# Totally Positive & Totally Non-Negative Matrices

A.P. Neate

August 31, 2020

**Abstract**

A real matrix is totally positive (TP) (respectively totally non-negative (TNN)) if each of its minors is strictly positive (respectively non-negative). (Note in some sources the terms strictly totally positive and totally positive are used instead [5]). In this report, we explore total positivity and total non-negativity of a variety of matrices as well as methods for verifying such properties.

## 1 Conventions and Definitions

The interval $[i, j] \subset \mathbb{Z}$ will be used to denote the set $\{k \in \mathbb{Z} : i \leq k \leq j\}$.
The notation $w_0$ will describe the permutation on the set $\{1, \ldots, n\}$ that sends $i$ to $n + 1 - i$. Matrices shall be denoted by upper-case Roman letters and their entries will be denoted by the corresponding lower-case letter with subscripts.

**Definition 1.1.** We define a *planar network* to describe an acyclic planar directed graph consisting of $n$ sources and $n$ sinks.

**Definition 1.2.** A *path weight* is the product of the weights of a set of connected edges where if an edge has no weight, we take it to be 1.

**Definition 1.3.** A *weight matrix* is defined for a directed planar network such that the $i, j$th entry gives the sum of path weights for each of the paths from source $i$ to sink $j$

**Definition 1.4.** For matrix $M$ and equally sized sets of rows indexed by $I$ and columns indexed by $J$, a *minor* $M(I, J)$ is the determinant of the square sub-matrix defined by the shared matrix elements of the rows in $I$ and the columns in $J$.

**Definition 1.5.** For matrix $M$, the minor $M(I, J)$ is *initial* if $I$ and $J$ are consecutive integers of the same length and one is of the form $[1, k]$ for integer $k$.

**Definition 1.6.** An *elementary matrix* $E_{ij}$ is defined such that it differs from the zero matrix only in its $i, j$th entry which is 1.

## 2 Lindström's Lemma

The importance of planar networks in connection with total nonnegativity derives from the following fundamental result due to Lindstrom and its converse due to Brenti.

**Lemma 2.1** (Lindström [1]). *The weight matrix of any planar network with edges only of non-negative weight is totally non-negative.*

*Proof.* We show this through induction on matrix size. For matrices of size $1 \times 1$ clearly a matrix of this size is totally non-negative if and only if its single entry is non-negative. There is an obvious bijection between $1 \times 1$ matrices and the planar network containing two nodes connected by an edge of weight equal to that of the entry of the matrix and hence the Lemma holds in this case.

Now for the inductive step; consider an $n \times n$ weight matrix $M$ with entries $m_{i,j}$ of a planar network $G$. Each minor of $M$ is a sub-graph of $G$ consisting of only sources corresponding to rows and sinks corresponding to columns as well as every connecting edge and node which is not a sink or source. Since each minor of $M$ is a weight matrix of some planar network, we can assume these are TNN and it remains to show that $\det(M)$ is non-negative.

Consider the formula $\det(M) = \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) m_{1\sigma(1)} m_{2\sigma(2)}, \ldots, m_{n\sigma(n)}$. We see products in the determinant sum are the same as the sum of path weights of families of $n$ paths where the $i$th path goes from the $i$th source to the $\sigma(i)$th sink. Notice that if two paths in a family intersect then the set of edges covered by this family are the same as those in the family where you swap the paths after their first intersection (least edges from source). Swapping paths after an intersection is the same as a transposition of the two sinks. Hence the two terms in the sum for the determinant that represent the path families being swapped have opposite signs and hence, cancel each other out. This leaves the families of paths which contain no intersections as the only ones which can contribute to the sum. If a family contains no intersections, it must be associated with the identity permutation as the graphs are planar. Therefore, families containing no intersections have sign of $+1$ and hence $\det(M)$ is the weighted sum of the families of non-intersecting paths which must be a non-negative number. □

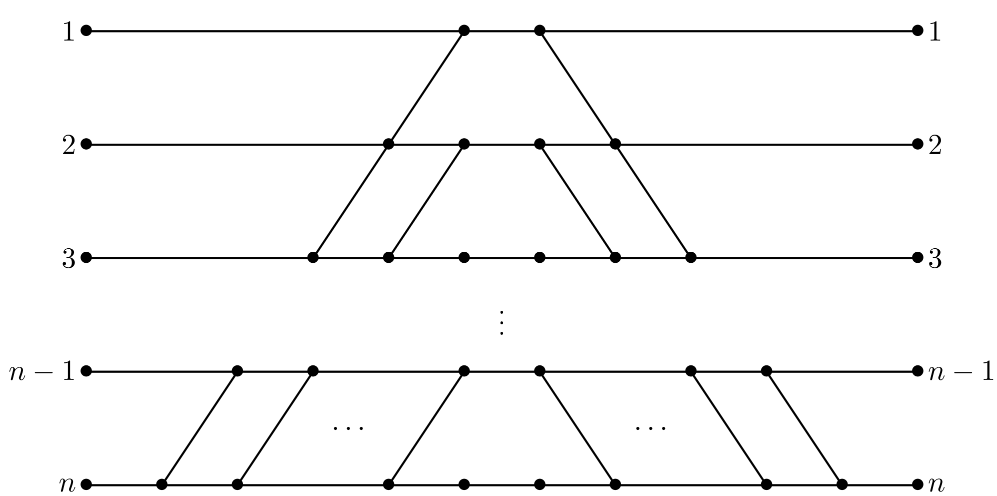


Figure 1: Planar Network of General Weight Matrix

In addition to Lindström's result, Brenti [6] also showed that the converse is true, that any TNN matrix is the weight matrix of some planar network. Now consider Figure 1 a planar network such that edges are directed left to right. If you assign non-negative weights to the diagonal and central horizontal edges on the graph in Figure 1 then there is a bijection between $n \times n$ TNN matrices and graphs of this form [4]. This standard form of a planar network also gives all the TP matrices if every edge is non-zero.

## 3 Pascal's Triangle

Take the $n \times n$ Pascal's triangle matrix to be the lower triangular matrix consisting of the rows of Pascal's triangle, that is the $n \times n$ matrix $M$ such that $m_{ij} = \binom{i}{j}$. This matrix is well known to be TNN, see, for example, [2, Chapter 32], where the Pascal's triangle matrix is realised as the weight matrix of a planar network. If we take the first 7 rows of Pascal's triangle, we have the following matrix:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 \\ 1 & 3 & 3 & 1 & 0 & 0 & 0 \\ 1 & 4 & 6 & 4 & 1 & 0 & 0 \\ 1 & 5 & 10 & 10 & 5 & 1 & 0 \\ 1 & 6 & 15 & 20 & 15 & 6 & 1 \end{pmatrix}$$

It can be shown this Pascal's matrix is TNN by direct calculation of the paths in Figure 2 if we consider the weight of every edge to be 1. The planar network given, however, has a clear structure, in fact, if we extend Figure 2 in the obvious way, we can show the number of paths between source $n$ and sink $k$ is $\binom{n}{k}$.

