MATH97440 — Formalising Maths

Coursework 3

[redacted]ic.ac.uk

I affirm that this represents my own work, completed independently, without assistance except where acknowledged and properly referenced.

1 Introduction

For my final project, I decided to explore an area of lean that was not so well-developed. As I have also been specialising into combinatorics for quite some time now, I thought it would be a good challenge to define and create some basic API for Hadamard matrices. Some of the generalisations of Hadamard matrices that are seen in the project come from my M4R project.

Although in theory, if I had proven all the theorems I stated and completed all the definnitions I started, I would have required many different areas of mathlib such as the API for linear algebra, quadratic residues, and finite fields, I unfortunately did not get the time to explore those areas of Lean in great depth.*

The content in this pdf will be presented in the same order as I have preesnted it in the code. The "sections" of the code (demarcated by decorated block comments) also correspond to the sections of this pdf.

2 Definitions

A Hadamard matrix can be presented as an object in linear algebra: a square matrix of the greatest determinant whose entries have absolute value no greater than 1. However, if we look at it from a combinatorial perspective, we can see that the information contained is that "each two rows agree on exactly half their entries". This gives rise to a generalisation which I have not found any prominent literature on: for a group G (written additively), a *cyclic Hadamard matrix*† over G of order n and height h is a $h \times n$ matrix, where each element of G appears an equal number of times in the element-wise difference between any two rows. To better manage these objects in Lean, I have opted to

Cyctors A *cyctor* is essentially a finite direct product of a finite group G, so called because initially I had considered cyctors only over the rings $\mathbb{Z}/n\mathbb{Z}$. Operations are mostly defined on cyctors pointwise, as expected, except that in my studies I have also found it useful to define two different products when the underlying algebraic structure is a ring: in addition to the element-wise product, there is also a "Kronecker-style" product. This distinction is the same as the distinction between `list.prod` and `list.product`. I have not included the former definition in the Lean code, however.

*Unfortunately this seems to always happen: I spend about 30 hours on this project and most of the time was spent getting to grips with new data structures (`lists` and `vectors` for this project) and API for no-so-interesting objects (`fin n` and `zmod n` for this project.

†I am aware that this is a misnomer, more on that in the next paragraph.

Cyctor sum We also have the *sum* of a cyctor, which is a map $C \mapsto \sum_{c \in C} c$. This can be used to prove several things, particularly in the general case of cyclic Hadamard matrices over general groups G, but notably it can also be used to prove that Hadamard matrices of singly-even order cannot exist. (More on this later.)

`has_add` and other instances... When I looked at mathlib, it seems that wheenver a new structure is defined, and there is some notion of operations on them siimilar to $+$, $-$, $\times$, etc., then these are labelled as `instance`s of `has_add`, `has_sub`, etc. I did not have time to figure out exactly what this mechanism did and how it worked, but I labelled my new definitions with them, as it seems would be expected of any code in mathlib.

Vector product and natural coercions Next we define the vector product (used in the definition of the Kronecker product of two cyclic Hadamard matrices), and coercions from `cyctor` to `vector`, and from `vector` to `list`.

Orthogonality Two cyctors can be orthogonal, and this basically means they can coexist in the same cyclic Hadamard matrix. Orthogonality is symmetric, but it is not reflexive or transitive. The fact that it is symmetric is proved later on.

Hadamard structures and existence definitions Finally, we wrap up with the definition of a cyclic Hadamard matrix as a collection of h cyctors of length n over an additive group G. Hadamard matrices are cyclic Hadamard matrices over $G = \mathbb{Z}/2\mathbb{Z}$ where $h = n$. The existence of such matrices are defined in `cyc_hadamard_exists G n h` and `hadamard_exists n`. (For example, the former says that there exists an $h \times n$ cyclic Hadamard matrix over the group G.) I think ideally, I would define a Hadamard matrix as an instance of a cyclic Hadamard matrix, but I did not have time to figure out how this would work.

Notes Throughout this process, I was made aware that if I was to count the number of elements equal to $g \in G$ in a certain cyctor, then "equality in G had to be decidable". I still do not fully understand what this means, but it makes some sort of sense that there has to be some way to compmute whether or not two expressions are equivalent in finite time.

