double joins

Summary

TreeKEM makes it possible to accommodate incomplete trees

This lets us do Remove without double-join… (see mailing list)

… and in a similar way gets us Add without double-join … [Bhargavan 2018]

… and gets us Init basically for free [Barnes 2018]

All of which results in a much simpler set of tree operations

Assume the following:

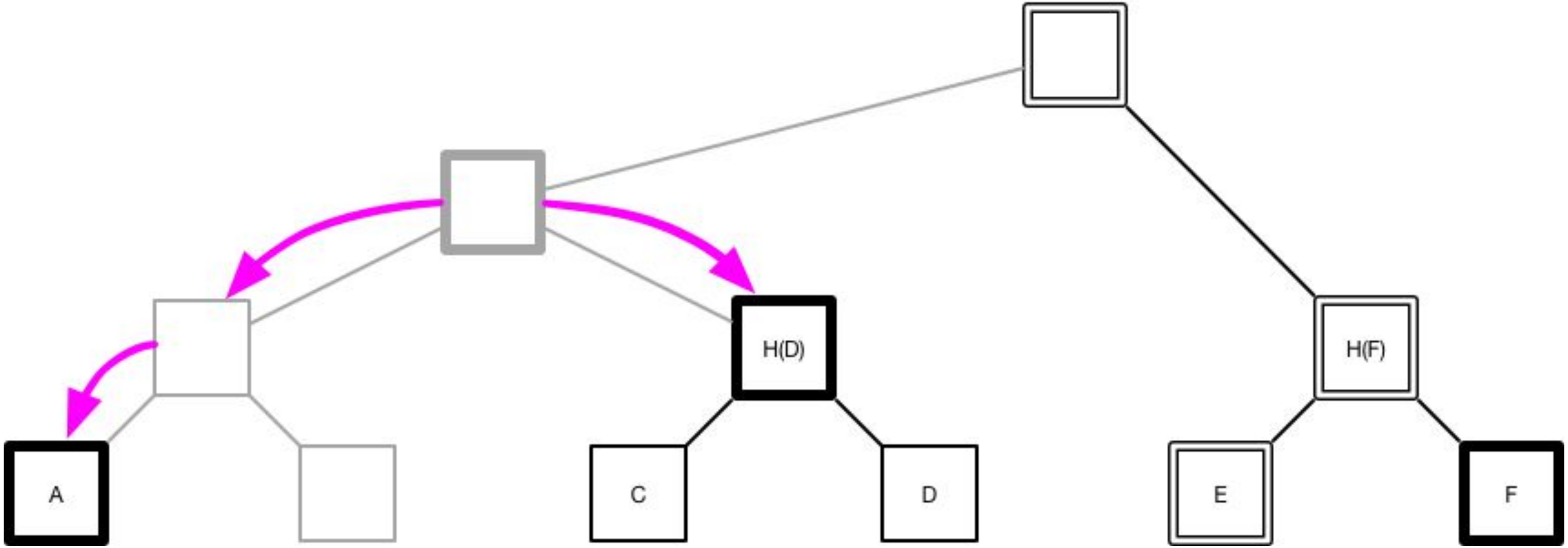
Trees cached by group members can be incomplete, with some nodes blank

If you need to encrypt to a blank node, you instead encrypt to whichever of its children is present

To add / remove:

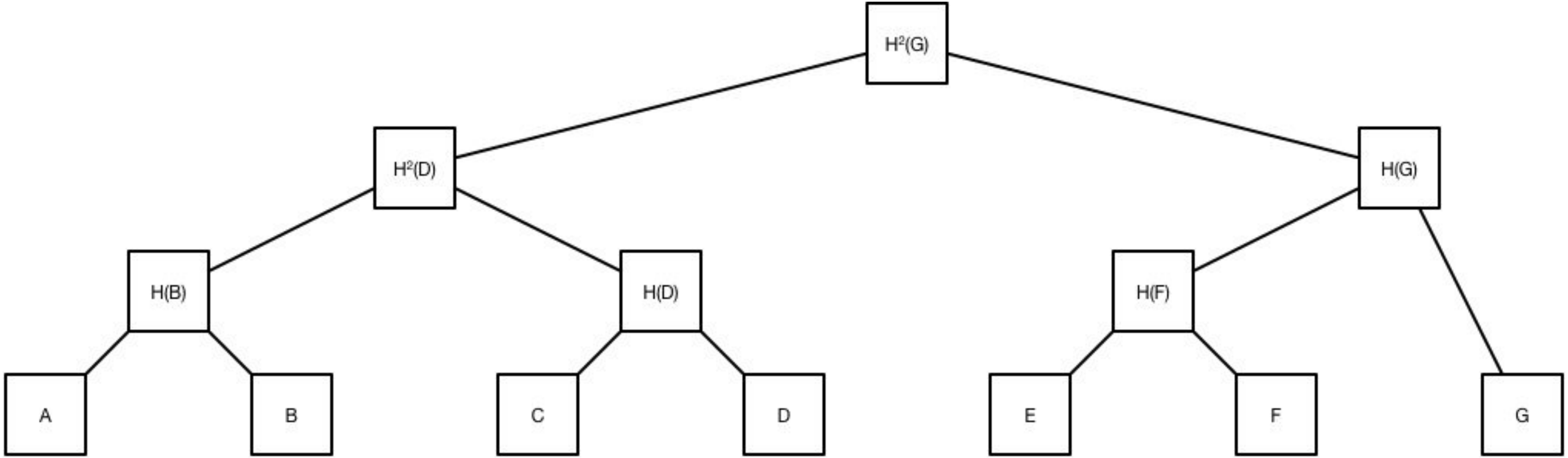
1. Send an update as if you were the targeted node (to introduce fresh entropy)
2. Blank out the nodes on the direct path of that node

