Final Project Part 1: Exploring the properties of 2-by-2 matrices

Oishik Ganguly

December 6, 2017

Contents

1 Introduction

Matrices are well understood mathematical structures as a consequence of the wide applicability of linear algebra to fields ranging from computer science to economics. Basic properties and operations relating to matrices are thus familiar even to high school students. In the first part of this final project, we consider a specific kind of matrix, namely, the 2-by-2 matrix. In particular, we formalize their structure as an inductive type in coq, formalize the operations of matrix addition, matrix multiplication, and matrix transposition for 2-by-2 matrices, and finally, prove certain properties about general 2-by-2 matrices and specific 2-by-2 matrices. In this report, we will not dwell on the formalization of the above mentioned operations: they are unremarkable and may be viewed in the coq project file. Instead, we will jump straight into a discussion of the properties we proved. Finally, note that henceforth, when we refer to a property of or operation on a matrix, we are referring to a 2-by-2 matrix in particular.

2 A note on matrix representation, operations, and paraphernalia

Given the inductive type defintion `m22` for the 2-by-2 matrices, we decided that a matrix of the form :

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

would be represented as an element of `m22` as follows : `M22 a11 a12 a21 a21`. The definitions `m22_addition` , `m22_multiplication`, `m22_exponentiation`, `m22_exponentiation_altern` `m22_transpose` follow from this representation of the 2-by-2 matrix. We also defined a notion of matrix equality (`A =m= B` in terms of notation) by an element-wise comparison of the two matrices concerned.

3 Associativity of matrix multiplication

This property was captured in coq as follows:

```
Proposition m22_multiplication_is_associative :
forall (matrix_a matrix_b matrix_c : m22),
  m22_multiplication matrix_a (m22_multiplication matrix_b matrix_c) =
  m22_multiplication (m22_multiplication matrix_a matrix_b) matrix_c.
```

The proof for this property was straightforward, and mostly challenging because of the algebraic manipulations required. We inducted over the structures of the 3 matrices, thus following the usual pen-and-paper approach to the proof of writing out the elements of the matrices as variables. Finally, we used coq's `do` construct to repeat a set of algebraic simplifications which gives us the required equality.

4 Neutrality of the identity element

The neutrality of the identity element, captured in mathematical notation as $M \times I = I \times M = M$, was represented in coq via 2 lemmas, `m22_mult_identity_r` and `m22_mult_identity_l`. The proofs for these lemmas were direct, and involved inducting over the structure of the given matrix.

5 Taking the n-th power of the M1 matrix

We defined the following 2-by-2 matrix :

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

as the matrix `M1` in coq, and then proved the following property about it :

```
Proposition about_M1 :
forall (n : nat),
  m22_exponentiation M1  n = M22 1 n 0 1.
```

We proved this by induction over `n`. The proof itself is standard, and goes through using the unfold lemmas for the exponentiation operation. We employed a similar method for the next property, which is exactly the same as above except that it uses the alternative definition of matrix exponentiation.

6 Exponentiation is commutative

This proposition captures the following property of exponentiation of matrices :

```
Proposition m22_exponentiation_is_commutative :
forall (matrix_a : m22) (n : nat),
  m22_multiplication (matrix_a) (m22_exponentiation matrix_a n) =
  m22_multiplication (m22_exponentiation matrix_a n) (matrix_a).
```

The proof for this proceeded by induction as well. We initially attempted to prove the property by further inducting on the structure of `matrix_a`. However, following the handwritten proof for the property, we could make use of the neutrality of the identity element and associativity of matrix multiplication properties that we proved earlier to see this proof through with minimal effort. Likewise, using these tools, we saw the proof through for the next property, involving `m22_exponentiation_alternative`, without a hitch.

7 The equivalence of the two definitions of exponentiation

The equivalence of the two definitions of matrix exponentiation, `m22_exponentiation` and `m22_exponentiation_alternative`, was proved using induction. Using the previous property concerning the commutativity of exponentiation, this proof followed through.

8 Taking the n-th power of the M2 matrix

The `M2` matrix was defined as the following matrix :

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

The proof for this, as with the proof for the property concerning `M1`, uses induction, and follows through directly. The only effort required is with the algebraic manipulation at the end of the proof.

