An explicit formula for ndinv, a new statistic for two-shuffle parking functions

Angela Hicks and Yeonkyung Kim†

Mathematics Department, University of California San Diego, La Jolla, USA

Abstract. In a recent paper, Duane, Garsia, and Zabrocki introduced a new statistic, "ndinv", on a family of parking functions. The definition was guided by a recursion satisfied by the polynomial $\langle \Delta_{h_m} C_{p_1} C_{p_2} \ldots C_{p_k} 1, e_n \rangle$, for Δ_{h_m} a Macdonald eigenoperator, C_{p_i} a modified Hall-Littlewood operator and $(p_1, p_2, \ldots, p_k)$ a composition of n. Using their new statistics, they are able to give a new interpretation for the polynomial $\langle \nabla e_n, h_j h_{n-j} \rangle$ as a q,t numerator of parking functions by area and ndinv. We recall that in the shuffle conjecture, parking functions are q,t enumerated by area and diagonal inversion number (dinv). Since their definition is recursive, they pose the problem of obtaining a non recursive definition. We solved this problem by giving an explicit formula for ndinv similar to the classical definition of dinv. In this paper, we describe the work we did to construct this formula and to prove that the resulting ndinv is the same as the one recursively defined by Duane, Garsia, and Zabrocki.

Résumé. Dans un travail récent Duane, Garsia et Zabrocki ont introduit une nouvelle statistique, "ndinv" pour une famille de Fonctions Parking. Ce "ndinv" découle d'une récurrence satisfaite par le polynôme $\langle \Delta_{h_m} C_{p_1} C_{p_2} \cdots C_{p_k} 1 \,,\, e_n \rangle$, oú Δ_{h_m} est un opérateur linéaire avec fonctions propres les polynômes de Macdonald, les C_{p_i} sont des opérateurs de Hall-Littlewood modifiés et $(p_1, p_2, \ldots, p_n)$ est un vecteur à composantes entières positives. Par moyen de cette statistique, ils ont réussi à donner une nouvelle interpretation combinatoire au polynôme $\langle \nabla e_n \,,\, h_j h_{n-j} \rangle$ on remplaçant "dinv" par "ndinv". Rappelons nous que la conjecture "Shuffle" exprime ce même polynôme comme somme pondérée de Fonctions Parking avec poids t à la "aire" est q au 'dinv". Puisque il donnent une definition récursive du "ndinv"' il posent le problème de l'obtenir d'une façon directe. On résout se problème en donnant une formule explicite qui permet de calculer directement le "ndinv" à la manière de la formule classique du "dinv". Dans cet article on décrit le travail qu'on a fait pour construir cette formule et on démontre que nôtre formule donne le même "ndinv" récursivement construit par Duane, Garsia et Zabrocki.

Keywords: parking functions, diagonal inversions, dinv, shuffle conjecture, Hall-Littlewood polynomials

1 Introduction

We start by introducing parking functions, fixing the notation and recalling some auxiliary results.

†Both authors' work supported by NSF grant.

1365–8050 © 2012 Discrete Mathematics and Theoretical Computer Science (DMTCS), Nancy, France

1.1 Parking Functions

Definition 1.1 (Parking Function). We call a two line array

$$PF = \begin{bmatrix} c_1 & c_2 & \dots & c_n \\ d_1 & d_2 & \dots & d_n \end{bmatrix}$$

a parking function if

- The first row is a permutation of $\{1, 2, \dots, n\}$.
- $d_1 = 0$ and $d_i \leq d_{i-1} + 1$ for $i > 1$,
- if $d_i = d_{i-1} + 1$, then $c_i > c_{i-1}$

We consider the numbers in the first row of the array as cars and we say car c_i is on the d_i^{th} diagonal, with the 0^{th} diagonal being referred to as the main diagonal. We may represent parking functions in an n by n lattice square using a Dyck path and labels in the lattice cells adjacent to its north steps. A Dyck path can be defined using the elements in the second row, as d_i represents the number of full squares between the Dyck path and the main diagonal on the i^{th} row from the bottom. Then, car c_i is placed directly to the right of the i^{th} north step of the Dyck path. (See Figure 1.)

In this paper, we will work with "Two-Shuffle Parking Functions", which is a subset of parking functions.

