: obtaining new recurrence relations for binary polynomial multiplication; finding new constructive bounds for multiplicative complexity of symmetric Boolean functions; and improving the complexity of Reed-Solomon codes.

Binary polynomial multiplication. Several applications, such as elliptic curve cryptography over finite fields of characteristic 2, benefit from the ability to design smaller circuits for binary polynomial multiplication. Our group has established new Karatsuba-like recurrence relations of the form $M(kn) \leq c_1 M(n) + c_2 n - c_3$, for $k = 4, ..., 8$, where $M(n)$ is the number of gates necessary and sufficient for multiplying two binary polynomials of degree $n - 1$, and c_1, c_2 and c_3 are constants.

Symmetric Boolean functions. Our group developed new techniques that enable a recursive construction of circuits with low number of `AND` gates for several symmetric Boolean functions of interest, including elementary symmetric (Σ_i^n); exactly-counting (E_i^n); and threshold (T_i^n) Boolean functions.

Reed-Solomon codes. The Reed-Solomon code **RS(255,223)** is widely used in communications and in data storage and retrieval. A standard implementation uses 32 linear circuits with about 24 gates each, yielding a total of 768 gates. Our linear circuit optimization techniques yielded a circuit with only 412 gates.