9 Transposition is involutive

The proof for this property directly follows by inducting over the structure of a matrix and unfolding `m22_transpose` (since the operation is *not* recursively defined).

10 Transpose of the product of matrices

Mathematically, this property is represented is as follows :

$$(M_1 \times M_2)^T = M_2^T \times M_1^T$$

.

The proof involved using the inductive structure of the two matrices, and then unfolding `m22_transpose`. At this point, the equality was clear, and some algebraic manipulation, namely, using the commutativity of multiplication of natural numbers, needed to be used. This took up the bulk of the proof.

11 Transposition and exponentiation commute

The interchangeability of the exponent and the transposition symbol for matrices is captured in coq as follows :

```
Proposition m22_transposition_and_exponentiation_commute :
  forall (matrix_a : m22) (n : nat),
    m22_transpose (m22_exponentiation matrix_a n) =
    m22_exponentiation (m22_transpose matrix_a) n.
```

The proof this uses induction on `n`. For the inductive case, we used the transposition of products property that we just proved. This brought us to the following stage :

```
m22_multiplication (m22_transpose matrix_a) (m22_exponentiation (m22_transpose matrix_a) n') =
   m22_exponentiation (m22_transpose matrix_a) (S n')
```

We realized that the left hand side corresponded to unfolding the alternative definition of exponentiation on `S n`. We thus used the equivalence of two definitions of exponentiation, refolded the left hand side, and finally used the equivalence on the right hand side to get an equality.

12 Proving the n-th power of M2 property without induction

If the coq representation does not immediately make obvious the fact that `M1` is the transpose of `M2`, the mathematical representation does :

$M2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^T = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = M1$. We use this simple fact and `M1`'s property to prove this proposition.

13 The special property of F

Matrix `F` was defined as follows:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Playing around with the powers of `F`, we found that it displayed properties similar to a Fibonacci sequence. We observed that $\forall n \geq 0, F^n$ returns a matrix with the n-th element of the fibonacci sequence in row 1 column 2. Furthermore, we found that $\forall n, F^{n+2} = F^{n+1} + F^n$. We prove this in `Proposition about_F`, represented as follows:

```
Proposition about_F :
  forall (n : nat),
    m22_addition (m22_exponentiation F n) (m22_exponentiation F (S n)) =
    m22_exponentiation F (S (S n)).
```

To prove this statement, we made use of our own induction principle, namely, `nat_ind2`. Proving the base cases involved a great deal of algebraic manipulation, as evidenced by the multiple use of the `do` tactic.

For the inductive case, our strategy was as follows: to show that $F^{n+4} = F^{n+3} + F^{n+2}$, we factorized the right hand side to obtain the form $(F^{n+1} + F^{n+2}) \cdot F$, and then rewrote the parenthesized expression with the second induction hypotheses (namely, that $F^{n+1} + F^{n+2} = F^{n+3}$). To enable this factorization, we also defined our distribution of matrix multiplication lemmas before the proof for `about_F`.

Finally, we proved that the element in the first row and second column of F^n was the n-th element of the fibonacci sequence. The proposition for this theorem was as follows :

```
Lemma the_fibonacci_character_of_F :
  forall (n : nat),
    (match m22_exponentiation F n with
    | M22 _ a12 _ _ =>
      a12
     end) = fib_v0 n.
```

where fib_v0 is a non-accumulator implementation of the Fibonacci function. The proof for this proceeded by our own induction principle. For the inductive case, we used about_F to represent the expression being matched in the left hand side of the goal as a sum. We then represented F^{Sn} and $F^{S(Sn)}$ in terms of the inductive structures of a matrix, and the equality we required followed after using the inductive hypotheses.

14 Conclusion

In the first section of this final project, we explored a coq based formalization of 2-by-2 matrices, and proved properties related to the operations of addition, multiplication, exponentiation, and transposition. We also looked at the properties of specific matrices, namely, M1, M2, and F. Over the course of these proofs, we gained a stronger understanding of inductive structures, inductive proofs, and how to make coq proofs correspond to hand written proofs.