Definition 1.2. (Reading Word). The reading word of a parking function $(\text{word}(PF))$ is the permutation of $\{1, 2, \dots, n\}$, reading cars from the highest diagonal to the main diagonal. When we read cars on the same diagonal, we record cars from northeast to southwest.

Example 1.1. The reading word of the parking function in Figure 1 is $(1, 3, 4, 5, 2)$.

Recall that a permutation σ is a shuffle of $(1, \dots, m)$ and $(m+1, \dots, m+n)$ when if $i_1 < i_2 \leq m$ or $m < i_1 < i_2$, then i_1 occurs before i_2 in σ.

Definition 1.3 (Two-Shuffle Parking Functions). A parking function PF is a two-shuffle parking function when, for two positive integers m, n, it satisfies following conditions:

1. $\text{word}(PF)$ is a shuffle of $(1, \dots, m)$ and $(m+1, \dots, m+n)$;
2. $c_{n+m} > m$; and
3. $d_{n+m} = 0$.

Example 1.2. The parking function in Figure 1 is a two-shuffle parking function for $m = 2$.

In this paper, we sometimes call a car c a "big car" if $c > m$ and a "small car" if $c \leq m$. We may denote a small car as "c_s" and a big car as "c_b".

Definition 1.4 (Composition). Define the sequence $(f_1, f_2, \cdots, f_k)$ such that c_{f_i} is a big car and $d_{f_i} = 0$ for each i and $f_1 < f_2 < \cdots < f_k$. Then we say a car c_j is in the first part if $j \leq f_1$ and otherwise in the i^{th} part if $f_{i-1} < j \leq f_i$. Then we can define the "composition of PF" $(\text{comp}(PF))$ as the sequence $(p_1, p_2, \dots, p_k)$, where p_i counts the number of big cars in the i^{th} part.

$$PF = \begin{bmatrix} 2 & 4 & 3 & 1 & 5 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

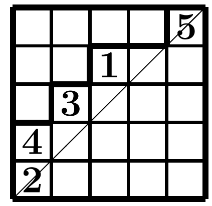

Fig. 1: PF as shown in the 5 x 5 lattice.

Example 1.3.

$$PF = \begin{bmatrix} 2 & 5 & 4 & 7 & 1 & 3 & 6 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

then for $m = 2$ we have

$$\mathrm{comp}(PF) = (3, 2).$$

There are two parking function statistics that have been extensively studied in the literature.

$$\mathrm{area}(PF) = \sum_i d_i$$

and

$$\mathrm{dinv}(PF) = \sum_{i<j} \chi(d_i = d_j \text{ and } c_i < c_j) + \chi(d_i = d_j + 1 \text{ and } c_i > c_j),$$

where χ gives the truth function. These two statistics are of interest because they play an essential role in a variety of results tying parking functions to the theory of Macdonald polynomials and the representation theory of the symmetric group; they are also core ingredients in the formulation of the "Shuffle Conjecture".

1.2 Some auxiliary facts and conjectures.

The shuffle conjecture as given in [Haglund et al.(2005)] states in particular that

$$\nabla e_n = \Delta_{e_n} e_n = \sum_{PF} t^{\mathrm{area}(PF)} q^{\mathrm{dinv}(PF)} Q_{\mathrm{ides}(PF)},$$

where Q is the Gessel quasi-symmetric function, $\mathrm{ides}(PF) = \mathrm{des}((\mathrm{word}(PF))^{-1})$, and Δ_f is the linear operator defined by setting for $\tilde{H}_\mu[X; q, t]$ the modified Macdonald basis [Garsia and Haiman(1996)]:

$$\Delta_f \tilde{H}_\mu[X; q, t] = f\left[\sum_{(i,j)\in\mu} t^{i-1} q^{j-1}\right] \tilde{H}_\mu[X; q, t].$$

