ON THE THEORY OF NON-MATROIDS

HENRY CRAPO

1. Unmatroids

I have always tried to avoid having to work with double negations, contrapositives, and the like. I had one such discouraging moment was when I was unable to follow what Tom Brylawski was trying to explain to me about tangential k-blocks, – at about the time that Geoff took up that precisely that subject as a central theme of his doctoral thesis on the critical problem for matroids. But I can't resist taking a close look at a problem in the theory of *non-matroids.* Say a *struct* $\mathbf{N}$ of *rank* r on a set E is a subset N of the anti-chain $\binom{E}{r}$ of r-element subsets of the set E. For lack of a better term, call the elements of E *points.* and the subsets in N the *bases* of $\mathbf{N}$. The struct $\mathbf{N}$ has the *transpose property* if, for every ordered multi-set $(A, B, \ldots, D)$ of r bases (not necessarily distinct) of $\mathbf{N}$, there is a way of placing permutations of these sets $A, B, \ldots, D$ as the rows of an $r \times r$ matrix, in such a way that its columns are also bases of $\mathbf{N}$. Clearly, the objective is to prove (or disprove) that every struct in which the transpose property fails is a non-matroid, or equivalently, that there is some pair S, T, of (distinct) bases of $\mathbf{N}$ such that the transpose property fails for the multi-set $(S, S, \ldots, S, T)$. Nothing definitive, but I'd like to talk a bit about what one encounters along the way.

2. So far, so good

There has been a great deal of fine work published on this problem, beginning with Rosa Huang and Gian Carlo Rota's collaboration in 1993, followed by Wendy Chan's charming 4-page proof for rank 3 matroids, and culminating in Jim Geelen and Peter Humphries beautiful 4-page proof for paving matroids of *all* ranks in 2006. Applications in statistics (Latin squares) and invariant theory (straightening coefficients) abound.

3. Python

Let $\mathbf{N}$ be a struct of rank r, as defined above. When we begin to study $r \times r$ matrices of points of $\mathbf{N}$, and their possible transposes, we are bound to make tons of frustrating clerical errors. We have to decide whether to continue like that, or write a *Python* program. I invariably take the latter option these days, since I sometimes have trouble finding the hat that is already on my head. The downside is that one then makes tons of programming errors. But the upside is that programming errors are often relatively easy to correct in *Python.* After about a week's work, we arrive at a stage where we can produce tons of reliable data: for each struct, a list of all the matrices whose row-bases have no transpose, and for those matrices that have

Date: December 16, 2015.

transposes, a list of all the transposes. The next problem is to develop the patience to study all the results.

Confidence reigns supreme! The first surprise is that, even when $\mathbf{N}$ is not a matroid, there is often a veritable flood of transposes. "All you need is one!", as the song goes. But how to find an algorithmic path toward *one* especially nice solution that is *sure* to exist *if* $\mathbf{N}$ is a matroid *and if* Rota's basis conjecture is valid?

4. The window-shade method

Let's assume for a moment that $\mathbf{N}$ *is* a matroid. Then for any two bases A, B of $\mathbf{N}$, there is a bijective exchange function $f : A \to B$ such that $A \backslash a + f(a)$ is a basis, for all $a \in A$.

The *Python* program, which produces all transposes that exist, for any given multi-set of r bases, works row by row, from top to bottom and from left to right. The first row is easy: stick with alphabetic order. If we were to find a transpose with another order in the first row, just reorder the columns, and we find a transpose with alphabetic order in the first row.

The second row is also easy. Say the first two rows are bases A and B. Let $f : A \to B$ be a bijective exchange function. Keep $A = (a, b, c, \ldots, y, z)$ in alphabetic order, and write B in the cyclically right-shifted order $f(z), f(a), f(b), \ldots, f(y)$ of the list of corresponding exchange values $f(a), \ldots, f(z)$. With the second row displayed in this fashion, we have r bases, one for each place in the first row, extending down one space in that column, and to the right (cyclically modulo r) $r - 1$ places. See the third line in the next figure.

It is vaguely conceivable that this *window-shade* approach, successively replacing *tail* elements in the first row, for each of the r bases so far obtained by *head* elements chosen *bijectively* from the basis in the next row. In Figure 1, we indicate the exchanges necessary at each stage by colored dots in the middle of the corresponding squares. (The use of colors in the final matrix at the bottom is inconsistent, but is meant simply to convey the idea that the columns are bases.)