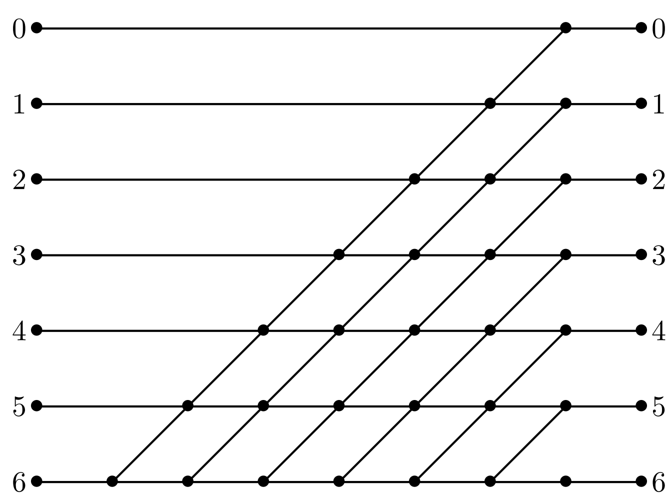


Figure 2: Pascal's triangle as a standard planar network

One method of proof is to consider the recurrence relation $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

**Theorem 3.1.** *Pascal's Triangle is the weight matrix for the planar network which is the generalisation of Figure 2 and hence is totally non-negative.*

*Proof.* We can see from Figure 2 that the recurrence $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$ holds for $n, k \leq 2$ and also that the initial conditions of $\binom{0}{0} = 1$, $\binom{0}{n} = 0$ for $n > 0$ are true. All that remains is to show that the recurrence will always hold.
Assume for induction that the recurrence holds up to $n-1$; we wish to show that the recurrence holds for $n$. Take some $k$ and consider the first node on the network after source $n$, we have a choice to either take the diagonal or the horizontal edge. If we take the diagonal edge then the remaining number of paths shall be equal to the number of paths from source $n-1$ to sink $k$ or $\binom{n-1}{k}$. Instead, consider the paths which take the horizontal edge. If we take the horizontal edge then we cannot traverse the outer diagonal or the first row of the network. Thus, if we remove these edges then we can see the network is identical except that all the sources and sinks are indexed one higher and hence the paths from the node after the horizontal edge to $k$ will be identical to the paths from node $n-1$ to $k-1$ as shown for $n = 6$ and $k = 4$ in Figure 3.

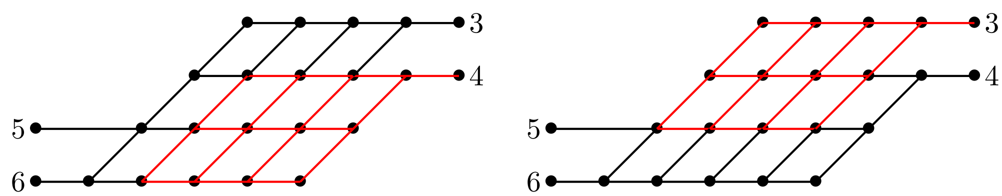


Figure 3: Comparing sub-graphs

Since every path from source $n$ to sink $k$ must travel through exactly one of the two edges described then the calculated paths together make up the total number of possible paths. Hence, we can see that the recurrence holds and therefore, Pascal's triangle is indeed the weight matrix of the network described. □

We could also consider the following proof. Consider that since the network is a directed graph, moving along an edge moves us to the right. Therefore, regardless of how we move, there must always be $n$ nodes were we make a choice as to which edge we take. At each of these nodes, we have the choice of taking the diagonal or the horizontal. If we wish to arrive at source $k$, we must go horizontally exactly $k$ times as we must go diagonally $n - k$ times. Since we are choosing $k$ horizontal edges out of $n$ total edges, we can see that the number of paths from source $n$ to sink $k$ is $\binom{n}{k}$.

What total non-negativity shows is that any minor of Pascal's triangle can also be written as a planar network simply by considering the sources and sinks corresponding to the rows and columns of the matrix. For example consider the minor

$$M = \begin{pmatrix} \binom{3}{1} & \binom{3}{3} & \binom{3}{4} \\ \binom{4}{1} & \binom{4}{3} & \binom{4}{4} \\ \binom{6}{1} & \binom{6}{3} & \binom{6}{4} \end{pmatrix}$$

We can re-draw the network in Figure 2 to that shown in Figure 4 simply by removing any non-functional edges to provide a planar network whose weight matrix is $M$.

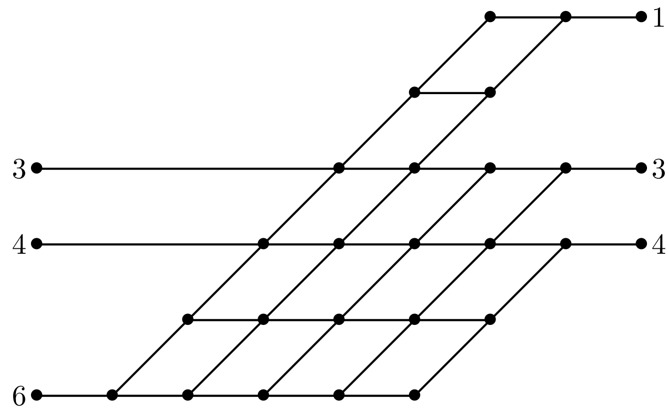


Figure 4: Planar network of weight matrix minor $M$

We can see here quite clearly by counting paths that $M$ is the weight matrix of network in Figure 4.

## 4 Stirling Numbers

In the previous section, we have seen how Lindström's Lemma can be used with non-weighted planar networks. In this section, we will consider weighted planar networks which realise Stirling numbers of the first and second kind whose matrices form triangles like that of Pascal's. Through Lindström's Lemma we will also show these triangles to be TNN.

**Definition 4.1.** The unsigned *Stirling number of the first kind* $C(n, k)$ is the number of permutations of $[1, n]$ with exactly $k$ cycles where we define $C(0, 0) = 1$

**Definition 4.2.** The *Stirling number of the second kind* $S(n, k)$ is the number of partitions of $[1, n]$ into $k$ non-empty disjoint subsets

We also have the following recurrence relations which can be proven combinatorially:

$$C(n, k) = C(n - 1, k - 1) + (n - 1)C(n - 1, k)$$

$$S(n, k) = S(n - 1, k - 1) + kS(n - 1, k)$$

We shall also use the following notation. Let $C_n$ denote the $n \times n$ matrix where $c_{ij} = C(i, j)$ and $S_{nn}$ denote the $n \times n$ matrix where $s_{ij} = S(i, j)$. Each of these matrices is lower triangular and hence gives us Stirling triangles as both kinds of Stirling numbers rely on order $k$ subsets of $n$ of which there can only be a non-zero number if $n \geq k$.

### 4.1 Stirling Numbers of the First Kind

First consider the first 4 rows of the triangle of Stirling numbers of the first kind.

$$C_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 2 & 3 & 1 & 0 \\ 6 & 11 & 6 & 1 \end{pmatrix}$$

Using the recurrence relation we can derive the planar network that the matrix $C_4$ is the weight matrix of.