A number of authors have given related expressions in terms of the area and dinv of particular families of parking functions. In [Haglund(2004)], Haglund proved the identity

$$\langle \Delta_{h_m} E_{n,k}, e_n \rangle = \sum_{F(n,k,m)} t^{\mathrm{area}(PF)} q^{\mathrm{dinv}(PF)}, \tag{1}$$

where $F(n,k,m)$ denotes the family of parking functions that start with a big car, have m small cars and n big cars, k of which are on the main diagonal and whose word is a shuffle of $1,2,\ldots,m$ with $m+1,m+2,\ldots,m+n$. Note that here $E_{n,k}$ are the symmetric functions introduced by Garsia-Haglund in [Garsia and Haglund(2002)] with the property that

$$E_{n,1}+E_{n,2}+\cdots+E_{n,n}=e_n.$$

Recent work in [Haglund et al.(2011)] used modified Hall-Littlewood operators (represented here as C_a) to give a refinement of the shuffle conjecture. The following identity, proved in [Haglund et al.(2011)]

$$E_{n,k}=\sum_{(p_1,p_2,\ldots,p_k)\models n} C_{p_1}C_{p_2}\ldots C_{p_k}1$$

suggested Duane, Garsia, and Zabrocki (in [Duane et al.(2012)]), that the polynomials

$$\langle \Delta_{h_m} C_{p_1}C_{p_2}\ldots C_{p_k}1, e_n\rangle,$$

might yield a refinement of (1). In particular they found that

$$\langle \Delta_{h_m} C_{p_1}\ldots C_{p_k}1, e_n\rangle|_{q=1}=\sum_{\substack{PF\,\in\, F(n,\,k,\,m)\\ \mathrm{comp}(PF)\,=\,(p_1,\,p_2,\,\ldots,\,p_k)}} t^{\mathrm{area}(PF)}.$$

In an effort to obtain a combinatorial interpretation of the left hand side without the restriction "$q=1$", they were led to introduce a new statistic, which they called "ndinv". In [Duane et al.(2012)] they only obtain an algorithmic construction of ndinv based on a recursion satisfied by the polynomial $\langle \Delta_{h_m} C_{p_1}\ldots C_{p_k}1, e_n\rangle$, and pose the problem of finding a non recursive definition. In this paper, we describe the work that we did to solve this problem.

2 An explicit formula for ndinv.

Our point of departure is the modified two line representation of a parking function, obtained in "Stage 1" of [Duane et al.(2012)]. We sometimes find it convenient to consider a parking function as a sequence of "dominoes" $\begin{bmatrix} c_i \\ d_i \end{bmatrix}$.

Procedure 2.1. Beginning with a parking function PF:

1. Working from left to right, if c_i is small, then shift $\begin{bmatrix} c_i \\ d_i \end{bmatrix}$ to the left past d_i big cars.
2. For every big car c_j, count the number of small cars which shifted past it in the previous step. Increase d_j by this number.

Use this modified parking function to define the first two lines of the following three line array.

$$\Psi(PF)=\begin{bmatrix} c_1^\Psi & c_2^\Psi & \ldots & c_{n+m}^\Psi \\ d_1^\Psi & d_2^\Psi & \ldots & d_{n+m}^\Psi \\ r_1^\Psi & r_2^\Psi & \ldots & r_{n+m}^\Psi \end{bmatrix}.$$

Next, in a departure from Duane, Garsia, and Zabrocki's work we assign to each car c_i^Ψ an explicit statistic r_i^Ψ by setting:

$$r_i^\Psi = \begin{cases} 1 & i = 1 \\ r_{i-1}^\Psi + 1 & c_{i-1}^\Psi \leq m \text{ and } i > 1 \\ d_{i-1}^\Psi + 1 & c_{i-1}^\Psi > m \text{ and } i > 1 \end{cases}. \tag{2}$$

Example 2.1. Again let $m = 3$ and

$$PF = \begin{bmatrix} 3 & 4 & 1 & 6 & 2 & 5 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

Then

$$\Psi(PF) = \begin{bmatrix} 3 & 1 & 4 & 6 & 2 & 5 \\ 0 & 1 & 2 & 0 & 0 & 0 \\ 1 & 2 & 3 & 3 & 1 & 2 \end{bmatrix}.$$

This three line array allows us to give an explicit construction for ndinv.

Definition 2.1. If m gives the number of small cars,

$$\mathrm{ndinv}(PF) := \sum_{c_b^\Psi > m} \sum_{c_s^\Psi \leq m} \Big(\chi(b < s)\chi(d_b^\Psi \leq r_s^\Psi < r_b^\Psi) + \chi(b > s)\chi(d_b^\Psi < r_s^\Psi \leq r_b^\Psi) \Big) - m \tag{3}$$

Mirroring previous conventions for dinv, we will say that:

Definition 2.2 (Diagonal Inversion). A big car c_b and a small car c_s form a diagonal inversion in $\Psi(PF)$ exactly when they contribute to the sum in the above definition.

