ReadMe for the program to Section 6.2

The program `DHO_search` is a simplified but functional version of the program used to show the claim of Section 6.2 of the paper [1]. The code that avoids, to a certain extend, that isomorphic DHO are generated and tested was removed for better readability. As the goal is to show that all generated DHO are of symmetric Type H or Type D, this simplification has no impact on the functionality. The program however is considerably slower (it takes about 15 minutes instead of some seconds in the case $dim(\mathcal{S}) = 9$.)
We will use the notation as in Section 6.2 of the paper [1].

The files `vec2.h`, `vs2.h` and `DHO.h` provide the infrastructure for binary vectors, vector spaces and DHOs as far as needed in `DHO_search.cpp`. The names of the functions are speaking, it might be possible to read the search program without looking in the `.h` files. The file `H3Data.txt` contains the Huybrechts DHO $\mathcal{T}_1$ of rank 3.
The search in `DHO_search.cpp` is launched by `SearchHindSupDHO<4,6,9> ...` for $dim(\mathcal{S}) = 9$ respectively `SearchHindSupDHO<4,6,10> ...` for $dim(\mathcal{S}) = 10$. Only one of these lines should be uncommented in the main program of `DHO_search.cpp`.

The program is written such that `SearchHindSupDHO<r,k,n> ...` can search for any rank-r DHO with $dim(U) = n$ with a give hyperplane induced subDHO with ambient space of dimension k. It consist of several chained recursive search procedures.

Step 1

`rec_search_W()` search for subspaces $W = Z \cap \mathcal{C} \subseteq \langle e_0, \dots e_{k-1}\rangle$ of dimension $r - (n - k)$. For the search for Section 6.2 we will have $W = \emptyset$ for $dim(\mathcal{S}) = 10$ and $W = \langle w \rangle$ in case of $dim(\mathcal{S}) = 9$ as detailed in Section 6.2. $Z_0 = W \oplus \langle e_k, \dots e_{n-2}\rangle$, $Z = Z_0 \oplus \langle e_{n-1}\rangle$. The indices range form 0 to $n - 1$ as in the program.

`rec_search_Pi` searches for the permutations π. The there tested condition, $X_i^\pi + X_j^\pi \notin \langle T_i, T_j\rangle$, $T_i \neq T_j \in \mathcal{T}_1$, is, in the current situation, equivalent to the DHO conditions given in Section 6.2.

Step 2

`genE()` and `rec_search_V()` generates spaces V for $\mathcal{E}_i$. I.e. the spaces V of rank r with $V \curlywedge X_0 = x_i$ such that $dim(V \cap X) = 1$ for all $X \in S_0$. $\mathcal{E}_i$. DHO condition (iii) is automatically fulfilled.

`genS1()` tries to combine the elements of $\mathcal{E}_i$ to the set S_1, such that $X \curlywedge X'$ $X \in S_0 \setminus Z$, $X' \in S_1$ are disjoint, and the elements of S_1 and fulfill conditions (ii) and (iii) in Section 6.2.

References

[1] U. Dempwolff and Y. Edel. The Radical of Binary Dimensional Dual Hyperovals. (`http://www.mathi.uni-heidelberg.de/~yves/Papers/radical.html`).