An Algorithm for Constructing two disjoint Hadamard Designs[1]

H.R. Maimani[ab] and **R. Torabi**[ac]

[a]*Institute for Studies in Theoretical Physics and Mathematics*

(IPM)

P.O.Box 19395-5746, Tehran, IRAN

[b]*Shahid Rajaee University (SRU)*

P.O.Box 16785-163, Tehran, IRAN

[c]*University of Tehran*

P.O.Box 19395-1795, Tehran, IRAN

Abstract

For a given Hadamard design D of order n, we construct another Hadamard design D' of the same order which is disjoint from D.

1. Introduction

Let X be a v-set and let $P_i(X)$ be the set of i-subsets of X. The pair $D = (X, \mathcal{B})$ in which $\mathcal{B}$ is a collection of the elements of $P_k(X)$ (blocks) is called a 2-(v, k, λ) design if every element

[1]This work was supported by IPM.

of $P_2(X)$ appears in $\mathcal{B}$, λ times. D is called a *symmetric design* if $|\mathcal{B}| = v$. For any natural number n, 2-$(4n-1, 2n-1, n-1)$ design is called a *Hadamard design* (H-design) of order n.

Two designs $D_1 = (X, \mathcal{B}_1)$ and $D_2 = (X, \mathcal{B}_2)$ are called disjoint if $\mathcal{B}_1 \cap \mathcal{B}_2 = \emptyset$. An interesting problem in the theory of combinatorial designs which has attracted some attentions in the past is to prove the existence of a design disjoint from a given one. The method of constructing such design could be very interesting as well. For the case $k \geq 3$, and $\lambda = 1$, an algorithm exists in the literature [2,3]. In the following lemma a sufficient condition for the existence of a disjoint design from a given design is presented.

Lemma 1 (Permutation Lemma [1]). Let $D = (X, \mathcal{B})$ be a 2-(v, k, λ) design such that $|\mathcal{B}|^2 < \binom{v}{k}$. Then a disjoint design $D' = (X, \mathcal{B}')$, exists such that $\mathcal{B} \cap \mathcal{B}' = \emptyset$.

We note that a more general feature of this lemma exists in the literature [1]. By this lemma, the existence of two disjoint H-designs is obvious. In what follows, we present an algorithm constructing a disjoint H-design, from a given one.

2. The Algorithm

Let $|X| = 4n - 1$, and $k = 2n - 1$. For any $x \in X$, the function f_x is defined as follows:

$$f_x : P_k(X) \to P_k(X)$$

$$f_x(B) = \begin{cases} B & x \in B \\ \overline{B}\backslash\{x\} & x \notin B, \end{cases}$$

where $\overline{B} = X\backslash B$. If $\mathcal{B}$ is a collection of blocks, then we define

$$f_x(\mathcal{B}) = \{f_x(B) | B \in \mathcal{B}\}.$$

If $\mathcal{B}$ is a collection of blocks and $x, y \in X$, then

$$\begin{aligned} \lambda_{\mathcal{B}}(xy) &= \text{the number of blocks of } \mathcal{B} \text{ containing } \{x, y\}. \\ r_{\mathcal{B}}(x) &= \text{the number of blocks of } \mathcal{B} \text{ containing } \{x\}. \end{aligned}$$

Lemma 2. $D = (X, \mathcal{B})$ be an H-design of order n and $x \in X$. Then $D' = (X, f_x(\mathcal{B}))$ is an H-design of order n.

Proof. Suppose that $\mathcal{B}_1$ is the collection of blocks containing x and $\mathcal{B}_2 = \mathcal{B}\backslash\mathcal{B}_1$. Since $f_x(\mathcal{B}_1) = \mathcal{B}_1$, therefore, $f_x(\mathcal{B}) = \mathcal{B}_1 \cup f_x(\mathcal{B}_2)$. To prove the lemma, it suffices to show that for $y, z \in X$, $\lambda_{f_x(\mathcal{B})}(yz) = n - 1$. For $x \in \{y, z\}$, the blocks containing $\{y, z\}$ belongs to $\mathcal{B}_1$, and clearly $\lambda_{f_x(\mathcal{B})}(yz) = n - 1$.

