

Alternative proof of the average number of occupied corners

May 22, 2015

We present here the initial proof of the Theorem 5, *i.e* in average there is one occupied corner per tree-like tableau in \mathcal{T}_n .

Tree-like tableaux

In this first part we recall basic notions and tools about tree-like tableaux. All the details can be found in the article [ABN13].

Definition 1 (Tree-like tableau). *A tree-like tableau is a filling of Young diagram with points inside some cells, with three rules:*

1. *the top left cell has a point called the root point;*
2. *for each non root point, there is a point above it in the same column or to its left in the same row, but not both at the same time;*
3. *there is no empty row or column.*

The *size* of a tree-like tableau is its number of points. We denote by \mathcal{T}_n the set of the tree-like tableaux of size n . In a tree-like tableau, we call *border edges*, the edges of the Southeast border. A tree-like tableau of size n has $n + 1$ border edges, we index them from 1 to $n + 1$ as it is done in Figure 1. The border edge numbered by i in a tree-like tableau T is denoted by $e_i(T)$. In the rest of the article, when there is no ambiguity, the tableau T might be omitted in all the notations.

The set of tree-like tableaux has an inductive structure given by an insertion algorithm called *Insertpoint* which constructs a tree-like tableau of size $n + 1$ from a tree-like tableau of size n and the choice of one of its border edges. We briefly present the algorithm in order for the article to be self-contained. A more detailed presentation is given in [ABN13].

The *special point* plays an important role in *Insertpoint*, it is the right-most point among those at the bottom of a column. We denote by $sp(T)$ the index of the horizontal border edge under the special point of a tree-like tableau T . In the figures of this article, the special point might be indicated by a square

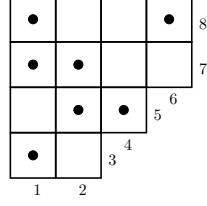


Figure 1: A tree-like tableau of size 7.

around it. The second notion we need is the *ribbon*, it is a connected set of empty cells (with respect to adjacency) containing no 2×2 squares. Now we can introduce the insertion algorithm.

Definition 2 (Insertpoint). *Let T be a tree-like tableau of size n and e_i one of its border edges. First, if e_i is horizontal (resp. vertical) we insert a row (resp. column) of empty cells, just below (resp. to the right of) e_i , starting from the left (resp. top) border of T and ending below (resp. to the right of) e_i . Moreover we put a point in the right-most (resp. bottom) cell. We obtain a tree-like tableau T' of size $n + 1$. Then, depending on the relative ordering of i and $sp(T)$, we define $Insertpoint(T, e_i)$, denoted also by $Insertpoint(T, i)$.*

1. If $i \geq sp$, then $Insertpoint(T, e_i) = T'$;
2. otherwise, we add to T' a ribbon along the border, from the new point to the special point of T . This new tree-like tableau will be $Insertpoint(T, e_i)$.

An example of the two possible insertions is given in Figure 2 where the cells of the new row/column are shaded and the cells of the ribbon are marked by crosses.

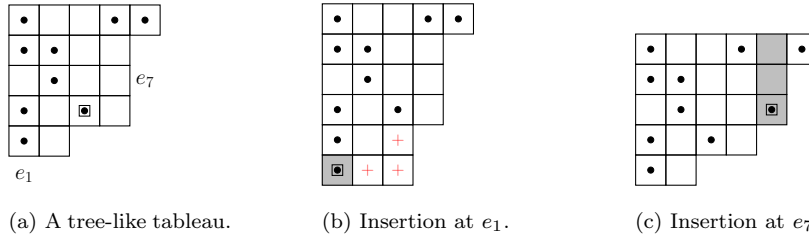


Figure 2: Insertion algorithm applied to T at e_1 and e_7 .

Remark 3. *We notice that the new point is the special point of the new tableau. In addition, the index of the horizontal border edge under the new point is equal to the one we chose during the algorithm, in other words:*

$$sp(Insertpoint(T, i)) = i.$$

tree-like tableaux. But it is difficult to control $oc(Insertpoint(T, i))$ when e_i is to the Southwest of the special point. Indeed, the ribbon kills the occupied corners between them. To get around this issue, the idea is to consider a second algorithm such that for each choice of border edge, each occupied corner is killed once, only by the first one or only by the second one. We denote by tT the tree-like tableau obtained from T by an axial symmetry with respect to the main diagonal of the Young diagram of T . We define the second algorithm, $Insertpoint_t$, as follows:

$$Insertpoint_t(T, i) = {}^t Insertpoint({}^tT, (n + 2 - i)).$$

In the same way there is a special point for $Insertpoint$, there is one also for $Insertpoint_t$.