Example 2.2. As in Example 1.3, $m = 3$ and

$$PF = \begin{bmatrix} 3 & 4 & 1 & 6 & 2 & 5 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

and

$$\Psi(PF) = \begin{bmatrix} 3 & 1 & 4 & 6 & 2 & 5 \\ 0 & 1 & 2 & 0 & 0 & 0 \\ 1 & 2 & 3 & 3 & 1 & 2 \end{bmatrix}$$

Then there are six pairs of cars which form diagonal inversions, namely (c_1^Ψ, c_4^Ψ), (c_1^Ψ, c_6^Ψ), (c_2^Ψ, c_4^Ψ), (c_2^Ψ, c_6^Ψ), (c_5^Ψ, c_4^Ψ), and (c_5^Ψ, c_6^Ψ).
Since there are 3 small cars, $\mathrm{ndinv}(PF) = 6 - 3 = 3$.

We should mention that with this definition of ndinv we have

Theorem 2.1. *For any $(p_1, p_2, \ldots, p_k) \models n$ and our definition of ndinv we have*

$$\langle \Delta_{h_m} C_{p_1} \ldots C_{p_k} 1, e_n \rangle = \sum_{\substack{PF \text{ an m,n two-shuffle parking function} \\ \mathrm{comp}(PF) = (p_1, \ldots, p_k)}} t^{\mathrm{area}(PF)} q^{\mathrm{ndinv}(PF)}. \tag{4}$$

3 A recursion satisfied by ndinv.

Since $\langle \Delta_{h_0} C_1 1, e_1 \rangle = 1$, to be consistent we must set

$$\text{ndinv}\left(\begin{bmatrix}1\\0\end{bmatrix}\right) = 0.$$

In [Duane et al.(2012)], Duane, Garsia, and Zabrocki prove the following recursion:

$$\langle \Delta_{h_m} C_{p_1} \ldots C_{p_k} 1, e_n \rangle = \sum_{p' \models p_1} t^{p_1-1} q^{k-1} \langle \Delta_{h_{m-1}} C_{p_2} \ldots C_{p_k} C_{p'} 1, e_n \rangle \\ + \chi(p_1 = 1) \langle \Delta_{h_m} C_{p_2} \ldots C_{p_k} 1, e_{n-1} \rangle$$

where $p' \models p_1$ denotes that $p' = (p'_1, \cdots, p'_{l(p')})$ is a composition of p_1 and we use $C_{p'}$ for $C_{p'_1} \ldots C_{p'_{l(p')}}$. Guided by this symmetric function recursion, Duane, Garsia, and Zabrocki give a recursive map on two-shuffle parking functions. We give a slightly modified version of their map below that we will use to show that, with ndinv as defined above, the right hand side of (4) satisfies the same recursion as the left hand side.

Procedure 3.1. We begin by modifying the first part:

$$PF = \left[\begin{array}{cccc|c} c_1 & c_2 & \ldots & c_{f_1} & \ldots \\ d_1 & d_2 & \ldots & d_{f_1} & \ldots \end{array}\right]$$

1. Remove its first domino $\begin{bmatrix}c_1\\d_1\end{bmatrix}$.
2. For each $1 < i < f_1$ such that $c_i > m$, replace $\begin{bmatrix}c_i\\d_i\end{bmatrix}$ by $\begin{bmatrix}c_i\\d_i - 1\end{bmatrix}$.
3. If adjacent dominoes in the result are of the form $\begin{bmatrix}\ldots & c_b & c_s & \ldots\\ \ldots & d-1 & d & \ldots\end{bmatrix}$, with $c_b > m$ and $c_s \leq m$, then replace them by $\begin{bmatrix}\ldots & c_s & c_b & \ldots\\ \ldots & d-1 & d & \ldots\end{bmatrix}$.
4. Move the modified first part (all $f_1 - 1$ dominoes) to the end of the sequence.