Now suppose that $x \notin \{y, z\}$, and let $\lambda_{f_x(\mathcal{B}_1)}(yz) = m$. Therefore, $\lambda_{\mathcal{B}_2}(yz) = \lambda - m$. It remains to show that $\lambda_{f_x(\mathcal{B}_2)}(yz) = \lambda - m$. For this, we have

$$r = r_{\mathcal{B}}(y) = r_{\mathcal{B}_1}(y) + r_{\mathcal{B}_2}(y) = \lambda_{\mathcal{B}_1}(xy) + r_{\mathcal{B}_2}(y).$$

Hence, $r_{\mathcal{B}_2}(y) = r - \lambda$. The number of blocks of $\mathcal{B}_2$ containing y or z, is

$$r_{\mathcal{B}_2}(y) + r_{\mathcal{B}_2}(z) - \lambda_{\mathcal{B}_2}(yz) = 2r - 3\lambda + m.$$

Therefore,

$$\begin{aligned}\lambda_{f_x(\mathcal{B}_2)}(yz) &= |\{B \in f_x(\mathcal{B}_2)|y, z \in B\}| \\ &= |\{\overline{B}\backslash\{x\}|B \in \mathcal{B}_2, y, z \in B\}| \\ &= |\{B \in \mathcal{B}_2|y, z \in B\}| \\ &= |\mathcal{B}_2| - 2r - 3\lambda + m = \lambda - m.\square\end{aligned}$$

From here on, we denote $f_x(B)$ by B^x. Also for $x_1, x_2, \cdots, x_m \in X$, we let $(B^{x_1\cdots x_m}) = (B^{x_1\cdots x_{m-1}})^{x_m}$. Note that the order of x_i's in $B^{x_1x_2\cdots x_m}$ is important.

In computing $B^{x_1x_2\cdots x_m}$ the element x_i is called *effective* with respect to $B^{x_1,\cdots x_{i-1}}$ if $x_i \notin B^{x_1x_2\cdots x_{i-1}}$ and otherwise *ineffective.* Clearly if x_i is ineffective then $B^{x_1x_2\cdots x_{i-1}} = B^{x_1\cdots x_{i-1}x_i}$.

Lemma 3. Let $B \in P_k(X)$, and let $x_1, \cdots, x_m$ be distinct elements of X. If all the x_i's are effective with respect to $B^{x_1\cdots x_{i-1}}$, Then for $0 \le l \le [m/2]$. $x_{2l} \in B$ and $x_{2l+1} \notin B$. Also

$$B^{x_1x_2\cdots x_m} = \begin{cases} \overline{B}\Delta\{x_1, x_2, \cdots, x_m\} & \text{for } m \text{ odd,} \\ B\Delta\{x_1, x_2, \cdots, x_m\} & \text{for } m \text{ even,}\end{cases}$$

where Δ denotes symmetric difference of sets.

Proof. By induction on m we prove the lemma.

For $m = 1$, since x_1 is effective, hence $x_1 \in B$. Therefore, by definition,

$$B^{x_1} = \overline{B}\backslash\{x_1\} = \overline{B}\Delta\{x_1\}.$$

If $m = 2$, then $x_1 \notin B$ and $x_2 \notin B^{x_1}$. Therefore $x_2 \in B$. Also

$$
\begin{aligned}
B^{x_1x_2} &= (\overline{B}\backslash\{x_1\})^{x_2}\\
&= (\overline{\overline{B}\backslash\{x_1\}})\backslash x_2\\
&= (B\backslash\{x_2\}) \cup x_1\\
&= B\Delta\{x_1, x_2\}.
\end{aligned}
$$

Now, let $m > 2$. We just work out the proof for even m's and for odd m's the proof is similar. For m even, we have

$$B^{x_1\cdots x_{m-1}} = \overline{B}\Delta\{x_1\cdots x_{m-1}\}.$$

Since x_m is effective,, therefore, $x_m \notin B^{x_1x_2\cdots x_{m-1}}$. From the other hand, $x_m \notin \overline{B}$ (since x_i's are distinct), hence $x_m \in B$ and we have

$$
\begin{aligned}
B^{x_1x_2\cdots x_m} = (B^{x_1\cdots x_{m-1}})^{x_m} &= (\overline{B}\Delta\{x_1, x_2, \cdots, x_{m-1}\})^{x_m}\\
&= (\overline{\overline{B}\Delta\{x_1, \cdots, x_m\}})\backslash\{x_m\}\\
&= B\Delta\{x_1, x_2, \cdots, x_m\}.\square
\end{aligned}
$$

Lemma 4. If $D = (X, \mathcal{B})$ is a symmetric design, then the elements $x_1, x_2, \cdots, x_m \in X$ exist such that $m \leq \lambda + 2$ and for every $B \in \mathcal{B}$, $\{x_1, x_2, \cdots, x_m\} \not\subseteq B$.