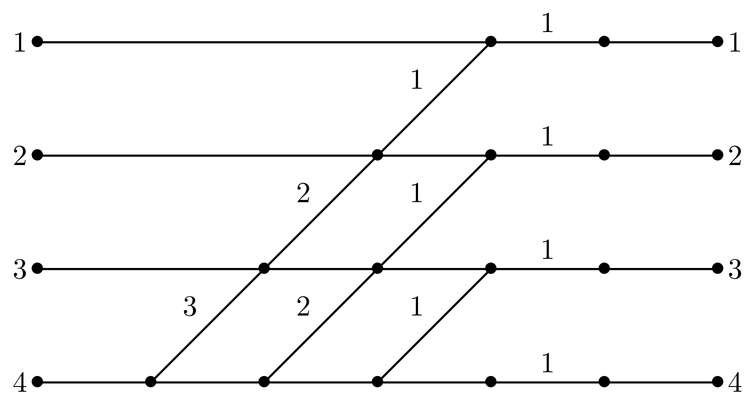


Figure 5: Planar network of $C_{4,4}$

From these first four rows we start to see a pattern in the network. In fact, this is due to the $n-1$ term in the recurrence. We will show this pattern does indeed hold in general.

**Theorem 4.3.** *The matrix $C_n$ is a weight matrix and hence is TNN. Furthermore, the planar network in the standard form (as in Figure 1) for which $C_n$ is the weight matrix is defined as follows. Let the weight of the diagonal edge connected to the $j$th node on the same row as the $i$th source be denoted $P_{ij}$ then $P_{ij} = i - j$, all horizontal edges have weight 1 and all weights of diagonals after the horizontal edge are 0.*

*Proof.* The idea is to define a planar network $P$ whose weight matrix can be proved to be the same as the matrix for the Stirling numbers of the first kind. The planar network $P$ is the obvious generalisation of the picture drawn in Figure 5 where $P$ has $n$ sources $s_i$ and $n$ sinks $t_i$. The $i$th row has one source $s_i$ and one sink $t_i$ and $i-1$ other vertices. For the vertices that are not sources or sinks, vertex $j$ on row $i$ connects to vertex $j$ on row $i-1$ and has weight $i-j$. Let $Q = (q_{ij})$ be the weight matrix for this planar network. We aim to show that $Q = C$ where $C$ is such that $c_{ij}$ is the Stirling number of the first kind $C(i,j)$. To do this, we see that the entries in $Q$ and $C$ agree up to the 4th row by counting paths on Figure 5. Next, we must prove that the entries of $Q$ satisfy the same recurrence relation as the Stirling numbers of the first kind; that is, prove that $q_{m,k} = q_{m-1,k-1} + (m-1)q_{m-1,k}$. We can check this by hand for the low values calculated above. Assume that the recurrence has been checked up to

$m-1$ and try to prove for $m$. Let $\Pi_{m,k}$ be the set of paths from $s_m$ to $t_k$. Then,

$$q_{m,k} = \sum_{\pi \in \Pi_{m,k}} w(\pi).$$

Define $e_1$ to be the edge connecting the first vertex of row $m$ to the first vertex of row $m-1$. We divide $\Pi_{m,k}$ into two disjoint subsets: those that include $e_1$ and those that don't. Now $e_1$ connects the first vertex of row $m$ with the first vertex of row $m-1$. Thus, a path taking this route starts at $e_1$ with weight $m-1$ and then continues along a path from $s_{m-1}$ to $t_k$. Hence, the sum of all the weights of these paths is $(m-1)q_{m-1,k}$.
Any path that does not contain $e_1$ cannot go up any of the first diagonal edges and cannot go along any part of the line from $s_1$ to $t_1$. Thus, we can delete these vertices and edges, and see that on the resulting planar network the sum of the weights of paths not going along $e_1$ is $q_{m-1,k-1}$. This proves that $q_{m,k} = q_{m-1,k-1} + (m-1)q_{m-1,k}$, as required. □

### 4.2 Stirling Numbers of the Second Kind

Similarly to that of Stirling numbers of the first kind, we also have a triangle of Stirling numbers of the second kind and a planar network for which the $4 \times 4$ case is the weight matrix of

$$S_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 3 & 1 & 0 \\ 1 & 7 & 6 & 1 \end{pmatrix}$$

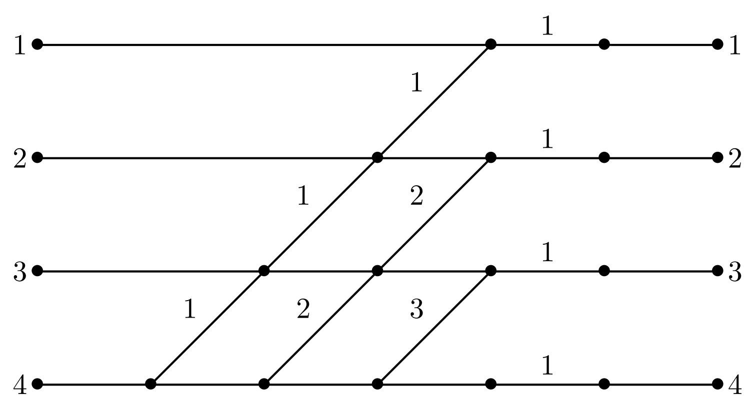


Figure 6: Planar network of $S_{4,4}$

As with Stirling numbers of the first kind, we have an obvious pattern in the graph tied to the recurrence relation. Here. each diagonal is one higher due to the $k$ term in the recurrence.

**Theorem 4.4.** *The matrix $S_n$ is a weight matrix and as such is TNN. Furthermore, the planar network in the standard form for which $S_n$ is the weight matrix is defined as follows. Let the weight of the $j$th diagonal edge from the left coming up from the $i$th source be denoted $P_{ij}$ then*

*$P_{ij} = j$, all horizontal edges have weight 1 and all weights of diagonals after the horizontal edge are 0.*

*Proof.* As before, the idea is to define a planar network $P$ whose weight matrix can be proved to be the same as the matrix for the Stirling numbers of the second kind. The planar network $P$ is the obvious generalisation of the picture drawn in Figure 6 where $P$ has $n$ sources $s_i$ and $n$ sinks $t_i$. The $i$th row has one source $s_i$ and one sink $t_i$ and $i-1$ other vertices. For the vertices that are not sources or targets, vertex $j$ on row $i$ connects to vertex $j$ on row $i-1$ and has weight $i-j$. Let $Q = (q_{ij})$ be the weight matrix for this planar network. We aim to show that $Q = C$ where $C$ is such that $c_{ij}$ is the Stirling number of the second kind $S(i,j)$. To do this, we see that the entries in $Q$ and $C$ agree up to the 4th row by counting paths on Figure 6. Next, we must prove that the entries of $Q$ satisfy the same recurrence relation as the Stirling numbers of the second kind; that is, prove that $q_{m,k} = q_{m-1,k-1} + (m-1)q_{m-1,k}$. We see the recurrence holds for $m \leq 4$ from above. Assume that the recurrence has been checked up to $m-1$ and try to prove for $m$. Let $\Pi_{m,k}$ be the set of paths from $s_m$ to $t_k$. Then,

$$q_{m,k} = \sum_{\pi \in \Pi_{m,k}} w(\pi).$$

Define $e_{-1}$ to be the edge connecting the last vertex of row $k+1$ connecting to a vertex of row $k$. We divide $\Pi_{m,k}$ into two disjoint subsets: those that include $e_{-1}$ and those that don't. Now, $e_{-1}$ connects the last vertex of row $k$ with row $k+1$. Thus, a path taking this route finishes at $e_{-1}$ with additional weight $k$ after having taken a path equivalent in weight to $q_{m-1,k}$. Hence, the sum of all the weights of these paths is $kq_{m-1,k}$.
Any path that does not contain $e_{-1}$ cannot go up any of the last diagonal edges. Thus, we can delete these edges and see that on the resulting planar network the sum of the weights of paths not going along $e_{-1}$ is $q_{m-1,k-1}$. This proves that $q_{m,k} = q_{m-1,k-1} + (m-1)q_{m-1,k}$, as required. □