We will call the resulting two line array

$$\Phi(PF) = \begin{bmatrix}\bar{c}_1 & \bar{c}_2 & \ldots & \bar{c}_{n+m-1}\\ \bar{d}_1 & \bar{d}_2 & \ldots & \bar{d}_{n+m-1}\end{bmatrix}$$

Example 3.1. As in Example 1.3, let $m = 3$ and

$$PF = \left[\begin{array}{cccc|cc} 3 & 4 & 1 & 6 & 2 & 5 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{array}\right].$$

Then

$$\Phi(PF) = \left[\begin{array}{cc|ccc} 2 & 5 & 1 & 4 & 6 \\ 0 & 0 & 0 & 1 & 0 \end{array}\right].$$

Remark 3.1 Notice that the resulting parking function may no longer be a proper shuffle as written. While it is convenient to keep track of the original numbers of the cars in future proofs, it is easy to slightly modify our result to again get a two-shuffle parking function. Thus the two line array $\Phi(PF)$ should represent the two-shuffle parking function $\overline{PF}$ obtained by the following steps:

- Let $m' = m - 1$ if the removed car is small and $m' = m$ if the removed car is big.
- Replace in $\Phi(PF)$ all $\overline{c}_i \leq m$ by a "1" and all $\overline{c}_i > m$ by a "2".
- Next, from the highest to the lowest $\overline{d}_i$ values and from right to left replace all the "$1's$" by $1, 2, \ldots, m'$ and all the "$2's$" by $m' + 1, m' + 2, \ldots, n + m - 1$.

For a proof that $\overline{PF}$ is always a two-shuffle parking function, we refer the reader to [Duane et al.(2012)]. Notice next that by calling a car $\leq m$ "small" and a car $> m$ "big", we can freely apply the operation $PF \rightarrow \Psi(PF)$ to the two line array $\Phi(PF)$ and denote the result $\Psi(\Phi(PF))$. Since the action of the map Ψ on a car domino depends only on whether the car is big or small, it follows that the second and third rows of $\Psi(\Phi(PF))$ will be identical to those we would obtain by constructing $\Psi(\overline{PF})$. Since the contents of these two rows together with the relative size of the corresponding cars (i.e. whether they are big or small) is the only information that will be used in the following, we will use $\Psi(\Phi(PF))$ rather than $\Psi(\overline{PF})$. Using these notational conventions, the recursive step used by Duane, Garsia, and Zabrocki in the algorithm giving their ndinv can be simply written in the form

Recursion 3.1. For a parking function PF with k parts,

$$\mathrm{ndinv}(PF) = \begin{cases} 0 & \text{if } c_1 \text{ is big, } n = 1 \\ \mathrm{ndinv}(\overline{PF}) + (k-1)\chi(c_1 \leq m) & \text{otherwise} \end{cases} \tag{5}$$

Thus, if Ψ is applied to the two line array $\Phi(PF)$, the result is the three line array

$$\Psi(\Phi(PF)) = \begin{bmatrix} \overline{c}_1^{\Psi} & \overline{c}_2^{\Psi} & \cdots & \overline{c}_{n+m-1}^{\Psi} \\ \overline{d}_1^{\Psi} & \overline{d}_2^{\Psi} & \cdots & \overline{d}_{n+m-1}^{\Psi} \\ \overline{r}_1^{\Psi} & \overline{r}_2^{\Psi} & \cdots & \overline{r}_{n+m-1}^{\Psi} \end{bmatrix}.$$

then to show that their ndinv and ours are one and the same we need only prove that (5) holds true with $\mathrm{ndinv}(PF)$ replaced by (3) and $\mathrm{ndinv}(\overline{PF})$ replaced by

$$\sum_{\overline{c}_b^{\Psi} > m} \sum_{\overline{c}_s^{\Psi} \leq m} \Big(\chi(b < s)\chi(\overline{d}_b^{\Psi} \leq \overline{r}_s^{\Psi} < \overline{r}_b^{\Psi}) + \chi(b > s)\chi(\overline{d}_b^{\Psi} < \overline{r}_s^{\Psi} \leq \overline{r}_b^{\Psi}) \Big) - m' \tag{6}$$

with m' as defined in Remark 3.1. Notice that to calculate ndinv using the recursion in (5), we need to apply Procedure 3.1 repeatedly. Every time we apply the procedure once, we will remove the first domino and move the resulting first part to the end. Suppose there are k parts in PF. If we apply Procedure 3.1 k times, the first car of each part will be removed. We call this the first round. Let k_1 be the number of parts after the first round. Again, applying Procedure 3.1 k_1 times removes the first car of each of these k_1 parts. We call this the second round.

Definition 3.1 (Round). We define the ith round as applying Procedure 3.1 an additional k_{i-1} times, where k_{i-1} is the number of parts after the $(i-1)$st round.

