Here's a cryptographic primitive that would be very useful to have. We would like to have a vector commitment scheme, which commits to elliptic curve points, and has some algebraic properties with respect to those elliptic curve points

. Some particular nice properties we would ideally like to have are:

- A commitment C

to [H_1, H_2 ... H_n]

is a member of some group, and you can compute C * k

, which becomes a valid commitment to [H_1 * k, H_2 * k ... H_n * k]

, where the *

in H_i * k

is elliptic curve multiplication

- Given commitments C

to [H_1, H_2 ... H_n]

and D

to [I_1, I_2 ... I_n]

, you an compute C + D

which is a valid commitment to [H_1 + I_1, H_2 + I_2 ... H_n + I_n]

, where the +

in H_i + I_i

is elliptic curve addition.

- Given a commitment C

to [H_1, H_2 ... H_n]

linear combination, expressed as a (sparse) list [c_1, c_2 ... c_n]

, you can generate a proof \Pi

that proves that some value Q = H_1 * c_1 + H_2 * c_2 + ... + H_n * c_n

.

Use cases for this include:

- It could be a useful ingredient in making bettertree-structured state commitments
- In eth2 verification, we would like to be able to compute the public key that is the sum of a particular subset of public keys in a long list. Currently, this is done by computing the subset sum manually, but this takes N elliptic curve additions (N ~= 1/32 of the validator set per block). Making a special-purpose proof for this could greatly reduce verification complexity

Currently, the best that I know of that goes in this direction ishttps://eprint.iacr.org/2019/1177.pdf

Weaknesses of this approach that I can see are (i) relatively low concrete efficiency, and (ii) reliance on instantiating target group elements (this is hard because it would require us to agree on a serialization and hence standardize a specific preferred pairing, instead of only requiring bilinearity). It would be nice to find something that avoids these weaknesses!

Using groups other than elliptic curves would also in principle be acceptable.