The final result is a set of r sequences of length $2r - 1$ of points, each extending up one column, turning right, and making a complete cycle of the r elements in row 1. In each of these sequences, every connected segment of length r is a basis. We call this a *window-shade solution.*

There is a problem with this approach. How to construct such a solution, using only known exchange properties for bases in a matroid? Look at the middle row of the figure. Call the basis in the third row C. We would presumably obtain the required order on row three by taking a bijective exchange g from the "red" basis in row 1 to the "blue" basis C in row three, arranging the exchange values in the order indicated by the dots in the squares. But we would only be sure that sets formed from $r - 1$ elements $A \backslash b$ of A and one element of $g(b) \in C$ are bases. We would *not* know that the positions marked in red in the fourth row of Figure 1 are bases.

Instead, we might concentrate on the r 'intermediate' bases obtained in the first stage of the process, and use bijective exchange from those bases toward the basis C. We know we can find points in C to replace the *tails* of the r intermediate bases, but we no longer know that the overall replacement is bijective.

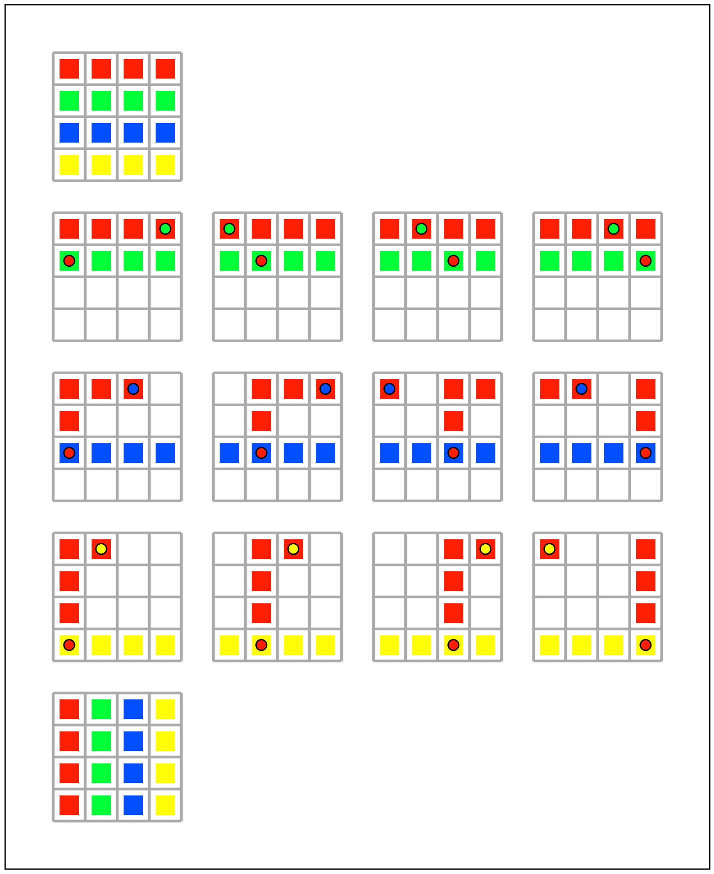

FIGURE 1. The window-shade method

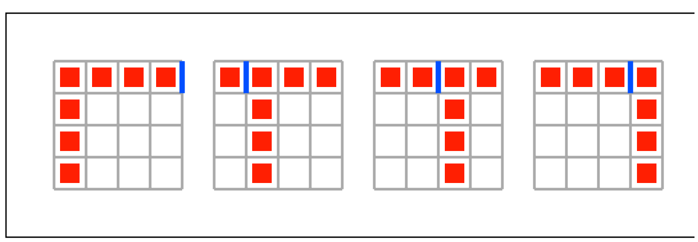

FIGURE 2. Window-shade sequences of length $2r-1$

We could avoid having to prove novel exchange properties involving more than two bases if we could count on finding a solution satisfying *all possible carpenters' rules*, as indicated in Figure 3 for $r=3$.

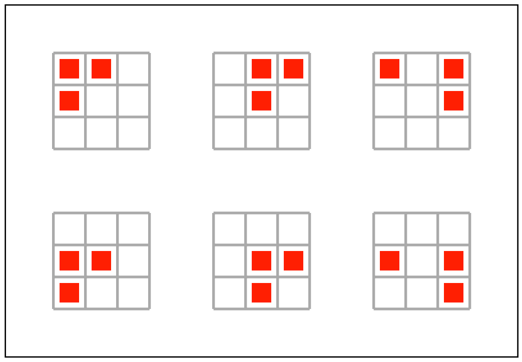

FIGURE 3. All possible ”carpenters’ rules”.