Fall through missing nodes to children

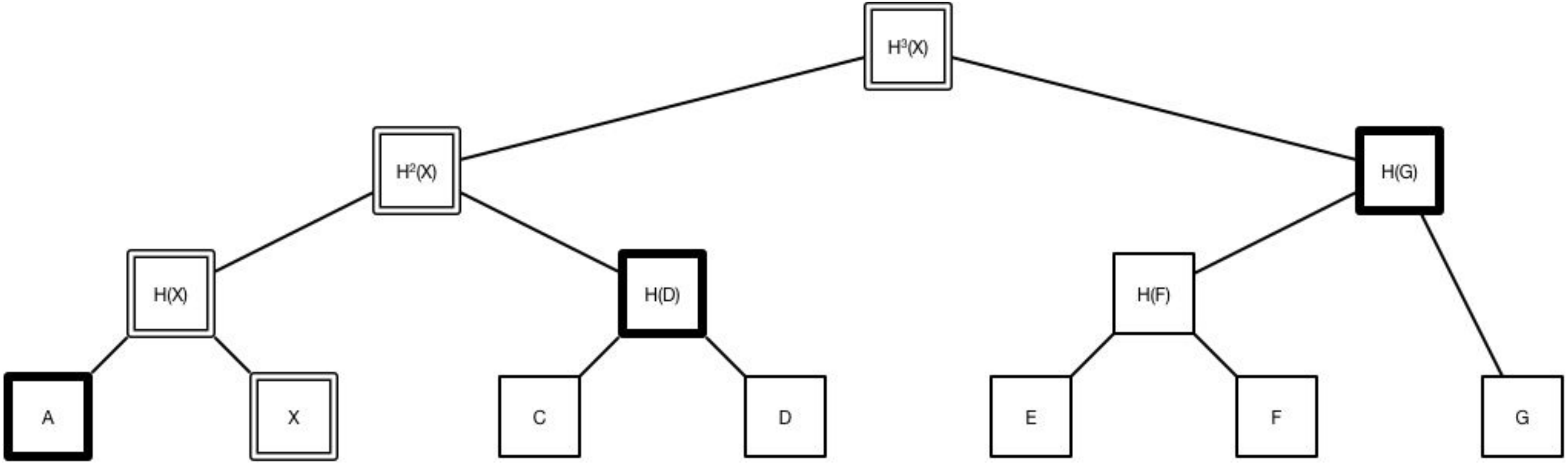


Remove w/o 2J

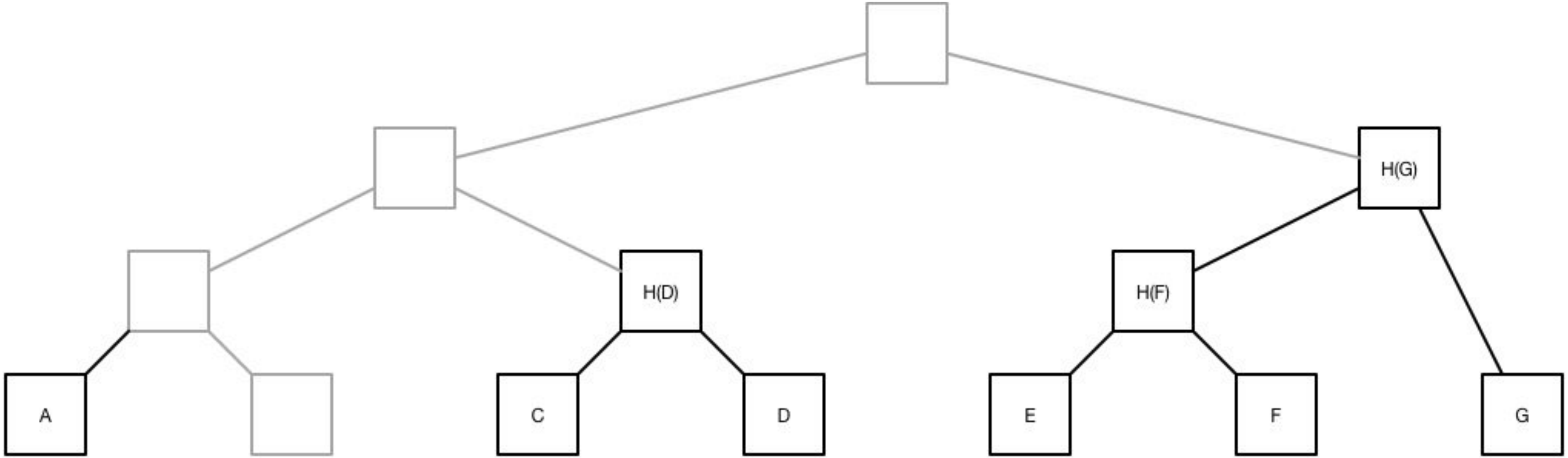
Starting state… we are going to remove B