Definition 6 (Conjugate special point). *The conjugate special point of a tree-like tableau T is the point corresponding to the special point of tT . It is also the lowest point among those at the right-most cell of a row. We denote by $\bar{sp}(T)$ the index of the border edge to the right of the conjugate special point.*

In the following figures, the conjugate special point might be indicated by a diamond. We have the following relation, $\bar{sp}(T) = n + 2 - sp({}^tT)$ as we can see in Figure 5. The first parts of both algorithms are the same. For the second

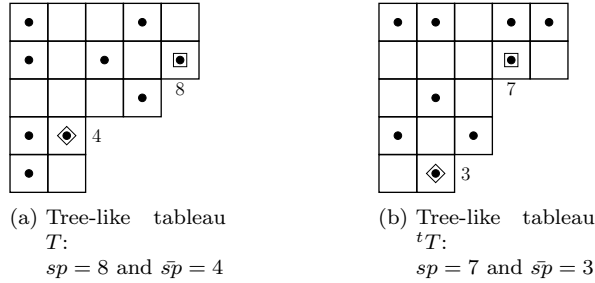


Figure 5: Relation between the conjugate special point and the special point

part, in the case of $Insertpoint_t$, we add a ribbon between the new point and the conjugate special point if, and only if, the new point is to the Northeast of the conjugate special point, *i.e* $i > \bar{sp}$. Figure 6 gives us an example. Using the previous notations, we have that:

$$Insertpoint_t(\{T\} \times \llbracket 1, n + 1 \rrbracket) = {}^t [{}^tT].$$

Lemma 7. *Let $T \in \mathcal{T}_n$, we have:*

$$oc([T]) + oc({}^t[{}^tT]) = (n - 1)oc(T) + (n + 1 - sp(T) + 1) + \bar{sp}(T). \quad (1)$$

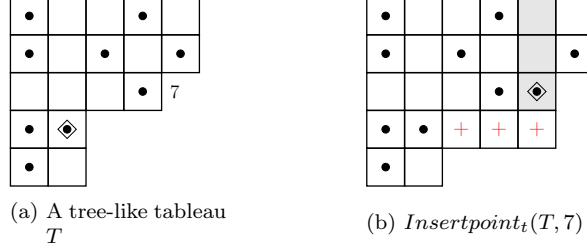


Figure 6: Example of $Insertpoint_t$ with a ribbon.

Proof. Suppose that T does not have any occupied corners, i.e. $oc(T) = 0$. Then $Insertpoint(T, i)$ has an occupied corner if no ribbon is added during the algorithm, else it has no occupied corner. In other words $oc(Insertpoint(T, i)) = 1$ if $sp \leq i \leq n+1$ and 0 otherwise. Similarly, $oc(Insertpoint_t(T, i)) = 1$ if $1 \leq i \leq \bar{sp}$ and 0 otherwise. Hence we have the formula

$$oc([T]) + oc({}^t[T]) = (n+1 - sp(T) + 1) + \bar{sp}(T)$$

which match with Equation (1).

From now on, let us suppose that $oc(T) \geq 1$. The points of occupied corners of T are at the right-most cell of a row and at the bottom of a column, so all the occupied corners of T are at the Southwest of the special point and to the Northeast of the conjugate special point. First, if the conjugate special point and the special point are the same point, then they correspond to the only occupied corner of T , thus $oc(T) = 1$ and $\bar{sp} = sp + 1$. We observe that for any choice of i , $oc(Insertpoint(T, i)) + oc(Insertpoint_t(T, i)) = 2$, indeed:

- $1 \leq i < sp$: a ribbon is added to obtain $Insertpoint(T, i)$, hence the new point is not in a corner and we no longer have the initial corner. Thus $oc(Insertpoint(T, i)) = 0$. Conversely, no ribbon is added in the case of $Insertpoint_t$, therefore we have a new occupied corner while keeping the initial one. As a result, $oc(Insertpoint_t(T, i)) = 2$.
- $i = sp$ or $i = sp + 1$: the same happens for both algorithm and for both choices of i , namely, the new occupied corner kills the initial one. Hence $oc(Insertpoint_t(T, i))$ and $oc(Insertpoint(T, i))$ are equal to 1.
- $sp + 1 < i \leq n + 1$: the symmetric case of $0 \leq i < sp$ happens here, i.e. $oc(Insertpoint(T, i)) = 2$ and $oc(Insertpoint_t(T, i)) = 0$.

