A quick algorithm for identifying conjugate groups in GF(2)

Peter M. Maurer
Department of Computer Science
Baylor University
Waco, Texas
Peter_Maurer@Baylor.edu

1 Abstract

This report gives the details an algorithm for determining whether two matrix groups are conjugate to one another. Each group is designated by a pair of matrices that generate the group. The algorithm is able to determine whether the two generated groups are conjugate to one another without actually generating the groups.

2 Introduction

The primary question we are trying to answer is how many isomorphic copies of S_5 exist in the set of all non-singular 5×5 matrices over GF(2). ($GF(2)$ is the integers modulo 2.) Especially we want to know how many conjugacy classes there are in the set of such groups. (At this point I suspect the answer is either 1 or 4, but later with that.)

Given two pairs of generators (M_1, M_2) and (N_1, N_2), we want to know if the generated groups are conjugate. (These are 5×5 matrices in our current application. We could use this algorithm for other sizes of matrices, and I've attempted to keep this discussion as generic as possible, but the 5×5 thing is going to creep in here and there.)

Note that this algorithm is motivated by the older algorithm which would take 20 years or so to run to completion for 5×5 matrices. The older algorithm enumerates the generators. All order 2's are placed in one set all order 5's in another. Choose one matrix from each set. Test all pairs. The test generates the group. If the group size exceeds 120 (we are looking for isomorphic copies of S_5) we abort and discard the generators. Otherwise, we generate all conjugates and compare against saved groups we've already found. If there is no match we publish the generator pair and save the group for other comparisons. Otherwise we discard it. The main difficulty is that there are $\prod_{i=0}^{n-1}(2^n - 2^i)$ nonsingular $n\times n$ matrices. For $n=5$ this is 9,999,360 potential conjugate matrices. There are 666,624 order 5 matrices and 6,975 order 2 matrices. We can cut the number of pairs by eliminating all powers of matrices from the order 5's. Given two matrices A and B, with A order 5 and B order 2, A^2, A^3, and A^4 are also order 5 matrices, and will generate the same group with B as does A. After eliminating all higher powers we end up with 166,656 order 5 matrices.

The idea that we need an order 5 and an order 2 matrix to generate a copy of S_5 comes from the well known fact that the permutations (1,2,3,4) and (1,2,3,4,5), in cycle notation, generate S_5. (1,2,3,4,5)(1,5)=(1,2,3,4), so (1,2,3,4,5) and (1,5) generate S_5 as well. Since (1,2,3,4,5)=(2,3,4,5,1)=(3,4,5,1,2)=(4,5,1,2,3)=(5,1,2,3,4), we can replace (1,5) with (1,2), (2,3), (3,4), or (4,5). (Problem: given an order 2 matrix, assume it is the matrix equivalent of one of these 5. Find the matrices for the other 4.)

3 Foundations of the Algorithm

To attempt to answer the conjugacy question more quickly, we ask ourselves under what circumstances is there a non-singular matrix A such that $M_1A = AN_1$ and $M_2A = AN_2$. The generated groups can still be conjugate, even if such an A does not exist (I think). The conjugacy matrix does not have to map the generators precisely. However, if the generators are conjugate in this fashion, then the generated groups are definitely conjugate. This technique may generate some false positives for non-conjugate groups, but these can be resolved using brute force techniques.

To begin with, we ask ourselves under what circumstances is there a matrix A such that

$$MA = AN \tag{1}$$

for some pair of matrices (M, N). (Note that the zero matrix is always a trivial solution to equation (1). We insist that the matrix A be non-singular.)

Let

$$M = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} & m_{1,5} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} & m_{2,5} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} & m_{3,5} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} & m_{4,5} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} & m_{5,5} \end{pmatrix}$$

$$N = \begin{pmatrix} n_{1,1} & n_{1,2} & n_{1,3} & n_{1,4} & n_{1,5} \\ n_{2,1} & n_{2,2} & n_{2,3} & n_{2,4} & n_{2,5} \\ n_{3,1} & n_{3,2} & n_{3,3} & n_{3,4} & n_{3,5} \\ n_{4,1} & n_{4,2} & n_{4,3} & n_{4,4} & n_{4,5} \\ n_{5,1} & n_{5,2} & n_{5,3} & n_{5,4} & n_{5,5} \end{pmatrix}$$

and

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} & a_{2,5} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} & a_{3,5} \\ a_{4,1} & a_{4,2} & a_{4,3} & a_{4,4} & a_{4,5} \\ a_{5,1} & a_{5,2} & a_{5,3} & a_{5,4} & a_{5,5} \end{pmatrix}$$

Multiplying out according to equation (1) gives us 25 equations:

$$m_{1,1}a_{1,1} + m_{1,2}a_{2,1} + m_{1,3}a_{3,1} + m_{1,4}a_{4,1} + m_{1,5}a_{5,1} = a_{1,1}n_{1,1} + a_{1,2}n_{2,1} + a_{1,3}n_{3,1} + a_{1,4}n_{4,1} + a_{1,5}n_{5,1}$$
$$m_{1,1}a_{1,2} + m_{1,2}a_{2,2} + m_{1,3}a_{3,2} + m_{1,4}a_{4,2} + m_{1,5}a_{5,2} = a_{1,1}n_{1,2} + a_{1,2}n_{2,2} + a_{1,3}n_{3,2} + a_{1,4}n_{4,2} + a_{1,5}n_{5,2}$$
$$m_{1,1}a_{1,3} + m_{1,2}a_{2,3} + m_{1,3}a_{3,3} + m_{1,4}a_{4,3} + m_{1,5}a_{5,3} = a_{1,1}n_{1,3} + a_{1,2}n_{2,3} + a_{1,3}n_{3,3} + a_{1,4}n_{4,3} + a_{1,5}n_{5,3}$$
$$m_{1,1}a_{1,4} + m_{1,2}a_{2,4} + m_{1,3}a_{3,4} + m_{1,4}a_{4,4} + m_{1,5}a_{5,4} = a_{1,1}n_{1,4} + a_{1,2}n_{2,4} + a_{1,3}n_{3,4} + a_{1,4}n_{4,4} + a_{1,5}n_{5,4}$$
$$m_{1,1}a_{1,5} + m_{1,2}a_{2,5} + m_{1,3}a_{3,5} + m_{1,4}a_{4,5} + m_{1,5}a_{5,5} = a_{1,1}n_{1,5} + a_{1,2}n_{2,5} + a_{1,3}n_{3,5} + a_{1,4}n_{4,5} + a_{1,5}n_{5,5}$$
$$m_{2,1}a_{1,1} + m_{2,2}a_{2,1} + m_{2,3}a_{3,1} + m_{2,4}a_{4,1} + m_{2,5}a_{5,1} = a_{2,1}n_{1,1} + a_{2,2}n_{2,1} + a_{2,3}n_{3,1} + a_{2,4}n_{4,1} + a_{2,5}n_{5,1}$$
$$m_{2,1}a_{1,2} + m_{2,2}a_{2,2} + m_{2,3}a_{3,2} + m_{2,4}a_{4,2} + m_{2,5}a_{5,2} = a_{2,1}n_{1,2} + a_{2,2}n_{2,2} + a_{2,3}n_{3,2} + a_{2,4}n_{4,2} + a_{2,5}n_{5,2}$$
$$m_{2,1}a_{1,3} + m_{2,2}a_{2,3} + m_{2,3}a_{3,3} + m_{2,4}a_{4,3} + m_{2,5}a_{5,3} = a_{2,1}n_{1,3} + a_{2,2}n_{2,3} + a_{1,3}n_{3,3} + a_{2,4}n_{4,3} + a_{2,5}n_{5,3}$$
$$m_{2,1}a_{1,4} + m_{2,2}a_{2,4} + m_{2,3}a_{3,4} + m_{2,4}a_{4,4} + m_{2,5}a_{5,4} = a_{2,1}n_{1,4} + a_{2,2}n_{2,4} + a_{2,3}n_{3,4} + a_{2,4}n_{4,4} + a_{2,5}n_{5,4}$$
$$m_{2,1}a_{1,5} + m_{2,2}a_{2,5} + m_{2,3}a_{3,5} + m_{2,4}a_{4,5} + m_{2,5}a_{5,5} = a_{2,1}n_{1,5} + a_{2,2}n_{2,5} + a_{2,3}n_{3,5} + a_{2,4}n_{4,5} + a_{2,5}n_{5,5}$$
$$m_{3,1}a_{1,1} + m_{3,2}a_{2,1} + m_{3,3}a_{3,1} + m_{3,4}a_{4,1} + m_{3,5}a_{5,1} = a_{3,1}n_{1,1} + a_{3,2}n_{2,1} + a_{1,3}n_{3,1} + a_{3,4}n_{4,1} + a_{3,5}n_{5,1}$$
$$m_{3,1}a_{1,2} + m_{3,2}a_{2,2} + m_{3,3}a_{3,2} + m_{3,4}a_{4,2} + m_{3,5}a_{5,2} = a_{3,1}n_{1,2} + a_{3,2}n_{2,2} + a_{1,3}n_{3,2} + a_{3,4}n_{4,2} + a_{3,5}n_{5,2}$$
$$m_{3,1}a_{1,3} + m_{3,2}a_{2,3} + m_{3,3}a_{3,3} + m_{3,4}a_{4,3} + m_{3,5}a_{5,3} = a_{3,1}n_{1,3} + a_{3,2}n_{2,3} + a_{1,3}n_{3,3} + a_{3,4}n_{4,3} + a_{3,5}n_{5,3}$$
$$m_{3,1}a_{1,4} + m_{3,2}a_{2,4} + m_{3,3}a_{3,4} + m_{3,4}a_{4,4} + m_{3,5}a_{5,4} = a_{3,1}n_{1,4} + a_{3,2}n_{2,4} + a_{1,3}n_{3,4} + a_{3,4}n_{4,4} + a_{3,5}n_{5,4}$$
$$m_{3,1}a_{1,5} + m_{3,2}a_{2,5} + m_{3,3}a_{3,5} + m_{3,4}a_{4,5} + m_{3,5}a_{5,5} = a_{3,1}n_{1,5} + a_{3,2}n_{2,5} + a_{1,3}n_{3,5} + a_{3,4}n_{4,5} + a_{3,5}n_{5,5}$$
$$m_{4,1}a_{1,1} + m_{4,2}a_{2,1} + m_{4,3}a_{3,1} + m_{4,4}a_{4,1} + m_{4,5}a_{5,1} = a_{4,1}n_{1,1} + a_{4,2}n_{2,1} + a_{4,3}n_{3,1} + a_{4,4}n_{4,1} + a_{4,5}n_{5,1}$$
$$m_{4,1}a_{1,2} + m_{4,2}a_{2,2} + m_{4,3}a_{3,2} + m_{4,4}a_{4,2} + m_{4,5}a_{5,2} = a_{4,1}n_{1,2} + a_{4,2}n_{2,2} + a_{4,3}n_{3,2} + a_{4,4}n_{4,2} + a_{4,5}n_{5,2}$$
$$m_{4,1}a_{1,3} + m_{4,2}a_{2,3} + m_{4,3}a_{3,3} + m_{4,4}a_{4,3} + m_{4,5}a_{5,3} = a_{4,1}n_{1,3} + a_{4,2}n_{2,3} + a_{4,3}n_{3,3} + a_{4,4}n_{4,3} + a_{4,5}n_{5,3}$$
$$m_{4,1}a_{1,4} + m_{4,2}a_{2,4} + m_{4,3}a_{3,4} + m_{4,4}a_{4,4} + m_{4,5}a_{5,4} = a_{4,1}n_{1,4} + a_{4,2}n_{2,4} + a_{4,3}n_{3,4} + a_{4,4}n_{4,4} + a_{4,5}n_{5,4}$$
$$m_{4,1}a_{1,5} + m_{4,2}a_{2,5} + m_{4,3}a_{3,5} + m_{4,4}a_{4,5} + m_{4,5}a_{5,5} = a_{4,1}n_{1,5} + a_{4,2}n_{2,5} + a_{4,3}n_{3,5} + a_{4,4}n_{4,5} + a_{4,5}n_{5,5}$$
$$m_{5,1}a_{1,1} + m_{5,2}a_{2,1} + m_{5,3}a_{3,1} + m_{5,4}a_{4,1} + m_{5,5}a_{5,1} = a_{5,1}n_{1,1} + a_{5,2}n_{2,1} + a_{5,3}n_{3,1} + a_{5,4}n_{4,1} + a_{5,5}n_{5,1}$$
$$m_{5,1}a_{1,2} + m_{5,2}a_{2,2} + m_{5,3}a_{3,2} + m_{5,4}a_{4,2} + m_{5,5}a_{5,2} = a_{5,1}n_{1,2} + a_{5,2}n_{2,2} + a_{5,3}n_{3,2} + a_{5,4}n_{4,2} + a_{5,5}n_{5,2}$$
$$m_{5,1}a_{1,3} + m_{5,2}a_{2,3} + m_{5,3}a_{3,3} + m_{5,4}a_{4,3} + m_{5,5}a_{5,3} = a_{5,1}n_{1,3} + a_{5,2}n_{2,3} + a_{5,3}n_{3,3} + a_{5,4}n_{4,3} + a_{5,5}n_{5,3}$$
$$m_{5,1}a_{1,4} + m_{5,2}a_{2,4} + m_{5,3}a_{3,4} + m_{5,4}a_{4,4} + m_{5,5}a_{5,4} = a_{5,1}n_{1,4} + a_{5,2}n_{2,4} + a_{5,3}n_{3,4} + a_{5,4}n_{4,4} + a_{5,5}n_{5,4}$$
$$m_{5,1}a_{1,5} + m_{5,2}a_{2,5} + m_{5,3}a_{3,5} + m_{5,4}a_{4,5} + m_{5,5}a_{5,5} = a_{5,1}n_{1,5} + a_{5,2}n_{2,5} + a_{5,3}n_{3,5} + a_{5,4}n_{4,5} + a_{5,5}n_{5,5}$$