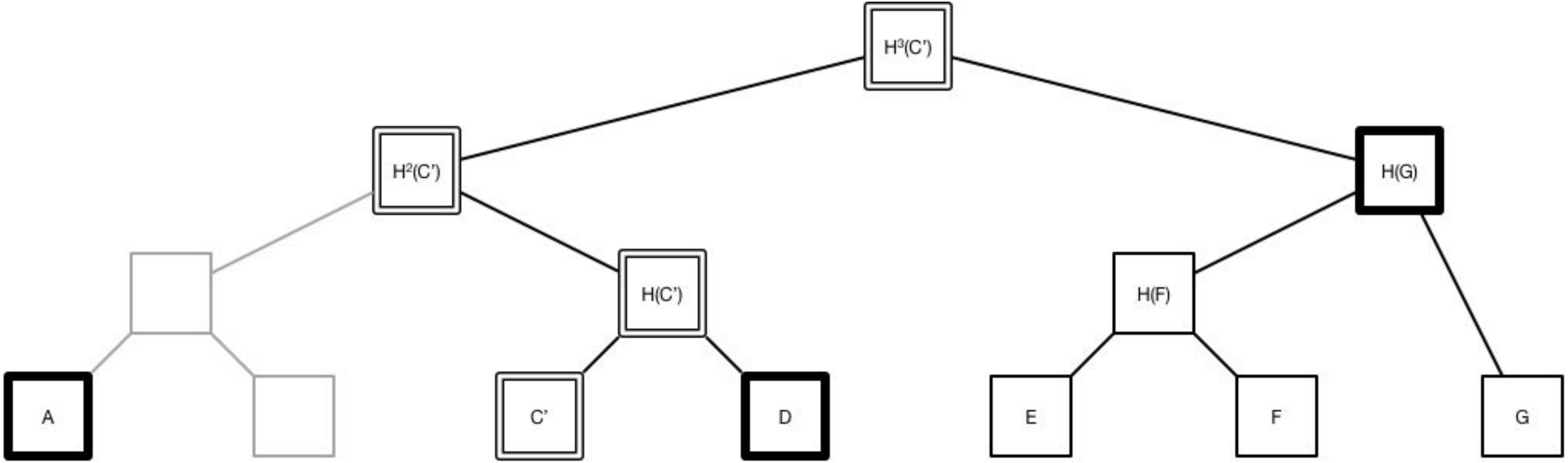
1. Send an update as B



2. Blank out B’s direct path (=> state after Remove)

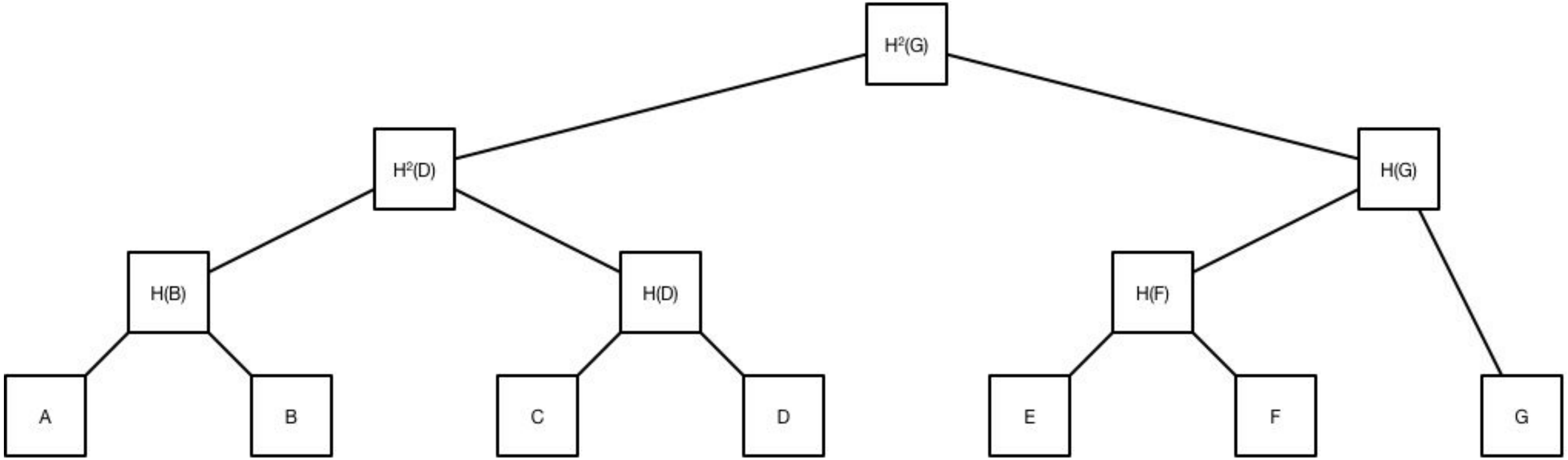


C sends an Update after the removal

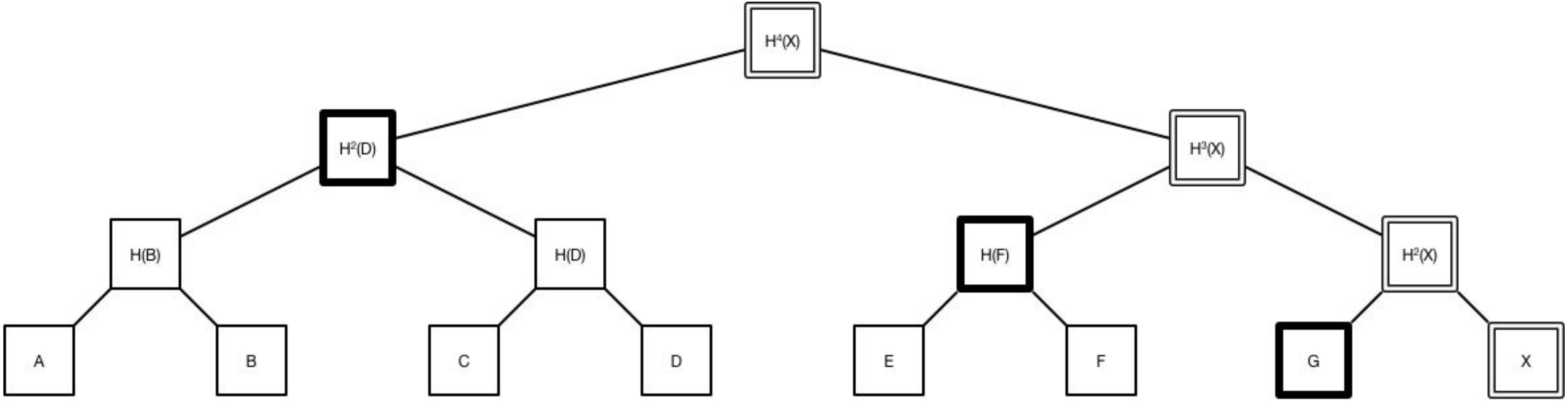


Add w/o 2J

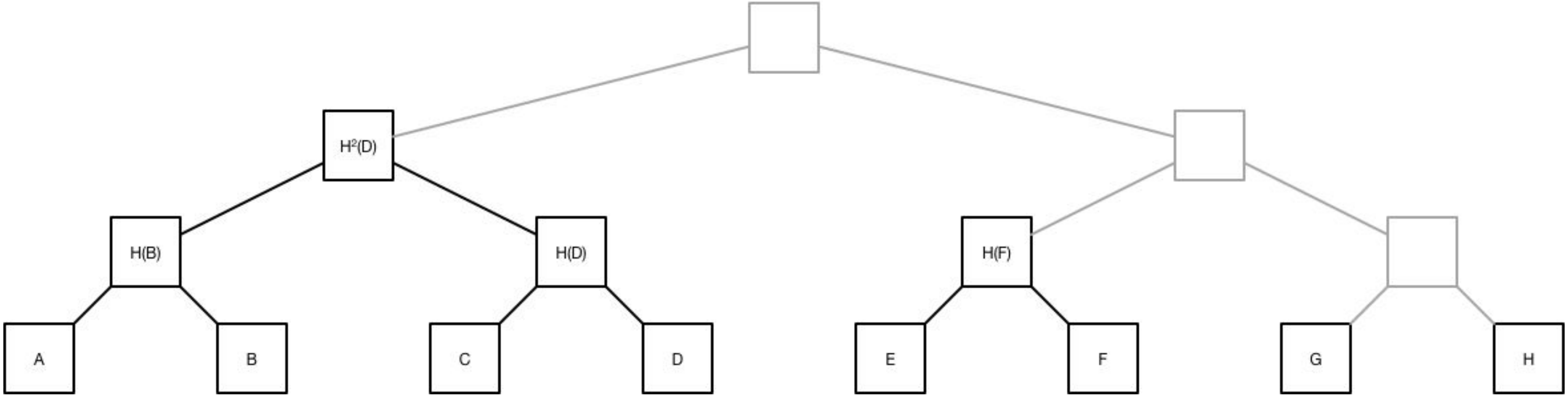
Starting state… we are going to add H



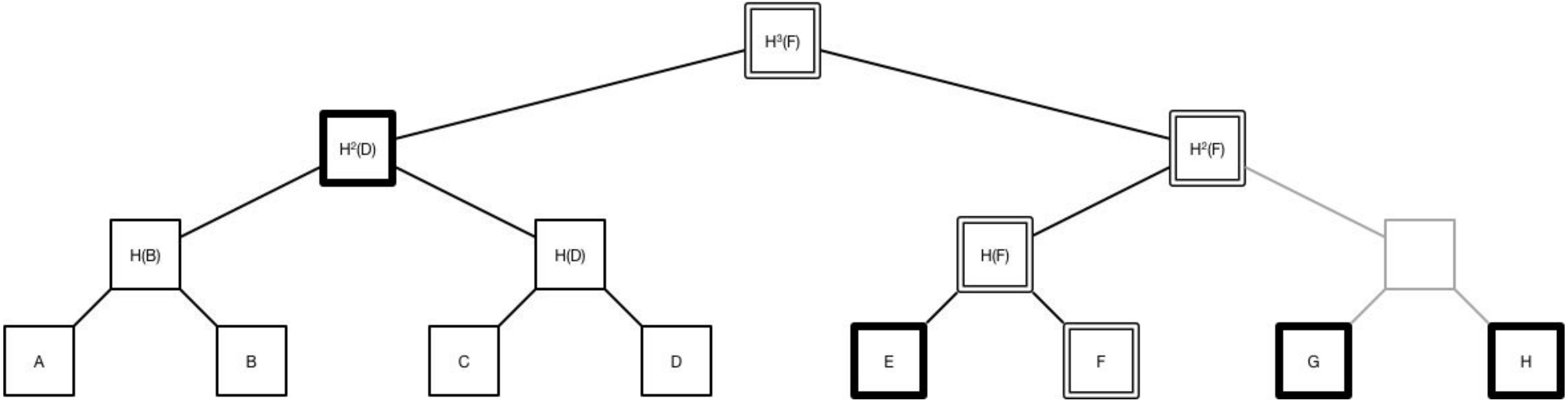