Finally,

$$oc([T]) + oc({}^t[T]) = 2(n+1),$$

as $sp + 1 = \bar{sp}$, we obtain again Equation (1).

Suppose now that the two special points are different, since we have at least one occupied corner, $\bar{sp} < sp$. We study the case where both special points are not in a corner. We have two cases:

- $1 \leq i \leq \bar{sp}$ or $sp \leq i \leq n+1$: in both cases one of the two algorithm gives a tree-like tableau without occupied corner and the other gives a tree-like tableau with $oc(T) + 1$ corners. Hence, we have $oc(Insertpoint(T, i)) + oc(Insertpoint_t(T, i)) = oc(T) + 1$.
- $\bar{sp} < i < sp$: suppose e_i is not the edge of an occupied corner, let k be the number of occupied corners Southwest to e_i . Then $oc(Insertpoint(T, i)) = k$ since the ribbon kills the occupied corners Northeast to e_i and also the new one we could have obtained from the insertion of the new row or column. By the same argument, $oc(Insertpoint_t(T, i)) = oc(T) - k$ so the sum is equal to $oc(T)$. Conversely, if we suppose that e_i is the horizontal or the vertical border edge of an occupied corner, this occupied corner is killed during $Insertpoint(T, i)$ and $Insertpoint_t(T, i)$, hence $oc(Insertpoint(T, i)) + oc(Insertpoint_t(T, i)) = oc(T) - 1$.

Summing over all positions, we get

$$oc([T]) + oc({}^t[T]) = (n+1)oc(T) + \bar{sp}(T) + (n+1 - sp(T) + 1) - 2oc(T),$$

which corresponds to Equation (1).

To end the proof, we need to study the case where the special points can be in a corner. For example, we study the case where the special point is in a corner but the conjugate special point is not. The results for $1 \leq i < sp$ and $sp+1 < i \leq n+1$ are the same keeping in mind that there are only $oc(T) - 1$ occupied corners in $\bar{sp} < i < sp$. The two remaining choices for i give us $oc(Insertpoint(T, i)) + oc(Insertpoint_t(T, i)) = oc(T)$, hence we obtain:

$$oc([T]) + oc({}^t[T]) = (n+1)oc(T) + \bar{sp} - 2(oc(T) - 1) + (n+1 - (sp+2) + 1)$$

which corresponds to Equation (1). The remaining two cases give us the same result. \square

Proof of Theorem 5. Using that $sp(Insertpoint(T, i)) = i$ and the fact that $\mathcal{T}_n = \bigsqcup_{T \in \mathcal{T}_{n-1}} [T]$, for $j \in \llbracket 1, n \rrbracket$ we have $|\{T \in \mathcal{T}_n, sp(T) = j\}| = (n-1)!$. Since $\bar{sp} = n+2 - sp$, we get

$$\sum_{T \in \mathcal{T}_n} \bar{sp}(T) - sp(T) = (n+2)n! - 2 \times \frac{n(n+1)}{2} \times (n-1)! = n!.$$

Therefore, if we sum Equation (1) over \mathcal{T}_n , we obtain:

$$2oc(\mathcal{T}_{n+1}) = (n-1)oc(\mathcal{T}_n) + (n+3)n!. \quad (2)$$

Finally, as the sequence $(n!)$ satisfies Equation (2) and $oc(\mathcal{T}_1) = 1!$, then $oc(\mathcal{T}_n) = n!$ for any positive integer n . \square

References

- [ABN13] Jean-Christophe Aval, Adrien Boussicault, and Philippe Nadeau.
Tree-like tableaux. *Electron. J. Combin.*, 20(4):Paper 34, 24, 2013.