If such a *”strong”* window-shade method were to work, we would have a solution in which the intermediate bases, which occupy positions that resemble a carpenter’s rignt-angle tool, are all bases. This is an additional condition on transposes, and which, *if applicable*, would yield a straightforward algorithm for finding transposes for matrices with rows that are bases in a matroid. Do such solutions always exist?

NO. There is already a counterexample on my favorite matroid, the ”random” matroid on five points, with two three-point lines sharing a point. There are 8 bases: A list of 8 ’bases’,

$$abd, abe, acd, ace, bcd, bce, bde, cde.$$

See Figure 4.

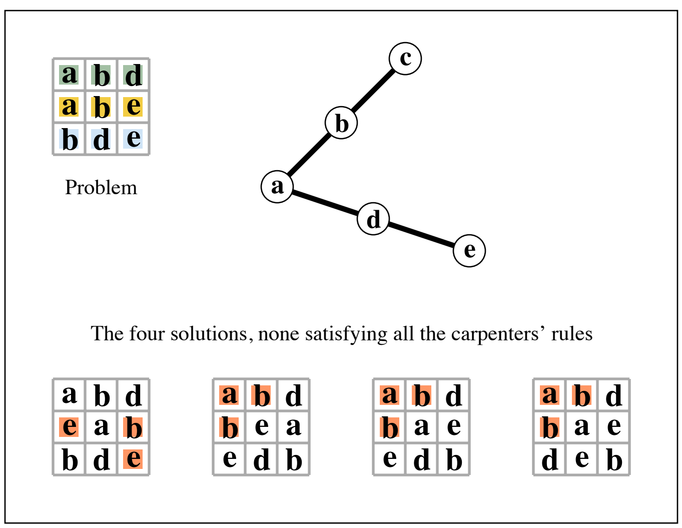

FIGURE 4. No transposes satisfy all the ”carpenters’ rules”.

However, the solution on the left in Figure 4 is windowshade, and is the unique such solution.

5. Window shade transposes.

Does every multi-set of r bases in a matroid of rank r have a transpose of window-shade form? I am prepared to guess that this is so.

In any window-shade transpose, and for $i = 2, \ldots, r$, we have a bijection from the first row to the ith row, replacing each "tail" element in row 1 by an element in row i, as each shade is "pulled". But this bijection from B_1 to B_i, for $3 <= i$ is not an exchange function from B_1 to B_i. Is is perhaps a predictable composite of exchange functions?