1. Send an update as H



2. Blank out H’s direct path (and put in H’s leaf)



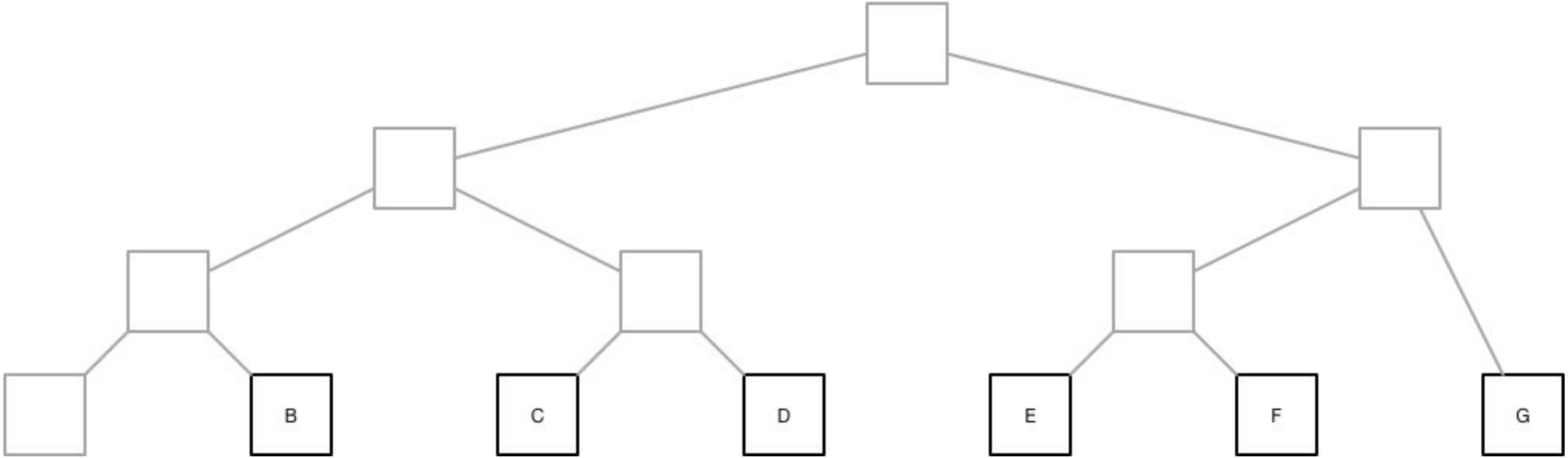
F sends an Update after the addition



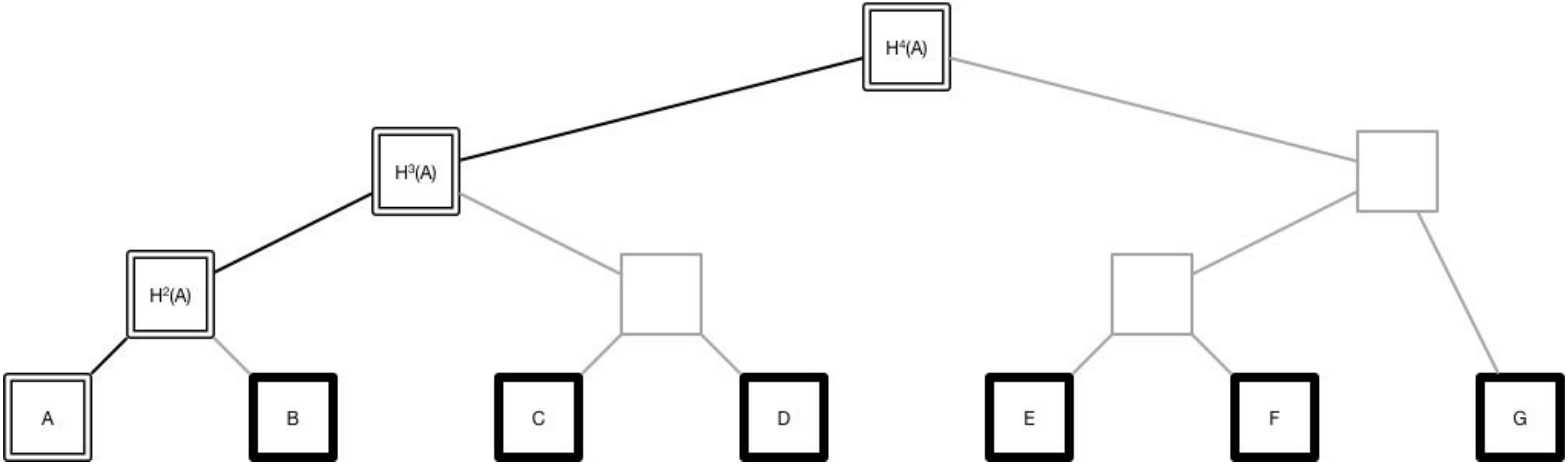
Update is still Update

Init for free

Initialize a tree with leaves from InitKeys



Send an Update from the creator



Protocol implications

Tree Operations

This gives us a very simple “API” for the tree:

1. Encrypt an update to a (possibly incomplete) tree from a specified slot
2. Decrypt an update
3. Blank out a direct path
4. Set a leaf public key

Major impact of incompleteness is more complexity in identifying the targets of encrypt/decrypt operations

Messages and State

Obviously, the GroupOperation messages will need to be updated

This scheme is easiest if everyone caches the whole tree, since it’s then trivial for them to blank out nodes

It can *probably* be made to work with nodes caching only their dirpath/copath

Just need to send the populated children below a node that will be deleted

Questions

Comprehensible?

Worth doing?

Willing to assume members cache the full tree?