vulpeculiar musing 13: extending dag representations to bands

vulpecule • 20 Aug 2026

Posets are in one-to-one correspondence with dags, and since semilattices can be viewed as special cases of posets, we have associated representations of semilattices as dags. Now, a semilattice is a commutative band, and so it stands to reason that there might be some way to canonically "complete" each side of a band into a semilattice, and then ideally a band could be represented as a map between dags.

But does this actually work? One issue is that you need a specific ordering in order to split the band into two "sides", i.e. we equip the band with a total ordering, and then construct two derived operations (which as we will see later are not actually necessarily semilattices): - $x \wedge_\ell y = xy$ if $x \leq y$ and $x \wedge_\ell y = yx$ otherwise - $x \wedge_r y = xy$ if $y \leq x$ and $x \wedge_r y = yx$ otherwise.

A canonical partial ordering is one thing, but how do we come up with a canonical *total* ordering of a band? Well, let's see how much it even matters with explicit examples. The only noncommutative bands of order two are the left/right zero bands. These have both elements lying in a single similarity class, so the ordering doesn't actually matter, and in both cases the dagmap representation is just a 2-path being flipped.

One immediate issue is that associativity can actually be broken by "mirroring" the Cayley table - as seen by this ordering of this band:

$$\begin{matrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 2 \end{matrix}$$

However, it turns out that not all bands admit a representation in this manner due to associativity being broken in all possible orderings. The first time this occurs is order 4, where the unique band with this property is also the unique smallest noncommutative self-converse band, but I checked, and it is neither the case that all noncommutative self-converse bands are exceptions, nor that all exceptions are self-converse. So much for this direction, but I still wish to understand the circumstances under which it fails…

But, well, regardless of how well turning bands into dagmaps goes, it is straightforward to turn dagmaps (with join or meet) into bnds… Yet, they are not guaranteed to be associative. So *neither* direction works, alas…