International Journal of Pure and Applied Mathematics
Volume 80 No. 4 2012, 515-534
ISSN: 1311-8080 (printed version)
url: http://www.ijpam.eu

ijpam.eu

# RELATING A COUNTING ALGORITHM TO THE JACOBIAN CONJECTURE

Kristen Lampe

Department of Mathematics
Carroll University
100, North East Avenue, Waukesha, Wisconsin, 53186, USA

**Abstract:** The Jacobian conjecture deals with a polynomial map $F : k^n \to k^n$ where $k$ is a field of characteristic zero. It says that if the Jacobian determinant is a unit in $k$, then $F$ has a polynomial inverse. A known equivalent statement of this conjecture uses the vanishing of a class of polynomials $P_r$, related to $F$.

A known formula expresses the inverse of $F$ as a power series whose coefficients are parameterized by labeled, rooted forests. Here, we develop a new expression for the polynomials $P_r$, in terms of labeled, rooted trees. We then express the coefficients of $P_r$ in terms of the coefficients of $F$. Using a known counting method originally developed to express the inverse of $F$ in terms of labeled, rooted forests, we explore the relationship between the coefficients of the $P_r$ and the coefficients of the inverse of $F$.

The result is a simplification of the inverse formula for $F$ under the Jacobian hypothesis.

**AMS Subject Classification:** 05E40, 12Y05
**Key Words:** Jacobian conjecture, counting formula, polynomials, reversion, trees, forests

## 1. Introduction

In this paper, we explore an application of the counting process discussed in "A

Received: June 19, 2012

Counting Formula for Labeled, Rooted Forests" [3]. Under the condition of the Jacobian Conjecture, this counting process shows that the reversion formula for the inverse function of a polynomial simplifies nicely. To understand what this means, we outline the background information here.

For $k$ a field of characteristic zero, we start with $F = (F_1, \ldots, F_n) : k^n \to k^n$ a polynomial map. Let $J(F) = (D_i F_j)$ denote the Jacobian matrix for $F$, where $D_i = \frac{\delta}{\delta x_i}$. Then, $j(F) = \det J(F)$ denotes the Jacobian determinant. The *Jacobian Conjecture* states that if $j(F) = 1$, then $F$ has a polynomial inverse. For a history of this open problem, see Bass et al. [1]. One reduction is that we may assume $F = (F_1, \ldots, F_n)$ is of the form $F_i = X_i - H_i$ where all $H_i$ are polynomials having degree at least two [1]. We write $F = X - H$.

In "Formal Inverse Expansion and the Jacobian Conjecture," [6] Wright defines a set of polynomials, $P_r$, that lead to an equivalent statement of the Jacobian Conjecture. These polynomials use indeterminates $U_{ij}$ for $1 \leq i, j \leq n$, and are defined as

$$P_r = (-1)^r \sum_{1 \leq j_1 < \cdots < j_r \leq n} \det \begin{pmatrix} U_{j_1 j_1} & \cdots & U_{j_1 j_r} \\ \vdots & & \vdots \\ U_{j_r j_1} & \cdots & U_{j_r j_r} \end{pmatrix}$$

For $r = 1, \ldots, n$, and $F = X - H$, then

$$P_r(J(H)) = P_r(D_1 H_1, D_1 H_2, \ldots, D_n H_n).$$

It is known that the Jacobian hypothesis, namely $j(F) = 1$, is equivalent to

$$\sum_{r=1}^{n} P_r(J(H)) = 0.$$

The Jacobian conjecture has been reduced to the special case where $H$ is homogeneous of degree $\delta = 3$ [1].

In the case that $H$ is homogeneous of any degree $\delta \geq 2$, then

$$j(F) = 1 \iff P_r(J(H)) = 0 \text{ for each } r,\ 1 \leq r \leq n.$$

In this paper, we express $P_r(J(H))$ in terms of labeled, rooted trees. Our goal is then to express the coefficients of $P_r(J(H))$ in terms of the coefficients of $H$. Next, we develop a relationship between these coefficients and the Raney coefficients defined below (see Section 2). This allows us to restate the inverse of $F$ in a much simpler manner when the Jacobian hypothesis of $j(F) = 1$ is applied.