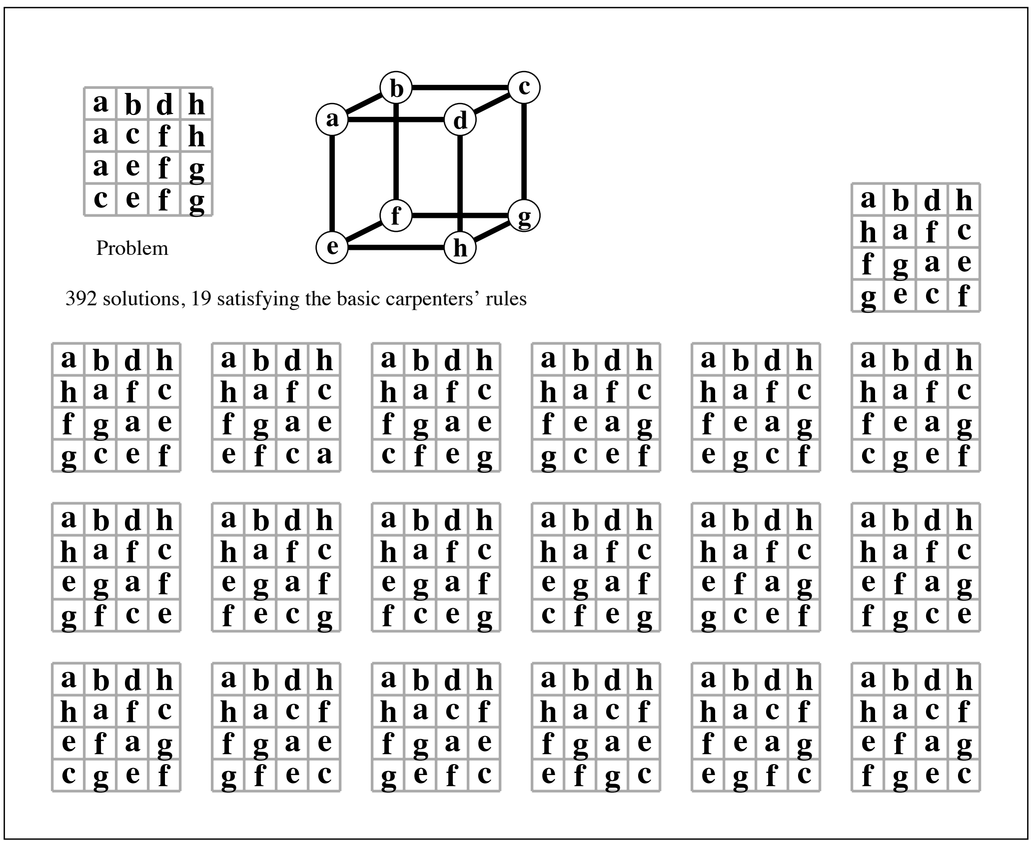

FIGURE 5. The 19 window-shade transposes for the four row bases indicated.

Are window-shade transpose solutions unique, as they seem often to be in rank 3. No, this is not the case. Consider the symmetric cube configuration on 8 points in rank 4, in Figure 5. The four bases given as rows of the matrix

$$\begin{pmatrix} a & b & d & h \\ a & c & f & h \\ a & e & f & g \\ c & e & f & g \end{pmatrix}$$

has 392 distinct transposes, 19 of which, as shown, are window-shade solutions. The two permutations of a, c, f, h that occur as second rows in these transposes, namely h, a, f, c and h, a, c, f, are obtainable directly (by cyclical left-shift) from the two bijective exchanges $\begin{pmatrix} a & b & d & h \\ a & c & f & h \end{pmatrix}$ and $\begin{pmatrix} a & b & d & h \\ a & f & c & h \end{pmatrix}$, respectively. The rather large number of window-shade solutions in this example may be related to the fact that there are four bijective exchanges from row 1 to row 3, and 15 bijective exchanges (!) from row 1 to row 4.

With another choice of four bases in the symmetric cube configuration, with each point occurring in only two of the bases,

$$\begin{pmatrix} a & b & d & g \\ a & c & e & f \\ b & d & g & h \\ c & e & f & h \end{pmatrix},$$

there are 808 transposes, of which 39 are window-shade. One such is given by the matrix

$$\begin{pmatrix} a & b & d & g \\ f & a & e & c \\ h & g & b & d \\ e & c & h & f \end{pmatrix},$$

6. From transpose failures to failures of exchange

For a struct with failures of transpose, it is quite shocking to watch the failures roll out. More often than not, especially when the rank of a struct $\mathbf{N}$ is not at roughly half of the cardinality of of the set E of points, it seems that the great majority of failures of transpose are for multi-sets of bases of the form $A, \ldots, A, B$, for just two bases A, B. These failures of transpose directly indicate a failure of bijective exchange, since any bijective exchange function provides a transpose of the form say for $A = (a, b, c, d)$ and exchange function $f : A \to B$,

$$\begin{pmatrix} a & b & c & d \\ d & a & b & c \\ c & d & a & b \\ f(b) & f(c) & f(d) & f(a) \end{pmatrix}$$

It *"seems"* that failures of transpose imply failures of transpose for multisets formed from just two bases. But since it is so hard to find failures of transpose on sets of r distinct bases, with cardinality $|E| < r^2$, I forced the issue by selecting a matrix with all points distinct, and generated structs in which that particular basis can have no transpose. Starting with rank 3, and the matrix

$$A = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix},$$

then removed from the list of 84 three-element subsets of the set $\{a, b, c, d, e, f, g, h, i\}$ a *minimal* set of triples that will block all feasible transposes of matrix A. The simplest such blocking set is the nine-element set

$$adg, adh, adi, aeg, aeh, aei, afg, afh, afi,$$

which blocks any possible first column in a transpose

$$A = \begin{pmatrix} a & b & c \\ \sigma(d) & \sigma(e) & \sigma(f) \\ \tau(g) & \tau(h) & \tau(i) \end{pmatrix}$$

where σ and τ are permutations of def and of ghi, respectively. There are many other types of minimal blocking sets, of slightly larger cardinality, such as this set of 14 triples:

$$adh, adi, aeg, afg, afh, afi, bdg, bdh, beh, bei, bfg, bfh, bfi$$

For this non-matroid struct, there is not only the matrix A that has no transpose, but a total of 1049 matrices with bases as rows, but no transposes. Of these matrices without transpose, 628 are of three distinct bases, while a surprisingly large number of 421 are multi-sets with just two bases, one repeated $r - 1$ times.