## 5 Vandermonde Matrices

We will now consider another class of matrices. Vandermonde matrices are used in areas of mathematics such as curve fitting and are defined as follows:

**Definition 5.1.** A *Vandermonde matrix* is defined for $0 < a_1 \leq a_3 \leq \cdots \leq a_m$ as:

$$V = \begin{pmatrix} 1 & a_1 & a_1^2 & \dots & a_1^{n-1} \\ 1 & a_2 & a_2^2 & \dots & a_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_m & a_m^2 & \dots & a_m^{n-1} \end{pmatrix}$$

We will show that when all $a_i$ are distinct, the Vandermonde matrix is TP but to do so we require the following result.

**Theorem 5.2** (Gasca and Peña, [3]). *A square matrix is totally positive if and only if all of its initial minors are positive.*

Now if we consider the general square Vandermonde $V$ then the determinant of $V$ is well known in general as $\det(V) = \prod_{1\leq i<j\leq n}(a_j - a_i)$. Since $(a_i)$ is increasing, we have $\det(V) \geq 0$ with equality if and only if there exists non-distinct $a_i$. Further notice that the initial minors of a Vandermonde matrix are either also Vandermonde matrices or are of the form:

$$V_m = \begin{pmatrix} a_1^m & a_1^{m+1} & \dots & a_1^{m+k-1} \\ a_2^m & a_2^{m+1} & \dots & a_2^{m+k-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_k^m & a_k^{m+1} & \dots & a_k^{m+k-1} \end{pmatrix} = \text{diag}(a_1^m, a_2^m, \dots, a_k^m) \times \begin{pmatrix} 1 & a_1 & \dots & a_1^{k-1} \\ 1 & a_2 & \dots & a_2^{k-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & a_k & \dots & a_k^{k-1} \end{pmatrix}$$

Hence, if we assume the $a_i$ are distinct, since each term is positive, then the diagonal matrix has positive determinant. Therefore, the sign of the determinant of these minors is decided by the sign of the determinant of some Vandermonde matrix which must be positive; hence $V$ is totally positive. Now this only proves TP for square Vandermonde matrices. However, notice that any rectangular Vandermonde matrix is the sub-matrix of some square Vandermonde matrix. Since TP implies TP of sub-matrices then all Vandermonde matrices with distinct $a_i$ are TP.

Given that Vandermonde matrices are totally positive, we must be able to draw a planar network for which a Vandermonde matrix is the weight matrix. The $4 \times 4$ case gives us the following planar network

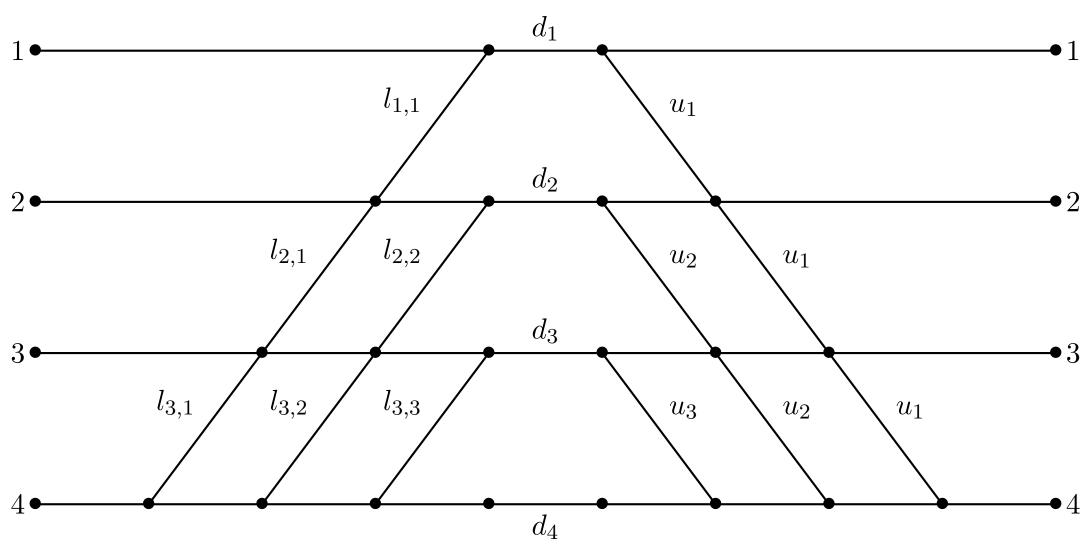


Figure 7: Planar Network of $4 \times 4$ Vandermonde Matrix

Where:

$$l_{ij} = \begin{cases} 1 & j = 1 \\ \prod_{k=1}^{j-1}(a_i - a_{i-k})(a_{i-1} - a_{i-k-1})^{-1} & j > 1 \end{cases}$$

$$d_i = \prod_{k=0}^{i-1} (a_i - a_k)$$

$$u_j = a_j$$

# 6 Algorithms for Checking TNN

## 6.1 Neville Elimination

We have spoken about proving TNN in specific instances of matrices. Now, we will consider an algorithmic approach to checking TNN. Neville Elimination is a method for factorising a matrix $A$ into upper and lower triangular matrices. By restricting the matrix operations to subtraction of multiples of adjacent rows, we can preserve and hence prove total non-negativity. The algorithm on a $m \times n$ matrix $A$ was given in [10]; since in this report we are only considering applying this algorithm to invertible matrices, the algorithm can be simplified for a $n \times n$ matrix to the following:

### Neville Elimination Algorithm

Step 1: Set $L = I_n$ and $U = A$

Step 2: If $U$ is upper triangular stop and output $L$ and $U$. Otherwise, go to Step 3.

Step 3: If the first column of $U$ has two or more non-zero entries then set $t = 1$. Otherwise, set $t$ such that the sub-matrix of $U$ consisting of the first $t-1$ columns is in upper echelon form but including column $t$ the sub-matrix is not. Take the largest integer $s$ such that $u_{st}, u_{s+1,t} \neq 0$ and $u_{ij} = 0$ for $i \geq s,\ j < t$, replace $U$ by $(I - u_{s+1,t}u_{st}^{-1}E_{s+1,s})U$ and replace $L$ by $L(I + u_{s+1,t}u_{st}^{-1}E_{s+1,s})$. Note $A = LU$, go to Step 2.

