It seems to me that using small fields (e.g. 64 bits) for STARKs would have a number of advantages. Specifically:

1. Proof sizes would be smaller. For example, with 256-bit fields, every state register adds about 1KB to proof size. With 64-bit fields, every state register would contribute 1/4 of that.

2. Hash functions would be more efficient. For example, for such hash functions as Rescue/Poseidon, the number of rounds can be reduced by almost 50% if we use a large number of 64-bit registers vs. a small number of 256-bit registers.

3. Computations would run faster. With 64-bit fields we can do all modular math with just a few native instructions.

It seems to me that the main drawback of using small fields is that 256-bit modular arithmetic becomes way more complex. And this makes it more difficult to work with elliptic curves. Are there any other reasons to avoid small fields?

Also, maybe there is an efficient way to do 256-bit modular arithmetic with 64-bit registers?