3 Basic properties

Here we prove two things: that the transpose of a Hadamard matrix is a Hadamard matrix, and that the orthogonality relation is symmetric. The proof of the latter is mathematically trivial, but the former is quite challenging. I unfortunately did not get to prove this in Lean, (it would have likely been the longest proof by far) but I present the proof here.

Theorem 1. *The transposition of a Hadamard matrix is a Hadamard matrix.*

Proof. We prove this result by showing that any two columns of a Hadamard matrix H agree on exactly half their entries. WLOG, suppose that the first and second columns agree on k entries, where $k > \frac{n}{2}$. (A very similar argument works if they agree on less than half their entries.) We can also assume WLOG that the entries they agree on are all 1's, as we can invert the rows where they agreed on -1's.

Now, forming a new $k \times n$ matrix with the the k rows above, we look at the other $n-2$ entries in each row. Of these entries in any two rows, $\frac{n}{2}-2$ must agree and $\frac{n}{2}$ must disagree. Hence, counting these agreements by pairs of rows, we obtain excatly

$$\binom{k}{2}\left(\frac{n}{2}-2\right) = \frac{k(k-1)(n-4)}{4} \tag{1}$$

agreements in total, whereas if we count these agreements by columns, then the total number of agreements is at least the number of columns multiplied by the minimum number of agreements per column, which is

$$(n-2)\left[\binom{\lceil k/2\rceil}{2} + \binom{\lfloor k/2\rfloor}{2}\right] \quad = \quad \frac{\lceil k/2\rceil^2 + \lfloor k/2\rfloor^2 - k}{2}(n-2).$$

Now, if k is even, this reduces to $k(k-2)(n-2)/4$, and since $2k > n$ we have

$$\begin{aligned}\frac{k(k-2)(n-2)}{4} &= \frac{k(kn-2k-2n+4)}{4}\\ &> \frac{k(kn-4k-n+4)}{4}\\ &= \frac{k(k-1)(n-4)}{4},\end{aligned}$$

a contradiction. Similarly, if k is odd, then equation (1) reduces to $(k^2-k+1)(n-2)/4$ and we have

$$\begin{aligned}\frac{(k^2-k+1)(n-2)}{4} &> \frac{(k^2-k+1)(n-2)+2k(1-k)-n+2}{4} \qquad \text{(when } n>2\text{)}\\ &= \frac{k(k-1)(n-4)}{4},\end{aligned}$$

which is again a contradiction. Note that the assumption $n > 2$ is always true, since otherwise k, being odd and greater than $\frac{n}{2}$, would be greater than n itself. □

Hence it seems quite evident that this proof supersedes all the other proofs I did manage to get through in complexity, and by quite a lot!

4 Existence

The first method of generating Hadamard matrices was actually using a kind of Kronecker product: if you had Hadamard matrices of orders n and m, then by taking their "Kronecker product" (in a suitable sense), you were able to obtain a Hadamard matrix of order nm. This construction can be generalised (and has been generalised in the Lean code) to the construction of cyclic Hadamard matrices in general: for two cyclic Hadamard matrices over a group G of sizes $n \times h_n$ and $m \times h_m$, we their kronecker product would be a matrix over G of size $nm \times h_n h_m$, and its rows would also be pairwise orthogonal. Here, I present the proof for just Hadamard matrices — the proof for general cyclic Hadamard matrices is very similar.

Theorem 2. *If H and G are Hadamard matrices, then so is $H \otimes G$, where $\otimes$ denotes the Kronecker product.*

Proof. Let A_i denote the i-th row of a matrix A, and let $H \otimes G = F$. If H has order n and G has order m, then

$$F = \begin{pmatrix} h_{11}G & \cdots & h_{1n}G \\ \vdots & \ddots & \vdots \\ h_{n1}G & \cdots & h_{nn}G \end{pmatrix}$$

has order nm.

We compare F_i and F_j for $i \neq j$. Note that, if $i \not\equiv j \pmod{m}$, then since $I = G_{i \bmod m}$ and $J = G_{j \bmod m}$ are orthogonal, so are $-I$ and J; I and $-J$; and $-I$ and $-J$. Ultimately, this means that F_i and F_j are orthogonal, regardless of the values in H.