Recalling that these matrices are over $GF(2)$, addition is the same as subtraction, and these equations can be rewritten as follows.

$$m_{1,1}a_{1,1} + m_{1,2}a_{2,1} + m_{1,3}a_{3,1} + m_{1,4}a_{4,1} + m_{1,5}a_{5,1} + a_{1,1}n_{1,1} + a_{1,2}n_{2,1} + a_{1,3}n_{3,1} + a_{1,4}n_{4,1} + a_{1,5}n_{5,1} = 0$$
$$m_{1,1}a_{1,2} + m_{1,2}a_{2,2} + m_{1,3}a_{3,2} + m_{1,4}a_{4,2} + m_{1,5}a_{5,2} + a_{1,1}n_{1,2} + a_{1,2}n_{2,2} + a_{1,3}n_{3,2} + a_{1,4}n_{4,2} + a_{1,5}n_{5,2} = 0$$
$$m_{1,1}a_{1,3} + m_{1,2}a_{2,3} + m_{1,3}a_{3,3} + m_{1,4}a_{4,3} + m_{1,5}a_{5,3} + a_{1,1}n_{1,3} + a_{1,2}n_{2,3} + a_{1,3}n_{3,3} + a_{1,4}n_{4,3} + a_{1,5}n_{5,3} = 0$$
$$m_{1,1}a_{1,4} + m_{1,2}a_{2,4} + m_{1,3}a_{3,4} + m_{1,4}a_{4,4} + m_{1,5}a_{5,4} + a_{1,1}n_{1,4} + a_{1,2}n_{2,4} + a_{1,3}n_{3,4} + a_{1,4}n_{4,4} + a_{1,5}n_{5,4} = 0$$
$$m_{1,1}a_{1,5} + m_{1,2}a_{2,5} + m_{1,3}a_{3,5} + m_{1,4}a_{4,5} + m_{1,5}a_{5,5} + a_{1,1}n_{1,5} + a_{1,2}n_{2,5} + a_{1,3}n_{3,5} + a_{1,4}n_{4,5} + a_{1,5}n_{5,5} = 0$$
$$m_{2,1}a_{1,1} + m_{2,2}a_{2,1} + m_{2,3}a_{3,1} + m_{2,4}a_{4,1} + m_{2,5}a_{5,1} + a_{2,1}n_{1,1} + a_{2,2}n_{2,1} + a_{2,3}n_{3,1} + a_{2,4}n_{4,1} + a_{2,5}n_{5,1} = 0$$
$$m_{2,1}a_{1,2} + m_{2,2}a_{2,2} + m_{2,3}a_{3,2} + m_{2,4}a_{4,2} + m_{2,5}a_{5,2} + a_{2,1}n_{1,2} + a_{2,2}n_{2,2} + a_{2,3}n_{3,2} + a_{2,4}n_{4,2} + a_{2,5}n_{5,2} = 0$$
$$m_{2,1}a_{1,3} + m_{2,2}a_{2,3} + m_{2,3}a_{3,3} + m_{2,4}a_{4,3} + m_{2,5}a_{5,3} + a_{2,1}n_{1,3} + a_{2,2}n_{2,3} + a_{1,3}n_{3,3} + a_{2,4}n_{4,3} + a_{2,5}n_{5,3} = 0$$
$$m_{2,1}a_{1,4} + m_{2,2}a_{2,4} + m_{2,3}a_{3,4} + m_{2,4}a_{4,4} + m_{2,5}a_{5,4} + a_{2,1}n_{1,4} + a_{2,2}n_{2,4} + a_{2,3}n_{3,4} + a_{2,4}n_{4,4} + a_{2,5}n_{5,4} = 0$$
$$m_{2,1}a_{1,5} + m_{2,2}a_{2,5} + m_{2,3}a_{3,5} + m_{2,4}a_{4,5} + m_{2,5}a_{5,5} + a_{2,1}n_{1,5} + a_{2,2}n_{2,5} + a_{2,3}n_{3,5} + a_{2,4}n_{4,5} + a_{2,5}n_{5,5} = 0$$
$$m_{3,1}a_{1,1} + m_{3,2}a_{2,1} + m_{3,3}a_{3,1} + m_{3,4}a_{4,1} + m_{3,5}a_{5,1} + a_{3,1}n_{1,1} + a_{3,2}n_{2,1} + a_{1,3}n_{3,1} + a_{3,4}n_{4,1} + a_{3,5}n_{5,1} = 0$$
$$m_{3,1}a_{1,2} + m_{3,2}a_{2,2} + m_{3,3}a_{3,2} + m_{3,4}a_{4,2} + m_{3,5}a_{5,2} + a_{3,1}n_{1,2} + a_{3,2}n_{2,2} + a_{1,3}n_{3,2} + a_{3,4}n_{4,2} + a_{3,5}n_{5,2} = 0$$