In [10], it was shown that each of these Neville moves in Step 3 preserve TNN, hence, if each of the matrices $I + u_{s+1,t}u_{st}^{-1}E_{s+1,s}$ is TNN (i.e. at every Step 4 we have that $u_{s+1,t}u_{st}^{-1} > 0$) then $L$ must be TNN as it is the product of TNN matrices. Similarly, we can apply this process to $A^T$ to find a factorisation of $U$ and hence prove the matrix TNN.

## 6.2 Deletion Algorithm

In [9], a result gives us a simpler way to calculate whether a specific instance of a matrix is TNN. For a $m \times p$ matrix $M$, we first define the function $g_{j\beta}(M) = (x'_{i\alpha})$ where:

$$\begin{cases} x_{i\alpha} - x_{i\beta}x_{j\beta}^{-1}x_{j\alpha} & \text{if } x_{j,\beta} \neq 0,\ i < j \text{ and } \alpha < \beta \\ x_{i\alpha} & \text{otherwise} \end{cases}$$

If we take the matrix $M$ then the output of the deletion algorithm is $\bar{M} = g_{1,1} \circ g_{1,2} \circ g_{m,p-1} \circ g_{mp}(M)$. The importance of this function is that the output is a matrix which has easy conditions to check that imply TNN of $M$. The first of the conditions on $\bar{M}$ for TNN of $M$ is that $\bar{M}$ contains only non-negative entries. The second condition is that $\bar{M}$'s zeroes form a Cauchon diagram, meaning that if a zero exists then either all elements above or all elements to the zero's left must be zero. These conditions are a result shown in Theorem 4.1 of [9].

I implemented both of these algorithms (see the Appendix for the code used). I used both the deletion algorithm and Neville elimination algorithm to fully factorise the same matrix finding that the deletion algorithm was more than twice as fast to compute on a $10 \times 10$ matrix.

## 7 Eulerian Numbers

**Definition 7.1.** For a permutation $\sigma \in S_n$, a *descent* is an element $i$ such that $\sigma(i) > \sigma(i+1)$.

**Definition 7.2.** The *Eulerian numbers* denoted $E(n, k)$ are the number of permutations of $S_n$ with $k$ descents.

Eulerian numbers follow the recurrence relation [11]

$$E(n, k) = (n - k)E(n - 1, k - 1) + (k + 1)E(n - 1, k).$$

This relation is similar to that of binomial and the Stirling numbers but does not give us a planar network that is quite as nice. Below is the $5 \times 5$ matrix of Eulerian numbers where for $n \in [1, 5],\ k \in [0, 4]$ the $n, k+1$th entry is $E(n, k)$ as well as the corresponding planar network for which this is the weight matrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 \\ 1 & 11 & 11 & 1 & 0 \\ 1 & 26 & 66 & 26 & 1 \end{pmatrix}$$

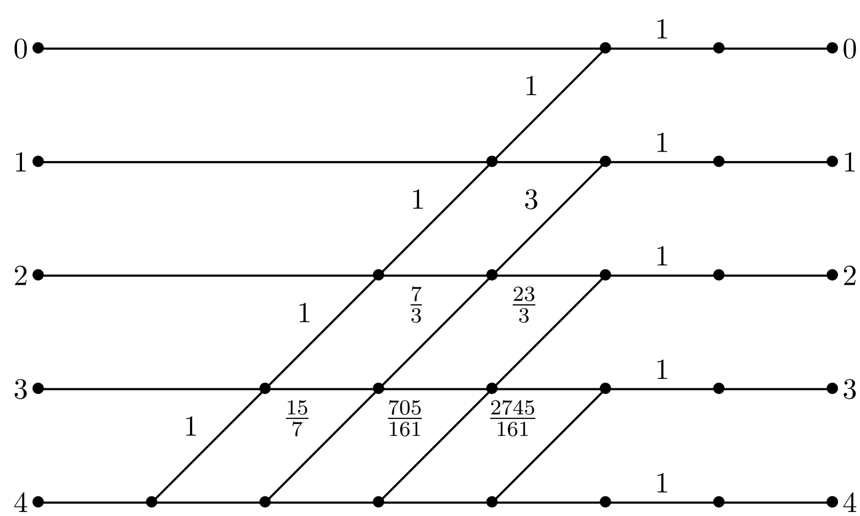


Figure 8: Planar network of the $5 \times 5$ Eulerian triangle

Regarding the general Eulerian triangle we have the following

**Conjecture 7.3** (Brenti [6]). *The Eulerian triangle is TNN.*

As we can see, there is no obvious pattern giving the network in Figure 8 and I have found no proof the the Brenti conjecture. I have however tested total non-negativity of the triangle using the deletion algorithm up to the matrix of size 100 and as of yet have found no contradiction.

# 8 Catalan Numbers

**Definition 8.1.** The *Catalan numbers* $C_n = \frac{1}{n}\binom{2n}{n+1} = \frac{1}{n+1}\binom{2n}{n}$

## 8.1 Catalan-Shapiro Triangle

**Definition 8.2.** The *Catalan-Shapiro Triangle* is the lower triangular matrix $B$ where $B_{nk} = \frac{k}{n}\binom{2n}{n+k}$. The first 5 rows are given below:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 \\ 5 & 4 & 1 & 0 & 0 \\ 14 & 14 & 6 & 1 & 0 \\ 42 & 48 & 27 & 8 & 1 \end{pmatrix}$$

**Lemma 8.3.** *The entries of the Catalan-Shapiro Triangle satisfy the recursion* $B_{n+1,k} = B_{n,k-1} + 2B_{nk} + B_{n,k+1}$

**Theorem 8.4** ([5] 4.7). *A Toeplitz matrix for bi-infinite sequence $(a_i)$ is defined $A = (a_{i-j})_{i,j=1}^{\infty}$ and is totally non-negative if the generating function for $(a_i)$ has the form $\sum_k a_k z^k = e^{\gamma z}\frac{\prod_i(1+\alpha_i z)}{\prod_i(1-\beta_i z)}$ for $\gamma, \alpha_i, \beta_i \geq 0$ and $\sum_i(\alpha_i + \beta_i) < \infty$. For finite sequence $(a_i)$, $i \in [0, n]$ this is equivalent to the polynomial $p(x) = \sum_{k=0}^{n} a_k x^k$ having $n$ negative roots.*

Wang-Wang [8] proved TNN of this matrix in the following way. Using Theorem 8.4, the Toeplitz matrix $T$ given by the sequence $(1, 2, 1)$ is totally non-negative (the generating function has a double root of -1). Due to the recursive formula for the Catalan-Shapiro triangle, we have that $B = \begin{pmatrix} 1 & \mathbf{0} \\ \mathbf{0} & B \end{pmatrix} T$ and since both the identity matrix and $T$ are TNN, it is easily shown by induction that the Catalan-Shapiro triangle must also be TNN.