This notion of "round" beautifully enlightens the relation between our definition of ndinv with the definition of Duane, Garsia and Zabrocki. In fact, it follows from our proofs that the d_i^Ψ gives the round at which a big car c_i^Ψ first appears in the main diagonal and $\bar{r}_i^\Psi$ gives the round at which car c_i^Ψ is removed. Using this it is not difficult to derive that for any given small car c_s^Ψ the expression

$$-1+\sum_{\overline{c}_b^\Psi>m}\Big(\chi(b<s)\chi(\overline{d}_b^\Psi\le\overline{r}_s^\Psi<\overline{r}_b^\Psi)+\chi(b>s)\chi(\overline{d}_b^\Psi<\overline{r}_s^\Psi\le\overline{r}_b^\Psi)\Big)$$

gives precisely the number of big cars that are to the right of c_s^Ψ at the round of its removal in the recursive algorithm of Duane, Garsia and Zabrocki.

4 Our ndinv and Recursion 3.1.

To show that our ndinv satisfies Recursion 3.1 we need to further examine the combination of Φ and Ψ, as it occurs in the following diagram:

$$\begin{array}{ccc} PF & \xrightarrow{\Psi} & \Psi(PF) \\ \Big\downarrow{\scriptstyle\Phi} & & \\ \Phi(PF) & \xrightarrow{\Psi} & \Psi(\Phi(PF)) \end{array}.$$

Let us say $c_i^\Psi=c_b$ and $\overline{c}_j^\Psi=c_b$ then, using the symbol "ind" to denote an index, we will, simply write "$\mathrm{ind}^\Psi(c_b)=i$" and "$\mathrm{ind}^{\Psi\Phi}(c_b)=j$". It will also be convenient to have an alternate notation for the $d,\overline{d}$ and $r,\overline{r}$ values. For instance, if we have $i=\mathrm{ind}^\Psi(c_b)$, $d_i^\Psi=3$ and $r_i^\Psi=5$, we will simply express this by writing $d^\Psi(c_b)=3$ and $r^\Psi(c_b)=5$. On the other hand if $j=\mathrm{ind}^{\Psi\Phi}(c_b)$, $\overline{d}_j^\Psi=3$ and $\overline{r}_j^\Psi=5$ we will write $\overline{d}^\Psi(c_b)=3$ and $\overline{r}^\Psi(c_b)=5$. The entries of PF and $\Phi(PF)$ will be handled in an analogous manner. Thus if $c_i=c_b$ and $d_i=4$ we may also write $\mathrm{ind}(c_b)=i$ and $d(c_b)=4$. Similarly if $\overline{c}_i=c_b$ and $\overline{d}_i=4$ we may write $\mathrm{ind}^\Phi(c_b)=i$ and $\overline{d}(c_b)=4$ etc. Finally, it will also be convenient to write $c_1\to c_2$ to state that car c_1 is to the left of car c_2 in a given expression. Using this notation, we can give an overview of the path we follow to establish that our ndinv and the ndinv of Duane, Garsia and Zabrocki satisfy the same recursion. To be precise, our proof is based on the following facts:

Theorem 4.1. *With $\Psi(PF)$ and $\Psi(\Phi(PF))$ as defined above, we have for any cars c,c_1,c_2:*

- *Fact (1) If $f_1<\mathrm{ind}^\Psi(c_1),\mathrm{ind}^\Psi(c_2)$, then $c_1\to c_2$ in $\Psi(PF)$ if and only if $c_1\to c_2$ in $\Psi(\Phi(PF))$.*
- *Fact (2) If $1<\mathrm{ind}^\Psi(c_1),\mathrm{ind}^\Psi(c_2)\le f_1$, then $c_1\to c_2$ in $\Psi(PF)$ if and only if $c_1\to c_2$ in $\Psi(\Phi(PF))$.*
- *Fact (3) If $1<\mathrm{ind}^\Psi(c_1)\le f_1<\mathrm{ind}^\Psi(c_2)$, then $\mathrm{ind}^{\Psi\Phi}(c_1)>\mathrm{ind}^{\Psi\Phi}(c_2)$.*
- *Fact (4) For $\mathrm{ind}^\Psi(c)>f_1$ we have $\overline{d}^\Psi(c)=d^\Psi(c)$ and $\overline{r}^\Psi(c)=r^\Psi(c)$.*
- *Fact (5) For $1<\mathrm{ind}^\Psi(c)<f_1$ we have $\overline{d}^\Psi(c)=d^\Psi(c)-1$ if c is a big car and $\overline{r}^\Psi(c)=r^\Psi(c)-1$, whether c is big or small.*

It may be good before closing to show how these facts give all that is needed to establish our desired goal.