$$m_{3,1}a_{1,3} + m_{3,2}a_{2,3} + m_{3,3}a_{3,3} + m_{3,4}a_{4,3} + m_{3,5}a_{5,3} + a_{3,1}n_{1,3} + a_{3,2}n_{2,3} + a_{1,3}n_{3,3} + a_{3,4}n_{4,3} + a_{3,5}n_{5,3} = 0$$
$$m_{3,1}a_{1,4} + m_{3,2}a_{2,4} + m_{3,3}a_{3,4} + m_{3,4}a_{4,4} + m_{3,5}a_{5,4} + a_{3,1}n_{1,4} + a_{3,2}n_{2,4} + a_{1,3}n_{3,4} + a_{3,4}n_{4,4} + a_{3,5}n_{5,4} = 0$$
$$m_{3,1}a_{1,5} + m_{3,2}a_{2,5} + m_{3,3}a_{3,5} + m_{3,4}a_{4,5} + m_{3,5}a_{5,5} + a_{3,1}n_{1,5} + a_{3,2}n_{2,5} + a_{1,3}n_{3,5} + a_{3,4}n_{4,5} + a_{3,5}n_{5,5} = 0$$
$$m_{4,1}a_{1,1} + m_{4,2}a_{2,1} + m_{4,3}a_{3,1} + m_{4,4}a_{4,1} + m_{4,5}a_{5,1} + a_{4,1}n_{1,1} + a_{4,2}n_{2,1} + a_{4,3}n_{3,1} + a_{4,4}n_{4,1} + a_{4,5}n_{5,1} = 0$$
$$m_{4,1}a_{1,2} + m_{4,2}a_{2,2} + m_{4,3}a_{3,2} + m_{4,4}a_{4,2} + m_{4,5}a_{5,2} + a_{4,1}n_{1,2} + a_{4,2}n_{2,2} + a_{4,3}n_{3,2} + a_{4,4}n_{4,2} + a_{4,5}n_{5,2} = 0$$
$$m_{4,1}a_{1,3} + m_{4,2}a_{2,3} + m_{4,3}a_{3,3} + m_{4,4}a_{4,3} + m_{4,5}a_{5,3} + a_{4,1}n_{1,3} + a_{4,2}n_{2,3} + a_{4,3}n_{3,3} + a_{4,4}n_{4,3} + a_{4,5}n_{5,3} = 0$$
$$m_{4,1}a_{1,4} + m_{4,2}a_{2,4} + m_{4,3}a_{3,4} + m_{4,4}a_{4,4} + m_{4,5}a_{5,4} + a_{4,1}n_{1,4} + a_{4,2}n_{2,4} + a_{4,3}n_{3,4} + a_{4,4}n_{4,4} + a_{4,5}n_{5,4} = 0$$
$$m_{4,1}a_{1,5} + m_{4,2}a_{2,5} + m_{4,3}a_{3,5} + m_{4,4}a_{4,5} + m_{4,5}a_{5,5} + a_{4,1}n_{1,5} + a_{4,2}n_{2,5} + a_{4,3}n_{3,5} + a_{4,4}n_{4,5} + a_{4,5}n_{5,5} = 0$$
$$m_{5,1}a_{1,1} + m_{5,2}a_{2,1} + m_{5,3}a_{3,1} + m_{5,4}a_{4,1} + m_{5,5}a_{5,1} + a_{5,1}n_{1,1} + a_{5,2}n_{2,1} + a_{5,3}n_{3,1} + a_{5,4}n_{4,1} + a_{5,5}n_{5,1} = 0$$
$$m_{5,1}a_{1,2} + m_{5,2}a_{2,2} + m_{5,3}a_{3,2} + m_{5,4}a_{4,2} + m_{5,5}a_{5,2} + a_{5,1}n_{1,2} + a_{5,2}n_{2,2} + a_{5,3}n_{3,2} + a_{5,4}n_{4,2} + a_{5,5}n_{5,2} = 0$$
$$m_{5,1}a_{1,3} + m_{5,2}a_{2,3} + m_{5,3}a_{3,3} + m_{5,4}a_{4,3} + m_{5,5}a_{5,3} + a_{5,1}n_{1,3} + a_{5,2}n_{2,3} + a_{5,3}n_{3,3} + a_{5,4}n_{4,3} + a_{5,5}n_{5,3} = 0$$
$$m_{5,1}a_{1,4} + m_{5,2}a_{2,4} + m_{5,3}a_{3,4} + m_{5,4}a_{4,4} + m_{5,5}a_{5,4} + a_{5,1}n_{1,4} + a_{5,2}n_{2,4} + a_{5,3}n_{3,4} + a_{5,4}n_{4,4} + a_{5,5}n_{5,4} = 0$$
$$m_{5,1}a_{1,5} + m_{5,2}a_{2,5} + m_{5,3}a_{3,5} + m_{5,4}a_{4,5} + m_{5,5}a_{5,5} + a_{5,1}n_{1,5} + a_{5,2}n_{2,5} + a_{5,3}n_{3,5} + a_{5,4}n_{4,5} + a_{5,5}n_{5,5} = 0$$

We can now collect a few terms.