Before we develop these polynomials further, we need to summarize the results of two papers, "Reversion of Power Series and the Extended Raney Coefficients" by Cheng et al. [2] and Lampe's paper [3].

## 2. Background

### 2.1. Forest Definitions

One approach to expressing the inverse of $F$ has used *labeled, rooted trees.* A labeled, rooted tree, $T$, is a minimally connected finite graph with one node designated the root and each node given a label chosen from $\{1, 2, \ldots, n\}$. Two adjacent nodes have a parent-child relationship, where that node closer to the root is the *parent.* A *forest*, $S$, is an ordered collection of trees, sorted by root-label. Each forests, $S$, has an *inventory* $\alpha = (\alpha_1, \ldots, \alpha_n)$ defined as follows. For $i = 1, \ldots n$, let $\alpha_i = (\alpha_i^{k_1,\ldots,k_n})$ be an $n$ dimensional array over the natural numbers with finitely many non-zero entries. Each entry, $\alpha_i^{k_1,\ldots,k_n}$, denotes the number of $i$-labeled nodes in the forest having $k_j$ children labeled $j$, for $1 \leq j \leq n$. The number of $i$-labeled nodes in $S$ is denoted $\sigma(\alpha_i) = \sum_k \alpha_i^k$, where we sum over all tuples $k = (k_1, \ldots, k_n)$. The number of $i$-labeled children of $j$-labeled nodes is denoted $\sigma_i(\alpha_j) = \sum_k k_i \alpha_j^k$. Notice that any forest $S$ has $\sigma(\alpha_i) \geq \sum_{j=1}^n \sigma_i(\alpha_j)$ for all $i$, so we will restrict ourselves to such $\alpha$. Finally, $R(\alpha)$ is the number of forests having inventory $\alpha$.

We delve a bit farther and see that $p_i = \sigma(\alpha_i) - \sum_{j=1}^n \sigma_i(\alpha_j)$ is the number of $i$-labeled roots in $S$. We say $p = (p_1, \ldots, p_n)$ is the *root-type* of $S$. Also, we let $q = (q_1, \ldots, q_n)$ denote the *leaf-type* of $S$ when there are $q_i$ $i$-labeled leaves. We call $k = (k_1, \ldots, k_n)$ the *child-type* of a node. Also, we let $|k| = k_1 + \cdots + k_n$.

### 2.2. The Reversion Formula

For $F_i = X_i - H_i$, where

$$H_i = \sum_{|k| \geq 2} a_i^k X^k,$$

Theorem 3.1 in [2] proves that

$$X_1^{p_1} \cdots X_n^{p_n} = \sum_{q=(q_1,\ldots,q_n)} e_p^q F_1^{q_1} \cdots F_n^{q_n}$$

where

$$e_p^q = \sum_* R(\alpha) \prod_{j=1}^n \prod_{|k|\geq 2} (a_j^k)^{\alpha_j^k}$$

and $*$ is all $\alpha$ having root-type $p$ and leaf-type $q$, such that $q_i = \alpha_i^{(0,\ldots,0)}$ and $\alpha_i^k = 0$ whenever $|k| = 1$. Notice that if $p = e_i$ a unit vector, this formula gives the inverse $G_i$ of $F_i$ as a power series. The coefficients of $G$ are called the *Raney coefficients*.

We can further develop this formula, and express $R(\alpha)$ as a determinant. Namely, if

$$M(\alpha_i) = \frac{\sigma(\alpha_i)!}{\prod(\alpha_i^k!)}$$

and

$$\widetilde{M}(\alpha_i) = \begin{cases} 1 & \text{if all } \alpha_i^k = 0 \\ \frac{1}{\sigma(\alpha_i)} M(\alpha_i) & \text{otherwise} \end{cases}$$