Proof. In a symmetric design every two blocks, B_1 and B_2, intersect on λ elements. Suppose that $B_1 \cap B_2 = \{x_1, \cdots, x_\lambda\}$. We choose the elements $x_{\lambda+1} \in B_1\backslash B_2$ and $x_{\lambda+2} \in B_2\backslash B_1$. Clearly $\{x_1, \cdots, x_{\lambda+2}\}$ possesses the desired properties. $\square$

Now the algorithm is defined as follows:

Algorithm *DisjHdesign*

begin

input $D = (X, \mathcal{B})$

select $x_1, \cdots, x_m \in X$ s.t.

$m \leq \lambda + 2$, and for all $B \in \mathcal{B}$; $\{x_1, \cdots, x_m\} \not\subseteq B$

for all $B \in \mathcal{B}$

compute $B^{x_1 \cdots x_m}$

output $D' = (X, \mathcal{B}^{x_1 \cdots x_m})$

end.

Theorem 5. The Algorithm *DisjHdesign* constructs a disjoint H-design from a given H-design of order n with $O(n^3)$ time complexity.

Proof. Let $D = (X, \mathcal{B})$ be an H-design of order n and be input of the algorithm and consider $D' = (X, \mathcal{B}')$ be the H-design obtained from the above algorithm. By Lemma 2, D' clearly is an H-design. Now we prove that $\mathcal{B} \cap \mathcal{B}' = \emptyset$. If $B \in \mathcal{B} \cap \mathcal{B}'$, then a block B_1 exists such that $B = B_1^{x_1 \cdots x_m}$. Now suppose that all the ineffective elements are omitted and $x_{i_1}, \cdots, x_{i_l}$ are effective elements which remain. Therefore, $B = B^{x_{i_1} \cdots x_{i_l}}$. By Lemma 3, $x_{i_2}, x_{i_4}, \cdots$ belong to B_1 and the rest are out of B_1. If l is even, then

$$\begin{aligned} \lambda = |B \cap B_1| &= |B \Delta (\{x_{i_1} \cdots x_{i_l}\} \cap B_1)| = |B_1 \backslash \{x_{i_1} \cdots x_{i_l}\}| \\ &= k - l/2 \geq k - m > \lambda, \end{aligned}$$

which is a contradiction. If l is odd, then

$$\begin{aligned}\lambda = |B \cap B_1| &= |\overline{B}\Delta(\{x_{i_1} \cdots x_{i_l}\} \cap B_1)| \\ &= |B_1 \cap \{x_{i_1} \cdots x_{i_l}\}| \\ &= \tfrac{l-1}{2} < \lambda,\end{aligned}$$

which is a contradiction again. Hence $\mathcal{B} \cap \mathcal{B}' = \emptyset$.

With regard to the proof given in Lemma 4 selecting of the elements $x_1, x_2, \cdots, x_m$ in the algorithm *DisjHdesign* takes $O(n)$. The **for** statement is performed $O(n)$ times, and in each iteration the computation of $\mathcal{B}^{x_1 \cdots x_m}$ is executed in $O(mn)$. Since m is less than $\lambda + 2 = O(n)$, Therefore $O(mn)$ can be considered as $O(n^2)$. Consequently, the total time of the algorithm is $O(n^3)$. □

3. Conclusion

In this paper an algorithm is given to construct a Hadamard design disjoint from a given Hadamard design of the similar order. The time complexity of the algorithm is $O(n^3)$.

It is still desirable to find an algorithm to give all disjoint Hadamard design from a given one. It is also interesting to find an algorithm that can be applied on other design. These algorithms are left as open problems.

Acknowledgments

The authors would like to thank the anonymous referee for making helpful suggestions.

References

[1] B. Ganter, J. Pelikan, and L. Teirlinck, *Small sprawling systems of equicardinal sets*, Ars Combin. **4**(1977), 133-142.

[2] L. Teirlinck, *On making two Steiner triple systems disjoint*, J. Combin. Theory, Ser. A **23**(1977), 349-350.

[3] A. P. Street, and D.J. Street, *Combinatorics of Experimental Designs*, Clarendon Press, Oxford, 1987.