TRANSLATION INVARIANT TOTAL ORDERS

GRANT BARKLEY
FALL 2025

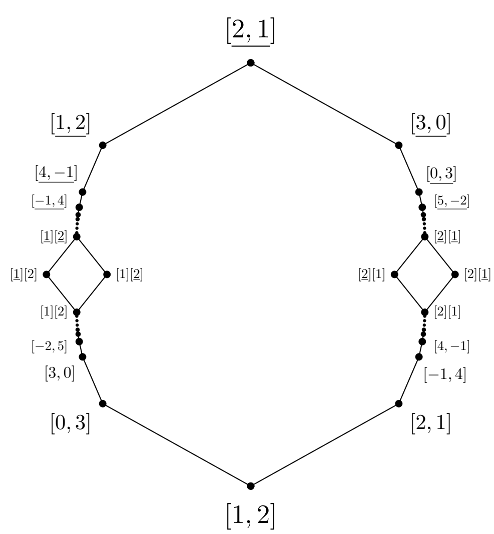
In this project, we will study *translation-invariant total orders* (TITOs). Fix an $n \in \mathbb{N}$. A TITO is a binary relation $\lhd$ on $\mathbb{Z}$ with the following properties:

- $\lhd$ is a *total order*: this means
  - $\lhd$ is transitive: If $i, j, k \in \mathbb{Z}$ are such that $i \lhd j$ and $j \lhd k$, then also $i \lhd k$.
  - $\lhd$ has trichotomy: If $i, j \in \mathbb{Z}$, then exactly one of $i \lhd j$, $j \lhd i$, or $i = j$ is true.
- $\lhd$ is invariant under translation by n: For all $i, j \in \mathbb{Z}$, we have $i \lhd j$ if and only if $i + n \lhd j + n$.

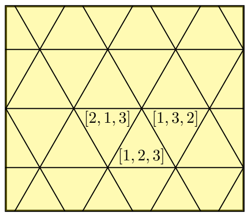
TITOs can be used to model *extended weak order* and *torsion classes* related to type $\widetilde{A}_{n-1}$ affine root systems and Coxeter groups.

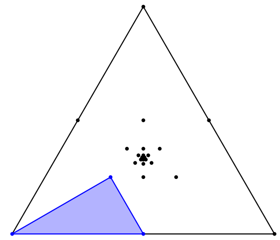
Let TTot_n be the set of TITOs. The goal of this project is to write code which implements TITOs and useful operations or visualizations of them. For example, there is a partial ordering of TTot_n, called *weak order*, which is important for applications. Weak order puts $(\lhd_1) \le (\lhd_2)$ if and only if every inversion of $(\lhd_1)$ is also an inversion of $(\lhd_2)$. An *inversion* of a TITO $(\lhd)$ is a pair of integers (a, b) so that $a < b$ and $b \lhd a$. Because a TITO can have infinitely many inversions, it can be non-trivial to check whether $(\lhd_1) \le (\lhd_2)$. One concrete goal for the project could be to implement this check. We can also consider the *lattice property* of the weak order on TTot_n. Given any two TITOs $(\lhd_1), (\lhd_2)$, there will always be a least upper bound for $(\lhd_1)$ and $(\lhd_2)$ in weak order, denoted $(\lhd_1) \vee (\lhd_2)$. A second goal could be to compute $(\lhd_1) \vee (\lhd_2)$ for any pair of TITOs $(\lhd_1), (\lhd_2)$.

In the case $n = 2$, the weak order on TTot_n looks like this:



We will see how notation like $[\underline{1}][2]$ encodes a TITO during the project. In the case $n = 3$, some TITOs can be visualized using diagrams like the following.





Automating such visualizations could be another direction for the project.

Prerequisites: None. Experience with partial orders, group theory, linear algebra, and coding will be helpful, but isn't required.