$$(m_{1,1} + n_{1,1})a_{1,1} + n_{2,1}a_{1,2} + n_{3,1}a_{1,3} + n_{4,1}a_{1,4} + n_{5,1}a_{1,5} + m_{1,2}a_{2,1} + m_{1,3}a_{3,1} + m_{1,4}a_{4,1} + m_{1,5}a_{5,1} = 0$$
$$n_{1,2}a_{1,1} + (m_{1,1} + n_{2,2})a_{1,2} + n_{3,2}a_{1,3} + n_{4,2}a_{1,4} + n_{5,2}a_{1,5} + m_{1,2}a_{2,2} + m_{1,3}a_{3,2} + m_{1,4}a_{4,2} + m_{1,5}a_{5,2} = 0$$
$$n_{1,3}a_{1,1} + n_{2,3}a_{1,2} + (m_{1,1} + n_{3,3})a_{1,3} + n_{4,3}a_{1,4} + n_{5,3}a_{1,5} + m_{1,2}a_{2,3} + m_{1,3}a_{3,3} + m_{1,4}a_{4,3} + m_{1,5}a_{5,3} = 0$$
$$n_{1,4}a_{1,1} + n_{2,4}a_{1,2} + n_{3,4}a_{1,3} + (m_{1,1} + n_{4,4})a_{1,4} + n_{5,4}a_{1,5} + m_{1,2}a_{2,4} + m_{1,3}a_{3,4} + m_{1,4}a_{4,4} + m_{1,5}a_{5,4} = 0$$

$$n_{1,5}a_{1,1} + n_{2,5}a_{1,2} + n_{3,5}a_{1,3} + n_{4,5}a_{1,4} + (m_{1,1} + n_{5,5})a_{1,5} + m_{1,2}a_{2,5} + m_{1,3}a_{3,5} + m_{1,4}a_{4,5} + m_{1,5}a_{5,5} = 0$$
$$m_{2,1}a_{1,1} + (m_{2,2} + n_{1,1})a_{2,1} + n_{2,1}a_{2,2} + n_{3,1}a_{2,3} + n_{4,1}a_{2,4} + n_{5,1}a_{2,5} + m_{2,3}a_{3,1} + m_{2,4}a_{4,1} + m_{2,5}a_{5,1} = 0$$
$$m_{2,1}a_{1,2} + n_{1,2}a_{2,1} + (m_{2,2} + n_{2,2})a_{2,2} + n_{3,2}a_{2,3} + n_{4,2}a_{2,4} + n_{5,2}a_{2,5} + m_{2,3}a_{3,2} + m_{2,4}a_{4,2} + m_{2,5}a_{5,2} = 0$$
$$m_{2,1}a_{1,3} + n_{1,3}a_{2,1} + n_{2,3}a_{2,2} + (m_{2,2} + n_{3,3})a_{2,3} + n_{4,3}a_{2,4} + n_{5,3}a_{2,5} + m_{2,3}a_{3,3} + m_{2,4}a_{4,3} + m_{2,5}a_{5,3} = 0$$
$$m_{2,1}a_{1,4} + n_{1,4}a_{2,1} + n_{2,4}a_{2,2} + n_{3,4}a_{2,3} + (m_{2,2} + n_{4,4})a_{2,4} + n_{5,4}a_{2,5} + m_{2,3}a_{3,4} + m_{2,4}a_{4,4} + m_{2,5}a_{5,4} = 0$$
$$m_{2,1}a_{1,5} + n_{1,5}a_{2,1} + n_{2,5}a_{2,2} + n_{3,5}a_{2,3} + n_{4,5}a_{2,4} + (m_{2,2} + n_{5,5})a_{2,5} + m_{2,3}a_{3,5} + m_{2,4}a_{4,5} + m_{2,5}a_{5,5} = 0$$
$$m_{3,1}a_{1,1} + m_{3,2}a_{2,1} + (m_{3,3} + n_{1,1})a_{3,1} + n_{2,1}a_{3,2} + n_{3,1}a_{1,3} + n_{4,1}a_{3,4} + n_{5,1}a_{3,5} + m_{3,4}a_{4,1} + m_{3,5}a_{5,1} = 0$$
$$m_{3,1}a_{1,2} + m_{3,2}a_{2,2} + n_{1,2}a_{3,1} + (m_{3,3} + n_{2,2})a_{3,2} + n_{3,2}a_{1,3} + n_{4,2}a_{3,4} + n_{5,2}a_{3,5} + m_{3,4}a_{4,2} + m_{3,5}a_{5,2} = 0$$
$$m_{3,1}a_{1,3} + m_{3,2}a_{2,3} + n_{1,3}a_{3,1} + n_{2,3}a_{3,2} + (m_{3,3} + n_{3,3})a_{3,3} + n_{4,3}a_{3,4} + n_{5,3}a_{3,5} + m_{3,4}a_{4,3} + m_{3,5}a_{5,3} = 0$$
$$m_{3,1}a_{1,4} + m_{3,2}a_{2,4} + n_{1,4}a_{3,1} + n_{2,4}a_{3,2} + n_{3,4}a_{1,3} + (m_{3,3} + n_{4,4})a_{3,4} + n_{5,4}a_{3,5} + m_{3,4}a_{4,4} + m_{3,5}a_{5,4} = 0$$
$$m_{3,1}a_{1,5} + m_{3,2}a_{2,5} + n_{1,5}a_{3,1} + n_{2,5}a_{3,2} + n_{3,5}a_{1,3} + n_{4,5}a_{3,4} + (m_{3,3} + n_{5,5})a_{3,5} + m_{3,4}a_{4,5} + m_{3,5}a_{5,5} = 0$$
$$m_{4,1}a_{1,1} + m_{4,2}a_{2,1} + m_{4,3}a_{3,1} + (m_{4,4} + n_{1,1})a_{4,1} + n_{2,1}a_{4,2} + n_{3,1}a_{4,3} + n_{4,1}a_{4,4} + n_{5,1}a_{4,5} + m_{4,5}a_{5,1} = 0$$
$$m_{4,1}a_{1,2} + m_{4,2}a_{2,2} + m_{4,3}a_{3,2} + n_{1,2}a_{4,1} + (m_{4,4} + n_{2,2})a_{4,2} + n_{3,2}a_{4,3} + n_{4,2}a_{4,4} + n_{5,2}a_{4,5} + m_{4,5}a_{5,2} = 0$$
$$m_{4,1}a_{1,3} + m_{4,2}a_{2,3} + m_{4,3}a_{3,3} + n_{1,3}a_{4,1} + n_{2,3}a_{4,2} + (m_{4,4} + n_{3,3})a_{4,3} + n_{4,3}a_{4,4} + n_{5,3}a_{4,5} + m_{4,5}a_{5,3} = 0$$
$$m_{4,1}a_{1,4} + m_{4,2}a_{2,4} + m_{4,3}a_{3,4} + n_{1,4}a_{4,1} + n_{2,4}a_{4,2} + n_{3,4}a_{4,3} + (m_{4,4} + n_{4,4})a_{4,4} + n_{5,4}a_{4,5} + m_{4,5}a_{5,4} = 0$$
$$m_{4,1}a_{1,5} + m_{4,2}a_{2,5} + m_{4,3}a_{3,5} + n_{1,5}a_{4,1} + n_{2,5}a_{4,2} + n_{3,5}a_{4,3} + n_{4,5}a_{4,4} + (m_{4,4} + n_{5,5})a_{4,5} + m_{4,5}a_{5,5} = 0$$
$$m_{5,1}a_{1,1} + m_{5,2}a_{2,1} + m_{5,3}a_{3,1} + m_{5,4}a_{4,1} + (m_{5,5} + n_{1,1})a_{5,1} + n_{2,1}a_{5,2} + n_{3,1}a_{5,3} + n_{4,1}a_{5,4} + n_{5,1}a_{5,5} = 0$$
$$m_{5,1}a_{1,2} + m_{5,2}a_{2,2} + m_{5,3}a_{3,2} + m_{5,4}a_{4,2} + n_{1,2}a_{5,1} + (m_{5,5} + n_{2,2})a_{5,2} + n_{3,2}a_{5,3} + n_{4,2}a_{5,4} + n_{5,2}a_{5,5} = 0$$
$$m_{5,1}a_{1,3} + m_{5,2}a_{2,3} + m_{5,3}a_{3,3} + m_{5,4}a_{4,3} + n_{1,3}a_{5,1} + n_{2,3}a_{5,2} + (m_{5,5} + n_{3,3})a_{5,3} + n_{4,3}a_{5,4} + n_{5,3}a_{5,5} = 0$$
$$m_{5,1}a_{1,4} + m_{5,2}a_{2,4} + m_{5,3}a_{3,4} + m_{5,4}a_{4,4} + n_{1,4}a_{5,1} + n_{2,4}a_{5,2} + n_{3,4}a_{5,3} + (m_{5,5} + n_{4,4})a_{5,4} + n_{5,4}a_{5,5} = 0$$
$$m_{5,1}a_{1,5} + m_{5,2}a_{2,5} + m_{5,3}a_{3,5} + m_{5,4}a_{4,5} + n_{1,5}a_{5,1} + n_{2,5}a_{5,2} + n_{3,5}a_{5,3} + n_{4,5}a_{5,4} + (m_{5,5} + n_{5,5})a_{5,5} = 0$$