## 8.2 Catalan Triangle

**Definition 8.5.** The *Catalan Triangle* is the lower triangular matrix $C$ defined for $n, k \geq 0$ such that $c_{nk}$ denotes the number of unique binary strings consisting of $n$ 0's and $k$ 1's where no initial segment of the string has more 1's than 0's

The first 5 rows of the Catalan triangle are given below:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 2 & 0 & 0 \\ 1 & 3 & 5 & 5 & 0 \\ 1 & 4 & 9 & 14 & 14 \end{pmatrix}$$

**Theorem 8.6** ([7]). *Ignoring the first row and column the main diagonal, the Catalan triangle is exactly the Catalan numbers. Furthermore, the triangle satisfies the following recurrence relation:*

1. $c_{n,0} = 1$ *for* $n \geq 0$
2. $c_{n,1} = n$ *for* $n \geq 0$
3. $c_{n,k} = c_{n,k-1} + c_{n-1,k}$ *for* $1 < k \leq n$

**Theorem 8.7** ([7]). *The triangle also satisfies the general formula:*

$$c_{nk} = \binom{n+k}{k} - \binom{n+k}{k-1} = \frac{(n+k)!(n-k+1)}{k!(n+k)!}$$

Catalan triangle is TNN up to the fourth row but if we consider the $5 \times 5$ case, we find a negative minor. Applying Neville elimination, clearing a column at a time, we get the following matrices

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 2 & 0 & 0 \\ 1 & 3 & 5 & 5 & 0 \\ 1 & 4 & 9 & 14 & 14 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 & 0 \\ 0 & 1 & 3 & 5 & 0 \\ 0 & 1 & 4 & 9 & 14 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 5 & 0 \\ 0 & 0 & 1 & 4 & 14 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & -1 & 14 \end{pmatrix}$$

Since Neville elimination preserves TNN, this shows that the Catalan triangle is not TNN. A minor total non-negativity fails for is [2345|1234]. We can understand why this fails when we consider the recurrence relation since subtracting adjacent rows will set an element to that on its left unless it is on the diagonal where we have just subtracted 0. When we reach the $5 \times 5$ case, the diagonal has larger elements and this is where we get the $-1$.

# 9 Cross-Symmetric Matrices

**Definition 9.1.** A $n \times n$ matrix $M$ is *Cross-Symmetric* (also known as centro-symmetric) if for permutation $w_0$, $M_{ij} = M_{w_0(i),w_0(j)}$ for all $i, j \in [1, n]$.

These matrices are given their name as the elements of this matrix are symmetric about its center. Cross-symmetric matrices arise in various places; interesting examples are Amazing Matrices [12][13] which we will introduce later.

**Definition 9.2.** An *elementary C-S (cross-symmetric) matrix* is one of the following two types:

1. *C-S Neville*: A matrix which for $a \geq 0$ has the form $F_i = I - aE_{i,i-1} - aE_{w_0(i),w_0(i-1)}$ or is the inverse of such a matrix

2. *C-S Row-Swap*: A cross symmetric matrix which swaps adjacent rows and is either of the form $R_{n/2} = I - E_{n/2,n/2} + E_{n/2+1,n/2} + E_{n/2,n/2+1} - E_{n/2+1,n/2+1}$ or when $i \neq n/2$ of the form $R_i = I + (-E_{i,i} + E_{i+1,i} + E_{i,i+1} - E_{i+1,i+1}) + (-E_{w_0(i),w_0(i)} + E_{w_0(i+1),w_0(i)} + E_{w_0(i),w_0(i+1)} - E_{w_0(i+1),w_0(i+1)})$

**Lemma 9.3.** *Multiplication by cross-symmetric elementary matrices preserves cross-symmetry.*

*Proof.* Clearly these matrices are defined as to be cross-symmetric and it is well known that the product of cross-symmetric matrices must also be cross-symmetric. This can easily be shown by introducing the matrix $J$ that is 1 on the anti-diagonal and 0 elsewhere and then noticing that cross-symmetry of a matrix $A$ is equivalent to the condition that $JA = AJ$. Now, if $A$ and $B$ are cross-symmetric then $JAB = AJB = ABJ$ and hence $AB$ is cross-symmetric also. □

These elementary C-S matrices closely resemble those of the matrices used in Neville elimination. In fact, we can use these matrices to similarly factorise matrices while preserving cross-symmetry while at the same time exploiting cross-symmetry to increase efficiency.

### Cross-Symmetric Neville Elimination Algorithm

Step 1: Take the non-zero entry $M_{ij}$ below the diagonal which minimises $j$ then maximises $i$ (most left then most bottom). If no such element exists we are done; else, if $M_{i-1,j}$ is non-zero go to Step 2. otherwise, go to Step 3.

Step 2: Take the C-S Neville matrix $F_i = I - M_{ij}/M_{i-1,j}E_{i,i-1} - M_{ij}/M_{i-1,j}E_{n-i+1,n-(i-1)+1}$ and replace $M$ by $F_iM$. Go to Step 1.

Step 3: If $n$ is even and $i - 1 = n/2$ then go to Step 4. Else, if $n$ is odd and $i = (n+1)/2$ go to Step 6. Otherwise go to Step 5.

Step 4: Replace $M$ by $R_{n/2}M$ swapping the central two rows. Go to Step 1.

Step 5: Replace $M$ by $R_iM$. Go to Step 1.

Step 6: Take the largest $k < i$ such that $M_{k,j} \neq 0$. If no such $k$ exists then go to Step 1 ignoring column $j$ in all future calculations; else, if $k = i - 1$ go to Step 1. Otherwise, replace $M$ by $F_kM$ where $F_k = I + E_{k,k+1} + E_{w_0(k),w_0(k+1)}$ and repeat Step 6.

**Lemma 9.4.** *Any invertible cross-symmetric matrix of size $n$ can be written as the product of elementary cross-symmetric matrices and a diagonal matrix.*

*Proof.* Consider the above algorithm for matrix $M$ of size $n$. At each stage when we change the matrix $M$, we do so by pre-multiplication of an elementary matrix, all of which are clearly invertible ($R_i^{-1} = R_i$ and the inverse of a C-S Neville matrix is defined to be a C-S Neville matrix). Assume we don't need the operation which ignores column $j$ in Step 6. Now by induction the algorithm must result in a diagonal matrix $D$ after the product of some set of elementary $C-S$ matrices as each step reduces the number of non-zero elements. Now, take the inverse of elementary $C-S$ matrices which reduced $M$ and we have $M$ as the product of elementary C-S matrices as required.

Assume instead we do ignore some column $j$ in Step 6 at some stage in the algorithm. In this case, we must have that $M$ is singular. To show this, notice that since $M_{ij}$ is the central element of an otherwise 0 column and since all operations preserve cross-symmetry, we must always have $M_{jj} = M_{w_0(j)j} = M_{jw_0(j)} = M_{w_0(j)w_0(j)}$. Similarly, for any other ignored column $k$, we must have $M_{jk} = M_{w_0(j)k} = M_{jw_0(k)} = M_{w_0(j)w_0(k)}$. Now notice that every column not ignored will be 0 except for on the diagonal and hence 0 on row's $j$ and $w_0(j)$. Therefore we must have that rows $j$ and $w_0(j)$ are identical and hence $M$ is singular. This implies that ignoring a column in Step 6 cannot occur if $M$ is invertible and thus we already have a factorisation of $M$ as required. □

## 10 Amazing Matrices

In [12], Holte discovered the aptly named Amazing matrices from studying the probability of 'carries' when adding together large numbers. Holte found these matrices give rise to several interesting properties including the appearance of the Eulerian and Stirling numbers discussed earlier in the eigenvectors of these matrices. The following definition for these matrices was given by Holte in [12].

