Evolutionary design of domain specific non-cryptographic hash functions

Hash functions are evolved using the genetic programming algorithm. Individuals are represented by an abstract syntax tree where nodes are represented by common hashing operations. Leaves may represent and IP address octet or an ephemeral random constant.

Each individual is assessed in terms of collision resistance. The number of successfully hashed IP addresses from the given data-set is measured. The more IP addresses individual hashes, the better its fitness is.

1. Comparison of the proposed IPHash genetic programming algorithm with random search algorithms and the human-created MurmurHash and diversity of the experiment. The result is an average of 50 independent runs.

2. The relation of various terminal and non-terminal sets to the resulting average (50 runs) best fitness. If the multiplication (*), addition (+) and rotation (>>>) are preserved, evolved hash functions are well performing.

3. Example of an evolved hash function. Such function is small in terms of operation count but well performing on the given dataset.