Multi-index Permutations

Given a multi-index $\alpha = (a_1, \ldots, a_n)$ with $\dim(a_i) = d_i$ an arbitrary positive integer, how does one compute $\sigma(\alpha) = (a_{\sigma(1)}, \ldots, a_{\sigma(n)})$ for any $\sigma \in S_n$? We are interested in this question so that we can permute multi-indices in matrices as needed to sort grouped physical indices and bond indices as needed without restricting our algorithms to strict formats. To begin, $\mathrm{id}(\alpha) = \alpha \Rightarrow (1, 2, \ldots, \prod_{j=1}^{n} d_j)$, so we might ask how do we recreate this sequence from the a_i.

When $d_i = 2$ it is easier to visualize in binary: $\alpha_i = (0, 1)$, but in the whole multi-index, this turns into

$$\alpha'_1 \Rightarrow (0, 1, 0, 1, \ldots, 0, 1, 0, 1)$$

$$\alpha'_2 \Rightarrow (0, 0, 1, 1, \ldots, 0, 0, 1, 1)$$

$$\vdots$$

$$\alpha'_n \Rightarrow (0, 0, 0, 0, \ldots, 1, 1, 1, 1)$$

where we obtained this from $I_k = \mathrm{diag}(\mathrm{id}_k)$

$$\alpha'_i = I_{d_n} \otimes \cdots \otimes I_{d_{i+1}} \otimes \alpha_i \otimes I_{d_{i-1}} \otimes \cdots \otimes I_{d_1}$$

But we wont recover $\boldsymbol{\alpha}$ by adding the a_i. It turns out that

$$\boldsymbol{\alpha} = \sum_{i=1}^{n} \left(\prod_{j=1}^{i-1} d_j \right) \alpha'_i$$ :: each term is weighted by the dimension of the multi-index that preceedes it.

To do a permotation, we want to keep the weights: $w_i = \prod_{j=1}^{i-1} d_j$

but modify the structure of the tensor product. The short story is

$$\sigma(\alpha) = \sum_{i=1}^{n} w_i \beta_i' \quad \text{where}$$

$$\beta_i' = I_{d_{\sigma(1)}} \otimes \cdots \otimes I_{d_{\sigma(i-1)}} \otimes \alpha_{\sigma(i)} \otimes I_{d_{\sigma(i+1)}} \otimes \cdots \otimes I_{d_{\sigma(n)}}$$

which situates the index in its location. The fact we keep the weights from before makes this permutation work, otherwise

$$\sum_{i=1} w_{\sigma(i)} \beta_i' = \sum_{i=1} w_{\sigma^{-1}(i)} \alpha'_{\sigma^{-1}(i)} = \sum_{j=1} w_j \alpha_j' = \alpha$$

by noting the bijectivity of permutations and making use of the dummy indices.

That's pretty much it. Implementing this requires a convention of $i, \sigma(i)$ or $\sigma^{-1}(j), j$ for $j = \sigma(i)$. Combinatorics isn't obvious.