**Definition 10.1.** The $n \times n$ *Amazing Matrix* $P_b$ is defined such that for $n$ random numbers in base $b$, the $i,j$th entry of $P_b$ is given by the probability of carrying $j$ to the column of addition given that you carried $i$ into the current column. It was proven [12] that each entry is given by the formula:

$$p_{ij} = b^{-n} \sum_{r=0}^{j-\lfloor i/b \rfloor} (-1)^r \binom{n+1}{r} \binom{n-1-i+(j+1-r)b}{n}$$

Diaconis and Fulman [13] gave an alternative definition in terms of the shuffling of cards. They also conjectured these matrices to be TNN and proved the $2 \times 2$ minors must always be positive.

**Conjecture 10.2** (Diaconis-Fulman [13]). *Any Amazing matrix $P$ is totally non-negative.*

This could be proven by showing the following:

1. $P$ is cross-symmetric.

2. $P$ can be factorised as in Lemma 9.4 to a positive diagonal matrix using only C-S Neville matrices with no row swaps required (i.e. no unexpected zeroes occur).

(1) was proven by Holte in [12] Theorem 2 and hence the conjecture reduces to (2) which as of yet has not been shown. It has, however, been shown that all $2 \times 2$ amazing matrices are TNN [13], total non-negativity of the $3 \times 3$ and $4 \times 4$ cases was obtained in an earlier student project by Natasha Strokes. I have also shown that all $5 \times 5$ cases are TNN by factorising the matrix $M = b^n P_b$ such that $M = L_1 L_2 L_3 L_4 D U_4 U_3 U_2 U_1$ where these matrices are shown on

the next page. The matrices $L_i$ and $U_i$ are the product of elementary Neville matrices given by the elements on the anti-diagonal. For example,

$$L_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{b-4}{b+1} & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{b-3}{b+2} & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & \frac{b-2}{b+3} & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{b-1}{b+4} & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

these anti-diagonals give the weights of the planar network for which the Amazing matrix is the weight matrix. This factorisation shows TNN for $b > 4$ since all terms in the factorisation are positive. For $b \in \{1, 2, 3, 4\}$ the matrices have been explicitly checked and hence all $5 \times 5$ Amazing matrices are TNN.

### Acknowledgements

I would like to thank my supervisor Prof. Tom Lenagan for agreeing to and providing support throughout this project. This work was funded by University of Edinburgh School of Mathematics.

$$L_1L_2L_3L_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ \frac{b-1}{b+4} & 1 & 0 & 0 & 0 \\ \frac{(b-2)(b-1)}{(b+3)(b+4)} & \frac{b-2}{b+3} & 1 & 0 & 0 \\ \frac{(b-3)(b-2)(b-1)}{(b+2)(b+3)(b+4)} & \frac{(b-3)(b-2)}{(b+2)(b+3)} & \frac{b-3}{b+2} & 1 & 0 \\ \frac{(b-4)(b-3)(b-2)(b-1)}{(b+1)(b+2)(b+3)(b+4)} & \frac{(b-4)(b-3)(b-2)}{(b+1)(b+2)(b+3)} & \frac{(b-4)(b-3)}{(b+1)(b+2)} & \frac{b-4}{b+1} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & \frac{(b+4)(2b-1)}{(b+3)(2b+3)} & 1 & 0 & 0 \\ 0 & \frac{(b-1)(b+4)(2b-1)}{(b+1)(b+2)(2b+3)} & \frac{(b-1)(b+3)}{(b+1)(b+2)} & 1 & 0 \\ 0 & \frac{(b-1)(b+4)(2b-3)(2b-1)}{(b+1)^2(2b+1)(2b+3)} & \frac{(b-1)(b+3)(2b-3)}{(b+1)^2(2b+1)} & \frac{(b+2)(2b-3)}{(b+1)(2b+1)} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{(3b-1)(2b+3)(b+3)}{2(3b+2)(b+1)(b+2)} & 1 & 0 \\ 0 & 0 & \frac{(3b-1)(3b-2)(2b+3)(b+3)}{(3b+1)(3b+2)(2b+1)(b+1)} & \frac{2(3b-2)(b+2)}{(3b+1)(2b+1)} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{2(4b-1)(3b+2)(b+2)}{(4b+1)(3b+1)(2b+1)} & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} \frac{b(b+1)(b+2)(b+3)(b+4)}{120} & 0 & 0 & 0 & 0 \\ 0 & \frac{b^2(b+1)(2b+1)(2b+3)}{6(b+4)} & 0 & 0 & 0 \\ 0 & 0 & \frac{b^3(3b+1)(3b+2)}{(2b+3)(b+3)} & 0 & 0 \\ 0 & 0 & 0 & \frac{b^4(24b+6)}{(3b+2)(b+1)(b+2)} & 0 \\ 0 & 0 & 0 & 0 & \frac{120b^5}{(4b+1)(3b+1)(2b+1)(b+1)} \end{bmatrix}$$

$$U_4U_3 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \frac{(b+2)(3b-1)(3b+2)(4b-1)}{4(2b+1)(4b+1)(15b^2-4)} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & \frac{4(b+3)(2b-1)(2b+1)(2b+3)\left(15b^2-4\right)}{3(b+1)(3b+1)(3b+2)(55b^3+89b^2-30b-36)} & \frac{(3b-1)(3b-2)(2b-1)(2b+3)(b-2)(b+3)}{3(3b+1)(3b+2)(b+1)(55b^3-89b^2-30b+36)} \\ 0 & 0 & 0 & 1 & \frac{-(3b-1)(3b-2)(b-2)(55b^3+89b^2-30b-36)}{-4(2b+1)(15b^2-4)(55b^3-89b^2-30b+36)} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$U_2U_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{3(b+4)(55b^3+89b^2-30b-36)}{2(b+2)(2b+1)(2b+3)(13b+24)} & \frac{(b-1)(b+4)(55b^3-89b^2-30b+36)}{2(b+1)(2b+1)(2b+3)(11b^2-24)} & \frac{(2b-1)(2b-3)(b-1)(b-3)(b+4)}{2(b+1)^2(2b+1)(2b+3)(13b-24)} \\ 0 & 0 & 1 & \frac{(b-1)(b+2)(13b+24)\left(55b^3-89b^2-30b+36\right)}{3(b+1)(11b^2-24)(55b^3+89b^2-30b-36)} & \frac{(13b+24)(2b-1)(2b-3)(b-1)(b+2)(b-3)}{3(b+1)^2(13b-24)(55b^3+89b^2-30b-36)} \\ 0 & 0 & 0 & 1 & \frac{(2b-1)(2b-3)(b-3)(11b^2-24)}{(13b-24)(b+1)(55b^3-89b^2-30b+36)} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & \frac{2(b-1)(13b+24)}{(b+3)(b+4)} & \frac{6(b-1)\left(11b^2-24\right)}{(b+2)(b+3)(b+4)} & \frac{2(b-2)(b-1)(13b-24)}{(b+2)(b+3)(b+4)} & \frac{(b-4)(b-3)(b-2)(b-1)}{(b+1)(b+2)(b+3)(b+4)} \\ 0 & 1 & \frac{3\left(11b^2-24\right)}{(b+2)(13b+24)} & \frac{(b-2)(13b-24)}{(b+2)(13b+24)} & \frac{(b-4)(b-3)(b-2)}{2(b+1)(b+2)(13b+24)} \\ 0 & 0 & 1 & \frac{(b-2)(13b-24)}{3(11b^2-24)} & \frac{(b-4)(b-3)(b-2)}{6(b+1)(11b^2-24)} \\ 0 & 0 & 0 & 1 & \frac{(b-4)(b-3)}{2(b+1)(13b-24)} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