My laptop worked all one day to make a list of *all* such minimal blocking sets, but got entirely bogged down after having found

$$abc, abd, acf, acg, ach, aci, ade, adg, adh, aef, aeg$$

at the end of a list of 100 such minimal blocking sets. Since neither abc nor abd have not yet been removed, this must be very far indeed from the end of the search.

7. A Quick Search for a Counterexample

Choose a value for rank. Let's be reasonable, start with $r = 3$. It suffices to look at one 3×3 matrix

$$M = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

with nine distinct names of points. We start with a struct that has only 3 bases: the rows A, B, C of M, and place this struct $\mathbf{S}_0$ on an otherwise empty stack. We create a branching structure, the *leaves* of which are either structs having a transpose for M, or are matroids. At each pass of the program, we pop one struct $\mathbf{S}$ from the stack. We call a basis of $\mathbf{S}$ *vertical* if it contains one element from each of the rows A, B, C. Check to see whether $\mathbf{S}$ has three *disjoint* vertical bases. If so, M has a transpose in $\mathbf{S}$, and we treat this struct no further; it becomes a *green* leaf of the tree.

If M does not have a transpose in $\mathbf{S}$, we *expect* that the struct $\mathbf{S}$ is not a matroid, and so there will also be an ordered pair A, B of bases of $\mathbf{S}$ that has no bijective exchange (not necessarily a pair from the set $\{A, B, C\}$). If there is no failure of bijective exchange, we have a counterexample to Gian Carlo Rota's basis conjecture. There we can stop, with one *red* leaf on the tree. When we do find a pair A, B with no bijective exchange, we build a collection of all minimal sets of triples that, when added, will provide the missing exchange from A to B. For instance, if $A = abc$, B $= def$, we would add as bases:

$$\begin{array}{ccccc} abd, ace, bcf & \text{or} & abd, acf, bce, & \text{or} & abe, acd, bcf, \quad \text{or} \\ abe, acf, bcd, & \text{or} & abf, acd, bce, & \text{or} & abf, ace, bcd. \end{array}$$

Some of these bases may already be in $\mathbf{S}$, but there is at least one missing basis in each of these six lists. We thus place on the stack 6 copies of $\mathbf{S}$, to each of which we have added one of these six sets of at least one and at most three new bases. If the bases A and B have a single element, say a in common, and say $A = abc$ and $B = ade$, then we would simply add

$$abd, ace \quad \text{or} \quad abe, acd$$

(If A and B are *neighbors*, having all but one element in common, there is already a bijective exchange.)

If the program finishes without producing that *red* leaf, then we have a proof of the contrapositive of Rota's conjecture for rank 3: every struct lacking a transpose for some matrix of bases is a non matroid. Nothing new, because Wendy Chan proved this in 1995.

It might be barely feasible to provide a proof of this nature for rank 4, but I expected that rank 5 would outpace my little computer's capacity, if not first my patience.

Well, here's what really happened for rank 3. I took care to make my stack of structs "*LIFO*" so that the program will finish the larger structs first, and not build up an enormous stack of structs of intermediate size. My first attempts to run the program, hours would pass with no movement on the *Python* shell. When I added a few messages, I found that almost in 15 minutes the program had produced and stored a hundred thousand "green leaves", and in 3 hours, it had produced a millionof them. The next marker I had placed was at ten times that value, but by bed-time it had not been reached. In all that time, the stack of structs remaining to be treated always remained a stack of about fifty to ninety. I decided to let the program run. The next morning it had considered ten million structs, and a day later, it had produced ten million green leaves. Unfortunately, I had set the next message for 100 million, and no no longer had either the time to wait before leaving for New Zealand, nor the optimism that the program would finish its task, having found progressively more transposes toward the end. When I stopped the computer run, there were 61 structs on the stack. Only seven of these were large enough to have any *vertical* bases (bases with one element from each row of the original matrix). The rest had between 6 and 46 bases, the first 5 having been produced in an extremely early round of the program (the first encounter with a pair of disjoint bases). It is clear that the program was nowhere near completion of its run. I closed the computer, not with a renewed fear of exponential growth, but a respect for the quality of the human mind, which can produce a four page proof of propositions of this complexity, using only logical implication.

Which brings us back, in admiration, to the sort of sustained logical thinking that the matroid minors gang has been doing these past several years! Congratulations. And Happy Birthday, Geoff!

Figure 9. Konrad Jacobs photo from Oberwolfach, 1967.

Henry Crapo <moutons.matheux@gmail.com>
IMG_9673

14 D

IMG_9673