Recall that M and N are known, while A is not. Thus we have 25 equations in 25 unknowns. We order the variables by row, $a_{1,1}, a_{1,2}, \cdots, a_{1,5}, a_{2,1}, \cdots, a_{5,5}$, and put the equations into a 25×25 matrix. Given the two matrices M and N, we can construct the 25×25 matrix using the following procedure. Let N^T be the transpose of N, and let $M_{i,j}$ be the 5×5 matrix created from the element $m_{i,j}$ of M as follows.

$$\begin{pmatrix} m_{i,j} & 0 & 0 & 0 & 0 \\ 0 & m_{i,j} & 0 & 0 & 0 \\ 0 & 0 & m_{i,j} & 0 & 0 \\ 0 & 0 & 0 & m_{i,j} & 0 \\ 0 & 0 & 0 & 0 & m_{i,j} \end{pmatrix}$$

We can assemble the 25×25 matrix out of 5×5 matrices by first constructing the following two matrices. (Note that each entry in the following matrices is itself a 5×5 matrix, giving us a 25×25 in each case.)

$$Z_1 = \begin{pmatrix} N^T & 0 & 0 & 0 & 0 \\ 0 & N^T & 0 & 0 & 0 \\ 0 & 0 & N^T & 0 & 0 \\ 0 & 0 & 0 & N^T & 0 \\ 0 & 0 & 0 & 0 & N^T \end{pmatrix} \qquad Z_2 = \begin{pmatrix} M_{1,1} & M_{1,2} & M_{1,3} & M_{1,4} & M_{1,5} \\ M_{2,1} & M_{2,2} & M_{2,3} & M_{2,4} & M_{2,5} \\ M_{3,1} & M_{3,2} & M_{3,3} & M_{3,4} & M_{3,5} \\ M_{4,1} & M_{4,2} & M_{4,3} & M_{4,4} & M_{4,5} \\ M_{5,1} & M_{5,2} & M_{5,3} & M_{5,4} & M_{5,5} \end{pmatrix}$$

The required matrix is $Z_1 + Z_2$. This procedure is (more or less) obvious from the equations given above.

Since the constant terms are all zero, we know that $A = 0$ is a solution to the 25×25 matrix. The question we must ask is whether there is more than one solution. Gaussian elimination, in its usual form, is useful for determining whether there is a single unique solution to a system of equations. We must adapt it to determine whether there are multiple solutions. Here is how this is done.

First we use Gaussian elimination to put our 25×25 matrix into row-echelon form. In row-echelon form the matrix must meet the following criteria.

1. Each row consists of a number of zeros followed by zero or more non-zero elements. The position of the first non-zero element of a row is the *lead* position. The lead position of row i is designated as $\text{lead}(i)$. By convention, the lead position of a zero row is $c+1$, where c is the number of columns in the matrix.
2. Suppose $i > j > 0$ and suppose both row i and row j have non-zero elements. Then the lead position of row i must be greater than the lead position of row j.
3. The lead position of a non-zero row must contain a 1.
4. The matrix may have one or more rows of all zeros, but these must appear at the end of the matrix. No non-zero row may follow a zero row.
5. All entries below the lead position of a row must be zero (by 2 above). Similarly, all the positions above the lead position of a row must be zero. (This last condition is not always specified as a condition for row-echelon form, but it is required by our algorithms.)

The following is an example of an 8×8 matrix in row echelon form. Row-echelon matrices need not be square.

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{2}$$

The lead positions of rows 1-8 of matrix (2) are 1, 4, 5, 6, 8, 9, 9, and 9. The positions above and below and to the left of a lead position must be zero, but there is no restriction on other positions.