On the other hand, if $i \equiv j \pmod{m}$, then since the relevant two rows of H, H_x and H_y, agree on exactly half their entries, $F_i = H_x \otimes G_{i \bmod m}$ and $F_j = H_y \otimes G_{i \bmod m}$ will also agree on exactly half their entries, as desired. □

This theorem, despite its power, is not very useful in isolation. For example, using the Hadamard matrix $\left(\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}\right)$, we can generate Hadamard matrices of order 2^n for all positive integers n, which I have done here in Lean (this was a srprisingly easy induction in Lean, compared to some of the other proofs), but it doesn't provide an easy way to generate Hadamard matrices of any other orders. What we really need is a way to construct Hadamard matrices from scratch. This is what the Paley construction (later) will provide.

5 Non-existence

It is known that Hadamardmatrices can only exist if they are of certain orders:

Theorem 3. *A Hadamard matrix of order n can exist only if $n = 1$, 2, or a multiple of 4.*

Proof. Clearly, if n is odd and greater than 1, then no two rows can agree on half their entries. If n is even and twice an odd number, then any two distinct rows agree on an odd number of entries, and hence their sums differ by twice an odd number. However, if there are more than 2 rows, then the sums of the first two rows differ by twice an odd number, as do the sums of the second two rows. But then the sums of the first and third rows must differ by twice an even number, a contradiction. □

At the outset, it might seem like this result is the more difficult part of establishing for which orders there exists a Hadamard matrix. In fact, the opposite is true, by a large margin.

Conjecture 4 (Hadamard)**.** *There exists a Hadamard matrix of order n for $n = 1$, 2, and all multiples of 4.*

In Lean, I managed to prove that a Hadamard matrix did not exist when n is odd. This fact is trivial to see mathematically, but in Lean it was quite an undertaking. I was preparing to do prove the other case (when $n \equiv 2 \pmod{4}$), via the sum and differences proof (as alluded to in section 1) but it would probably have been the longest proof in the project, had it been completed.

6 The Paley construction

Finally, we present the Paley construction. I did not get to spend much time on this in Lean, and I will keep this section brief. However, I thought I would add this for completeness. As an example, for p prime and $p \equiv 3 \pmod 4$, we can construct a Hadamard matrix as follows:

Theorem 5. *There exists a Hadamard matrix of order $4k$ whenever $4k-1$ is prime, or a prime power.*

Proof. By taking the Hadamard matrix to standard form, negating, and then removing the first row and column, it suffices to show that there exists a square matrix of order $4k-1$ with $2k$ copies of 1 and $2k-1$ copies of -1 on each row, such that any two rows agree on exactly $2k-1$ entries. We will construct such a matrix using the theory of quadratic residues.

In particular, let $q = 4k-1$ be a prime power p^n, where n is a positive integer. Then let $Q(x)$ be an irreducible polynomial of degree n over $\mathbb{Z}/p\mathbb{Z}$. We now work with variables in the quotient $R = (\mathbb{Z}/p\mathbb{Z}[x])/\langle Q(x)\rangle$.

Then we construct the square matrix M indexed by R with 1 in the position (i,j) if $i+j \equiv a^2 \pmod{Q(x)}$ for some $a \in R$, and -1 otherwise. Note that since just over half of the residues modulo $Q(x)$ are quadratic residues, there will be exactly $2k$ 1's in each row. Furthermore, given two rows $i_1 \neq i_2$, the quantity of j's where (i_1, j) and (i_2, j) are both 1's can be determined by noting that this is equivalent to finding a and b such that

$$\left.\begin{aligned} i_1 + j &\equiv a^2 \pmod{Q(x)} \\ i_2 + j &\equiv b^2 \pmod{Q(x)} \end{aligned}\right\} \implies i_1 - i_2 \equiv (a-b)(a+b) \pmod{Q(x)},$$

where we can then reconstruct j as $a^2 - i_1 \pmod{Q(x)}$ or $b^2 - i_2 \pmod{Q(x)}$. Since $i_1 - i_2$ is fixed, and $Q(x)$ is irreducible, for each $(a-b)$ there is exactly one $(a+b)$ which works, from which we can the obtain the unique a and b. □