Recall that, in the present notation, by definition, a big car c_b and a small car c_s form a diagonal inversion in $\Psi(PF)$ if either $c_b \to c_s$ and $d^\Psi(c_b) \le r^\Psi(c_s) < r^\Psi(c_b)$ or $c_s \to c_b$ and $d^\Psi(c_b) < r^\Psi(c_s) \le r^\Psi(c_b)$.

Theorem 4.2. *In $\Psi(PF)$, if the first car is small, then it forms a diagonal inversion with a big car c_b only when the big car is on the main diagonal. ($d^\Psi(c_b) = 0$.) If the first car is big, it forms no diagonal inversions.*

Proof: By definition $r_1^\Psi = 1$. We look at the two cases separately.

- ($c_1^\Psi = c_s \le m$.) Then we want all big cars (c_b) such that $d^\Psi(c_b) < 1 \le r^\Psi(c_b)$. Those are exactly the big cars on the main diagonal.
- ($c_1^\Psi = c_b > m$.) Then we want all small cars (c_s) such that $d^\Psi(c_b) \le r^\Psi(c_s) < 1$. Since $r^\Psi(c_s) \ge 1$, there are no such cars.

□

Keeping this in mind let us see how these diagonal inversions change after we apply Φ.

Theorem 4.3. *For $\operatorname{ind}^\Psi(c_s) > 1$ and $\operatorname{ind}^\Psi(c_b) > 1$, a small car c_s and a big car c_b form a diagonal inversion in $\Psi(PF)$ if and only if they form a diagonal inversion in $\Psi(\Phi(PF))$.*

Proof: We split the argument into cases:

- ($\operatorname{ind}^\Psi(c_s), \operatorname{ind}^\Psi(c_b) > f_1$.) Fact (1) and Fact (4) make this case trivial.
- ($1 < \operatorname{ind}^\Psi(c_s), \operatorname{ind}^\Psi(c_b) \le f_1$.) Fact (2) gives $c_b \to c_s$ or $c_s \to c_b$ in both $\Psi(PF)$, $\Psi(\Phi(PF))$ and Fact (5) gives that $\overline{d}^\Psi(c_b) \le \overline{r}^\Psi(c_s) < \overline{r}^\Psi(c_b)$ is $d^\Psi(c_b) - 1 \le r^\Psi(c_s) - 1 < r^\Psi(c_b) - 1$ in the first case and $\overline{d}^\Psi(c_b) < \overline{r}^\Psi(c_s) \le \overline{r}^\Psi(c_b)$ is $d^\Psi(c_b) - 1 < r^\Psi(c_s) - 1 \le r^\Psi(c_b) - 1$ in the second case.
- ($1 < \operatorname{ind}^\Psi(c_s) \le f_1 < \operatorname{ind}^\Psi(c_b)$.) Then $c_s \to c_b$ in $\Psi(PF)$ but Fact (3) gives $c_b \to c_s$ in $\Psi(\Phi(PF))$. Nevertheless, Facts (4) and (5) convert $d^\Psi(c_b) < r^\Psi(c_s) \le r^\Psi(c_b)$, or better $d^\Psi(c_b) \le r^\Psi(c_s) - 1 < r^\Psi(c_b)$ into $\overline{d}^\Psi(c_b) \le \overline{r}^\Psi(c_s) < \overline{r}^\Psi(c_b)$ as desired.
- ($1 < \operatorname{ind}^\Psi(c_b) \le f_1 < \operatorname{ind}^\Psi(c_s)$.) Then $c_b \to c_s$ in $\Psi(PF)$ but Fact (3) gives $c_s \to c_b$ in $\Psi(\Phi(PF))$. Nevertheless, again Facts (4) and (5) convert $d^\Psi(c_b) \le r^\Psi(c_s) < r^\Psi(c_b)$, or better $d^\Psi(c_b) - 1 < r^\Psi(c_s) \le r^\Psi(c_b) - 1$ into $\overline{d}^\Psi(c_b) < \overline{r}^\Psi(c_s) \le \overline{r}^\Psi(c_b)$ as desired.