To transform a matrix into row-echelon form, we perform Gaussian elimination as usual, however, when we fail to find a pivot point for a column we move forward to the next column of the same row and continue the elimination process from that point. Note that failure to find a pivot point in a column implies that the entry in that column is zero and all entries below it are zero. Reduction of pivot points to 1 is done immediately (This is a no-op in GF(2).) Back substitution is done immediately after reduction of the pivot point to 1. (This can be done without multiplication or division in GF(2).) Gaussian elimination stops when all columns have been exhausted. Unlike conventional Gaussian elimination, the matrix does not need to be square.

In matrix (2), we would fail to find pivot points at row 2 column 2, row 2 column 3 and at row 5 column 7. We also note that in ordinary Gaussian elimination we could do the reduction to 1 and back-substitution immediately. It's not normally done that way because the algorithm is cleaner to save back-substitution until the end. However in our case, finding the lead positions to do the back substitution is fairly complicated, so we do it right away, when we have the lead position automatically identified.

In the general case, there can be three outcomes of the row echelon reduction. First if there are no rows of all zeros, and the matrix is square, there is a single solution to the system of equations. If there are no rows of all zeros and the matrix is not square, then there are multiple solutions. If there are rows of all zeros, there may be one solution, multiple solutions or there may be no solution. For there to be any solution the constant term of all zero rows must be zero. In our case this will always be true since all constant terms are zero to begin with.

To determine the number of solutions it is necessary to compute the *differential* of all non-zero rows. If there are multiple solutions, the solutions form a *k*-dimensional vector space. If the original matrix was square, k is the number of rows of all zeros in the row-echelon matrix. If $\text{lead}(i)$ is the lead position of row i, then the differential of row i is given by $\text{diff}(i) = \text{lead}(i+1) - \text{lead}(i) - 1$. Recall that if row i is all zeros then $\text{lead}(i) = c + 1$ where c is the number of columns in the matrix. By convention, the differential of a row of all zeros is zero.

The differential of a row-echelon matrix is the sum of the differentials of all non-zero rows. The differential of the row-echelon matrix is the dimension of the solution space. If

the differential of the matrix is zero, then there is a single solution. Otherwise we use the rows of non-zero differential to compute a set of basis vectors for the solution space.

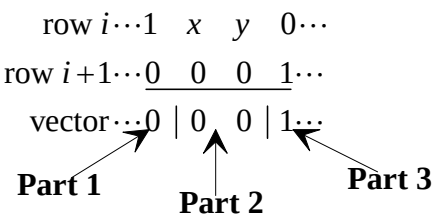

Figure 1. The Regions of basis vectors.

Each basis vector will have a number of elements equal to the number of columns in the matrix and is divided into three regions, which are illustrated in Figure 1. Part 1 is positions 1 through $lead(i)$, part 2 is positions $lead(i)+1$ through $lead(i+1)-1$, and part 3 is positions $lead(i)$ through c, where c is the number of columns in the matrix. Part 3 is filled with zeros. Part 2 will have exactly one position equal to 1 and zeros elsewhere. The first vector will have 1 in position $lead(i)+1$, the second vector will have a 1 in position $lead(i)+2$ and so forth. The values of part 1 are determined by the rows preceding row i. Suppose we are creating the k^{th} vector with a 1 in position $lead(i)+k$. Now consider row $j<i$. If position $lead(i)+k$ of row j contains the value x, and $lead(j)=p$, then position p of the k^{th} vector must contain $-x$.

To return to the original problem, we have two pairs of matrices, (M_1, M_2) and (N_1, N_2), and we want to know if there is a matrix A such that $M_1A = AN_1$ and $M_2A = AN_2$. We use the above procedure to solve two 25×25 matrices and obtain two sets of basis vectors, $B=\{b_1, b_2, \cdots, b_n\}$ and $C=\{c_1, \cdots, c_m\}$, with n and m elements respectively. We want to know if there is some vector in the intersection of B and C. Even more, if V_B is the vector space generated by B and V_C is the vector space generated by C, we want to compute a basis for the vector space $V_B \bigcap V_C$. If a vector $v \in V_B \bigcap V_C$ then there are linear combinations of the B and C vectors such that $v = \delta_1 b_1 + \delta_2 b_2 + \cdots + \delta_n b_n = \gamma_1 c_1 + \gamma_2 c_2 + \cdots + \gamma_m c_m$. Since we want a basis for all such v, we want to generate a basis for all solutions $((\delta_1, \delta_2, \cdots \delta_n, \gamma_1, \gamma_2, \cdots, \gamma_m)$ of the equation $\delta_1 b_1 + \delta_2 b_2 + \cdots + \delta_n b_n = \gamma_1 c_1 + \gamma_2 c_2 + \cdots + \gamma_m c_m$. Alternatively, this can be written $\delta_1 b_1 + \delta_2 b_2 + \cdots + \delta_n b_n + \gamma_1 c_1 + \gamma_2 c_2 + \cdots + \gamma_m c_m = 0$. Each b_i and c_j is a 25 element vector. Let $b_{i,j}$ and $c_{i,j}$ denote the j^{th} position of vectors b_i and c_i respectively. The equation given above reduces to the following 25 equations.

