MY HASH IS NICE PROTOCOL

Make your Bitcoin transactions nicer and earn MHIN.

By Ouziel Slama

1 Motivations

- For aesthetic reasons. To make each transaction not just a financial operation, but a brushstroke in the blockchain's grand canvas of mathematical beauty.
- These cascading patterns of zeros would not only create beauty but could also enhance compression, potentially streamlining the blockchain's storage and processing efficiency.
- Anyone can earn MHIN by creating beautiful transactions - no single-winner-per-block like in Bitcoin block mining!
- If successful, MHIN tokens could eventually reimburse transaction fees - rewarding those who make nice looking transactions!

2 MHIN mining

To mine MHIN you must broadcast a Bitcoin transaction whose txid starts with at least 6 zeros. The reward is calculated based on how your transaction compares to the nicest transaction in the block:

- In a given block, transactions starting with the most zeros earn 4096 MHIN
- Transactions with one zero less than the best transactions earn 4096/16 or 256 MHIN
- Transactions with two fewer zeros earn 4096 / 16 /16 or 16 MHIN
- etc.

The formula used is therefore as follows:

```
reward = 4096 / 16 ^ (max_zero_count - zero_count)
```

With `max_zero_count` equal to the number of zeros which start the best transaction and `zero_count` the number of zeros which start the transaction for which we calculate the reward.

Note: Coinbase transactions are not eligible for MHIN rewards.

3 MHIN distribution

MHINs earned with a transaction starting with 6 or more zeros are distributed to UTXOs. The distribution is carried out as follows:

- If there is a single non-OP_RETURN UTXO it receives the entire reward.
- If there are two or more non-OP_RETURN UTXOs, the reward is distributed to all UTXOs, except the last one, in proportion to the value of each UTXO
- The calculations being made only with integers, the possible remainder of the division is distributed to the first non-OP_RETURN UTXO.

For example, if a transaction earning 256 MHIN contains 4 outputs with 500, 500, 500 and 2000 Satoshis respectively, the first output receives 86 MHIN of the reward, the second and third 85 MHIN.

4 Moving MHIN

When UTXOs with attached MHINs are spent, the MHINs are distributed to the new UTXOs in the transaction. There are two methods for distributing MHINs when moving them:

4.1 Method 1: Automatic Proportional Distribution

By default, distribution is done in exactly the same way as rewards - proportionally based on the Bitcoin values of the output UTXOs, excluding the last output if there are multiple outputs.

4.2 Method 2: Custom Distribution via OP_RETURN

You can specify exactly how MHINs should be distributed by including an OP_RETURN output in your transaction with custom distribution data. This allows for precise control over MHIN transfers.

4.2.1 OP_RETURN Format:

- The OP_RETURN script must contain data that starts with the 4-byte prefix "MHIN"
- Following the prefix, the data must be encoded in CBOR format
- The CBOR data should represent a vector of unsigned 64-bit integers (Vec)
- Each integer specifies how many MHINs to send to the corresponding output UTXO

4.2.2 Distribution Rules:

- The number of values in the distribution array is automatically adjusted to match the number of non-OP_RETURN outputs
- If the array is too long, extra values are removed
- If the array is too short, zeros are appended
- The total sum of the distribution values cannot exceed the total MHINs being spent
- If the sum is less than the total, the difference is added to the first output
- If the sum exceeds the total, the transaction falls back to proportional distribution
- Newly mined MHIN rewards are always distributed proportionally and then combined with the custom distribution

4.2.3 Example:

If you have 1000 MHINs to distribute across 3 outputs and want to send 600 to the first, 300 to the second, and 100 to the third, your OP_RETURN would contain "MHIN" followed by the CBOR encoding of [600, 300, 100].

Note: If no valid OP_RETURN distribution is found, the transaction automatically uses the proportional distribution method.