□ As we can clearly see, Theorem 4.2 accounts for the second term in the second case of (5) and Theorem 4.3 accounts for the first term second case of (5), when we replace $\operatorname{ndinv}(\overline{PF})$ by the expression in (6). Since our ndinv and the ndinv of Duane, Garsia and Zabrocki are equal in the base case, Theorem 4.1 is all that is needed to show that these two ndinvs satisfy the same recursion and that, consequently, they must be identical.

5 Conclusion.

We have seen that Theorems 4.1, 4.2 and 4.3 prove that our ndinv satisfies the Recursion 3.1. In particular, this combinatorial result combined with the the symmetric function results of Duane, Garsia and Zabrocki in [Duane et al.(2012)]. proves that

Theorem 5.1. *With the ndinv defined in (6) and* $(p_1, p_2, \ldots p_k) \vdash n$ *for any integer* $m \geq 0$ *we have*

$$\langle \Delta_{h_m} C_{p_1} \ldots C_{p_k} 1, e_n \rangle = \sum_{\substack{PF \text{ an m,n two-shuffle parking function} \\ \mathrm{comp}(PF) = (p_1, \cdots, p_k)}} t^{\mathrm{area}(PF)} q^{\mathrm{ndinv}(PF)}.$$

It would be interesting to consider if the ndinv statistic could be extended to give a statistic on all parking functions.

We end now with the proof of a prior statement about the sequence r_i^Ψ.

Theorem 5.2. *For each car* c*,* $\bar{r}^\Psi(c)$ *gives the number of the round in which car* c *is removed when we apply Procedure 3.1 repeatedly.*

Proof: Suppose $\bar{r}^\Psi(c) = 1$ for some car c. This happens if and only if for the car c_b' preceding c is a big car and $\overline{d}(c_b') = 0$, in other words c_b' is a big car on the main diagonal of PF. This is true exactly when c is the first car in some part of PF and will be removed in the first round. Moreover, by Fact (5), the r_j^Ψ value of any car will decreased by 1 in any round where it is not removed. This completes the proof by induction. □

References

[Duane et al.(2012)] A. Duane, A. M. Garsia, and M. Zabrocki. A new 'dinv' arising from the two part case of the Shuffle Conjecture. *ArXiv e-prints.*, May 2012. URL http://adsabs.harvard.edu/abs/2012arXiv1205.6128D.

[Garsia and Haglund(2002)] A. M. Garsia and J. Haglund. A proof of the q, t-Catalan positivity conjecture. *Discrete Math.*, 256(3):677–717, 2002. ISSN 0012-365X. doi: 10.1016/S0012-365X(02)00343-6. URL http://dx.doi.org/10.1016/S0012-365X(02)00343-6. LaCIM 2000 Conference on Combinatorics, Computer Science and Applications (Montreal, QC).

[Garsia and Haiman(1996)] A. M. Garsia and M. Haiman. Some natural bigraded S_n-modules and q, t-Kostka coefficients. *Electron. J. Combin.*, 3(2):Research Paper 24, approx. 60 pp. (electronic), 1996. ISSN 1077-8926. URL http://www.combinatorics.org/Volume_3/Abstracts/v3i2r24.html. The Foata Festschrift.

[Haglund(2004)] J. Haglund. A proof of the q, t-Schröder conjecture. *Int. Math. Res. Not.*, (11):525–560, 2004. ISSN 1073-7928. doi: 10.1155/S1073792804132509. URL http://dx.doi.org/10.1155/S1073792804132509.

[Haglund et al.(2005)] J. Haglund, M. Haiman, N. Loehr, J. B. Remmel, and A. Ulyanov. A combinatorial formula for the character of the diagonal coinvariants. *Duke Math. J.*, 126(2):195–232, 2005. ISSN 0012-7094. doi: 10.1215/S0012-7094-04-12621-1. URL http://dx.doi.org/10.1215/S0012-7094-04-12621-1.

[Haglund et al.(2011)] J. Haglund, J. Morse, and M. Zabrocki. A compositional shuffle conjecture specifying touch points of the dyck path. *Canadian Journal of Mathematics*, 2011. (to appear), (arXiv:1008:0828).