# Appendix

I used Python with the Sympy module to program the algorithms covered in this report. The implementation is shown below.

```python
'''Imports sympy module'''
import sympy as sym

'''Deletion Algorithm'''
def DDA(M):
    N=M.copy()
    '''Loops through [0,n-1] matrix elements backwards'''
    for j in reversed(range(M.shape[0])):
        for b in reversed(range(M.shape[1])):
            M=N.copy()
            '''Carries out function'''
            for i in range(j):
                for a in range(b):
                    if(N[j,b] != 0):
                        '''Carries out function operation'''
                        N[i,a] = sym.simplify(M[i,a] - \
                                 M[i,b]/M[j,b]*M[j,a])
    return N

'''Checks if the deletion algorithm output satisfies
the required conditions'''
def IsTnnAfterDDA(N):
    for i in range(N.shape[0]):
        for j in range(N.shape[0]):
            '''Not TNN if an element is < 0'''
            if(N[i,j] < 0):
                return False, i,j
            '''Checks Cauchy diagram condition'''
            if(N[i,j] == 0):
                if(N[0:i+1,j].norm() != 0 and\
                   N[i,0:j+1].norm() != 0):
                    return False
    return True

'''Carries out Neville elminiation'''
def Neville(V):
    '''Creates initial matrices and places them in arrays'''
    n = V.shape[1]
    U = [V]
    L = [sym.eye(n)]
    D = [sym.eye(n)]
    F = []

    '''Loops through columns of matrix'''
    for i in range(n):
    '''Creates sub-diagonal matrix which represent
    the subtraction of rows'''
        F.append(sym.eye(n))
```

```
        for j in range(i+1,n):
            if(U[-1][j-1,i]==0):
                continue
        '''Creates Neville matrix'''
            F[-1][j,j-1] = sym.simplify(-U[-1][j,i]/U[-1][j-1,i])
    '''Appends new upper and lower matrices'''
        U.append(F[-1]*U[-1])
        L.append(F[-1].inv())
        E = sym.eye(n)
        E[i,i] = U[-1][i,i]
        D.append(D[-1]*E)
        '''Creates diagonal matrix so the triangular matrices
    have 1 on the diagonal'''
        U[-1] = E.inv()*U[-1]
    return L, D, U, F

'''Cross-Symmetric Neville Elimination'''
def CrossSymNeville(V):
    '''Creates initial matrices and places them in arrays'''
    n = V.shape[0]
    F = []
    L = []
    U = [V]
    for i in range(n):
        for j in reversed(range(i,n)):
            if (i == j or (U[-1][j-1,i] == 0 and U[-1][j,i] == 0)):
                continue
            if (U[-1][j-1,i] == 0):
                if (j == n/2): '''Step 4'''
                    F.append(sym.eye(n))
                    F[-1][j-1,j-1] = 0
                    F[-1][j,j] = 0
                    F[-1][j,j-1] = 1
                    F[-1][j-1,j] = 1
                    U.append(F[-1]*U[-1])
                    L.append(F[-1].inv())
                    continue
                elif (j != (n-1)/2): '''Step 5'''
                    F.append(sym.eye(n))
                    F[-1][j-1,j-1] = 0
                    F[-1][j,j] = 0
                    F[-1][j,j-1] = 1
                    F[-1][j-1,j] = 1

                    F[-1][n-j,n-j] = 0
                    F[-1][n-1-j,n-1-j] = 0
                    F[-1][n-1-j,n-j] = 1
                    F[-1][n-j,n-1-j] = 1

                    U.append(F[-1]*U[-1])
                    L.append(F[-1].inv())
                    continue
                else: '''Step 6'''
```

```
                    for k in reversed(range(j-1)):
                        if (U[-1][k,i] != 0):
                            while (U[-1][j-1,i] == 0):
                                F.append(sym.eye(n))
                                print(k,k+1)
                                F[-1][k+1,k] = 1
                                F[-1][n-2-k,n-1-k] = 1
                                U.append(F[-1]*U[-1])
                                L.append(F[-1].inv())
                                k = k+1
                            break
                    continue
            '''Neville Move'''
            F.append(sym.eye(n))
            F[-1][j,j-1] = -U[-1][j,i]/U[-1][j-1,i]
            F[-1][n-1-j,n-1-j+1] = -U[-1][j,i]/U[-1][j-1,i]
            U.append(F[-1]*U[-1])
            L.append(F[-1].inv())
    return F,U,L
```

# References


[1] B. Lindström *On the Vector Representations of Induced Matroids*. Bulletin London Math. Soc. **5** (1973) 85-90

[2] M. Aigner, G.M. Ziegler. *Proofs from the Book, sixth edition*. Springer-Verlag, Berlin, Including illustrations by Karl H. Hofmann, (2018), Chapter 32 (Lattice paths and determinants)

[3] M. Gasca and J.M. Peña *Total positivity and Neville elimination*. Lin. Alg. and Appl. **165**, (1992), 25–44.

[4] S. Fomin and A. Zelevinsky *Total Positivity: Tests and Parametrizations*. Math Intelligencer **22** (2000), no. 1, 23-33.

[5] A. Pinkus *Totally Positive Matrices*. Cambridge Tracts in Mathematics **181**, Cambridge University Press, Cambridge, 2010

[6] F. Brenti *The applications of total positivity to combinatorics, and conversely*. In: Gasca, M., Micchelli, C.A. (eds.) Total Positivity and its Applications. Kluwer, Dordrecht (1996)

[7] N. J. A. Sloane, (ed.). *Sequence A009766 (Catalan's triangle)*. The On-Line Encyclopedia of Integer Sequences. OEIS Foundation.

[8] Wang, Charles and Yi, Wang. *Total positivity of Catalan triangle*. Discrete Mathematics. 338. (2015)

[9] K.R. Goodearl, S. Launois, and T.H. Lenagan. *Totally nonnegative cells and matrix Poisson varieties*. Advances in Mathematics 226 (2011), 779-826.

[10] K.R. Goodearl and T.H. Lenagan *LU decomposition of nonnegative matrices*. Linear Algebra and its Applications, 436 (2012), 2554-2566

[11] T.K. Petersen. *Eulerian Numbers*. Birkhäuser Advanced Texts, Birkhäuser, (2015), 6-8

[12] J.M. Holte *Carries, Combinatorics and an Amazing Matrix*. American Mathematical Monthly **104**:2 (1997) 138-149

[13] P. Diaconis and J. Fulman *Carries, Shuffling, and an Amazing Matrix*. American Mathematical Monthly **116**:9 (2009) 788-803