$$\delta_1 b_{1,1} + \delta_2 b_{2,1} + \cdots + \delta_n b_{n,1} + \gamma_1 c_{1,1} + \gamma_2 c_{2,1} + \cdots + \gamma_m c_{m,1} = 0$$
$$\delta_1 b_{1,2} + \delta_2 b_{2,2} + \cdots + \delta_n b_{n,2} + \gamma_1 c_{1,2} + \gamma_2 c_{2,2} + \cdots + \gamma_m c_{m,2} = 0$$
$$\delta_1 b_{1,3} + \delta_2 b_{2,3} + \cdots + \delta_n b_{n,3} + \gamma_1 c_{1,3} + \gamma_2 c_{2,3} + \cdots + \gamma_m c_{m,3} = 0$$
$$\delta_1 b_{1,4} + \delta_2 b_{2,4} + \cdots + \delta_n b_{n,4} + \gamma_1 c_{1,4} + \gamma_2 c_{2,4} + \cdots + \gamma_m c_{m,4} = 0$$

$$\delta_1 b_{1,5} + \delta_2 b_{2,5} + \cdots + \delta_n b_{n,5} + \gamma_1 c_{1,5} + \gamma_2 c_{2,5} + \cdots + \gamma_m c_{m,5} = 0$$
$$\delta_1 b_{1,6} + \delta_2 b_{2,6} + \cdots + \delta_n b_{n,6} + \gamma_1 c_{1,6} + \gamma_2 c_{2,6} + \cdots + \gamma_m c_{m,6} = 0$$
$$\delta_1 b_{1,7} + \delta_2 b_{2,7} + \cdots + \delta_n b_{n,7} + \gamma_1 c_{1,7} + \gamma_2 c_{2,7} + \cdots + \gamma_m c_{m,7} = 0$$
$$\delta_1 b_{1,8} + \delta_2 b_{2,8} + \cdots + \delta_n b_{n,8} + \gamma_1 c_{1,8} + \gamma_2 c_{2,8} + \cdots + \gamma_m c_{m,8} = 0$$
$$\delta_1 b_{1,9} + \delta_2 b_{2,9} + \cdots + \delta_n b_{n,9} + \gamma_1 c_{1,9} + \gamma_2 c_{2,9} + \cdots + \gamma_m c_{m,9} = 0$$
$$\delta_1 b_{1,10} + \delta_2 b_{2,10} + \cdots + \delta_n b_{n,10} + \gamma_1 c_{1,10} + \gamma_2 c_{2,10} + \cdots + \gamma_m c_{m,10} = 0$$
$$\delta_1 b_{1,11} + \delta_2 b_{2,11} + \cdots + \delta_n b_{n,11} + \gamma_1 c_{1,11} + \gamma_2 c_{2,11} + \cdots + \gamma_m c_{m,11} = 0$$
$$\delta_1 b_{1,12} + \delta_2 b_{2,12} + \cdots + \delta_n b_{n,12} + \gamma_1 c_{1,12} + \gamma_2 c_{2,12} + \cdots + \gamma_m c_{m,12} = 0$$
$$\delta_1 b_{1,13} + \delta_2 b_{2,13} + \cdots + \delta_n b_{n,13} + \gamma_1 c_{1,13} + \gamma_2 c_{2,13} + \cdots + \gamma_m c_{m,13} = 0$$
$$\delta_1 b_{1,14} + \delta_2 b_{2,14} + \cdots + \delta_n b_{n,14} + \gamma_1 c_{1,14} + \gamma_2 c_{2,14} + \cdots + \gamma_m c_{m,14} = 0$$
$$\delta_1 b_{1,15} + \delta_2 b_{2,15} + \cdots + \delta_n b_{n,15} + \gamma_1 c_{1,15} + \gamma_2 c_{2,15} + \cdots + \gamma_m c_{m,15} = 0$$
$$\delta_1 b_{1,16} + \delta_2 b_{2,16} + \cdots + \delta_n b_{n,16} + \gamma_1 c_{1,16} + \gamma_2 c_{2,16} + \cdots + \gamma_m c_{m,16} = 0$$
$$\delta_1 b_{1,17} + \delta_2 b_{2,17} + \cdots + \delta_n b_{n,17} + \gamma_1 c_{1,17} + \gamma_2 c_{2,17} + \cdots + \gamma_m c_{m,17} = 0$$
$$\delta_1 b_{1,18} + \delta_2 b_{2,18} + \cdots + \delta_n b_{n,18} + \gamma_1 c_{1,18} + \gamma_2 c_{2,18} + \cdots + \gamma_m c_{m,18} = 0$$
$$\delta_1 b_{1,19} + \delta_2 b_{2,19} + \cdots + \delta_n b_{n,19} + \gamma_1 c_{1,19} + \gamma_2 c_{2,19} + \cdots + \gamma_m c_{m,19} = 0$$
$$\delta_1 b_{1,20} + \delta_2 b_{2,20} + \cdots + \delta_n b_{n,20} + \gamma_1 c_{1,20} + \gamma_2 c_{2,20} + \cdots + \gamma_m c_{m,20} = 0$$
$$\delta_1 b_{1,21} + \delta_2 b_{2,21} + \cdots + \delta_n b_{n,21} + \gamma_1 c_{1,21} + \gamma_2 c_{2,21} + \cdots + \gamma_m c_{m,21} = 0$$
$$\delta_1 b_{1,22} + \delta_2 b_{2,22} + \cdots + \delta_n b_{n,22} + \gamma_1 c_{1,22} + \gamma_2 c_{2,22} + \cdots + \gamma_m c_{m,22} = 0$$
$$\delta_1 b_{1,23} + \delta_2 b_{2,23} + \cdots + \delta_n b_{n,23} + \gamma_1 c_{1,23} + \gamma_2 c_{2,23} + \cdots + \gamma_m c_{m,23} = 0$$
$$\delta_1 b_{1,24} + \delta_2 b_{2,24} + \cdots + \delta_n b_{n,24} + \gamma_1 c_{1,24} + \gamma_2 c_{2,24} + \cdots + \gamma_m c_{m,24} = 0$$
$$\delta_1 b_{1,25} + \delta_2 b_{2,25} + \cdots + \delta_n b_{n,25} + \gamma_1 c_{1,25} + \gamma_2 c_{2,25} + \cdots + \gamma_m c_{m,25} = 0$$

This system of 25 equations in $m+n$ unknowns can be solved using the procedure given above. We concatenate the vector sets B and C to create a $25\times(m+n)$ matrix. We then transpose the matrix, reduce it to row-echelon form and then find the set of basis vectors for the solution.

This will tell us whether there is a matrix satisfying the original conditions, but it will not tell us whether there is a *non-singular* matrix satisfying the conditions. To determine this we generate all matrices in the intersection and test them for singularity. If we find a non-singular matrix we stop.

If we have *k* basis vectors there are 2^k linear combinations of these vectors. We generate a linear combination of these vectors, which gives us a selection vector that can be used to create a linear combination of the B (or C) vectors. This linear combination is actually a matrix in the intersection which can be tested for singularity.