and

$$M(\alpha) = \prod_{i=1}^n M(\alpha_i)$$

then [2] shows we have

$$\begin{aligned} R(\alpha) &= \left| \begin{pmatrix} M(\alpha_1) & 0 & \cdots & 0 \\ 0 & M(\alpha_2) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & M(\alpha_n) \end{pmatrix} \right. \\ &- \left[ \begin{pmatrix} \widetilde{M}(\alpha_1) & 0 & \cdots & 0 \\ 0 & \widetilde{M}(\alpha_2) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \widetilde{M}(\alpha_n) \end{pmatrix} \right. \\ &\quad \left. \left. \cdot \begin{pmatrix} \sigma_1(\alpha_1) & \sigma_2(\alpha_1) & \ldots & \sigma_n(\alpha_1) \\ \sigma_1(\alpha_2) & \sigma_2(\alpha_2) & \cdots & \sigma_n(\alpha_2) \\ \vdots & \vdots & & \vdots \\ \sigma_1(\alpha_n) & \sigma_2(\alpha_n) & \ldots & \sigma_n(\alpha_n) \end{pmatrix} \right] \right| \\ &= M(\alpha) \left| I_n - \begin{pmatrix} \frac{\sigma_1(\alpha_1)}{\sigma(\alpha_1)} & \cdots & \frac{\sigma_n(\alpha_1)}{\sigma(\alpha_1)} \\ \vdots & \vdots & \vdots \\ \frac{\sigma_1(\alpha_n)}{\sigma(\alpha_n)} & \cdots & \frac{\sigma_n(\alpha_n)}{\sigma(\alpha_n)} \end{pmatrix} \right| \end{aligned} \tag{1}$$

In what follows, we assume $\sigma(\alpha_i) > 0$ for every $i$. Indeed, otherwise, we may move to a lower dimension.

Now, in [3], a counting method is developed, explaining in terms of forests why the above formula holds. This counting method is of integral importance to the results of this paper, so we next endeavor to summarize the algorithm.

### 2.3. The Counting Method

Here, we fix an inventory, $\alpha$, and develop the formula for $R(\alpha)$. First, we represent each forest, $S$, as a *sorted permutation* as follows. For $i \in \{1, \ldots, n\}$ and $k = (k_1, \ldots, k_n)$, define $L_\alpha$ to be the set containing the formal symbol, or *letter* $i^k$, $\alpha_i^k$ times. We regard the elements of $L_\alpha$ as distinct: put a different shade on each element of the set of a given type. An $L_\alpha$*-sorted permuatation* is an arrangement of the elements of $L_\alpha$:

$$(1^{a_1}, \ldots, 1^{a_l})(2^{b_1}, \ldots, 2^{b_k}) \ldots (n^{c_1}, \ldots, n^{c_m}).$$

Each entry's base represents a label and the superscript is an $n$-tuple representing a child-type.

A recording procedure enables us to express each forest as a unique sorted-permutation, where each letter, $i^k$, represents an $i$-labeled vertex with child-type $k$. Since recording is injective, but not surjective, not every sorted-permutation represents a forest. Conversely, we have a building procedure that starts with a sorted-permutation and builds a forest. Those sorted-permutations not resulting in forests are called *failures*. A failing sorted-permutation results in what we call an *initial forest* which does not use all the letters. The remaining letters then build to *wreaths*, each of which is a connected, directed, labeled planar graph with precisely one loop and one vertex in the loop designated the wreath root. We then call the ordered collection of wreaths a *bramble*. An initial forest and its bramble together form an *arbor*. Let $A_\alpha$ denote the set of arbors formed from $L_\alpha$.

Notice that the inventory, $\alpha$, can be easily recovered from a sorted-permutation. Fixing $\alpha$, if we remove the shading on the letters, the number of distinct sorted-permutations is $M(\alpha)$. Each forest corresponds to a unique sorted-permutation, so the number of sorted-permutations representing a forest is $R(\alpha)$. Hence, there are $M(\alpha) - R(\alpha)$ sorted-permutations whose arbors have non-trivial brambles.

Thus, we can verify the formula for $R(\alpha)$ by showing that the number of failing sorted-permutations is

$$M(\alpha) - R(\alpha) = M(\alpha) - M(\alpha)\left| I_n - \begin{pmatrix} \frac{\sigma_1(\alpha_1)}{\sigma(\alpha_1)} & \cdots & \frac{\sigma_n(\alpha_1)}{\sigma(\alpha_1)} \\ \vdots & \vdots & \vdots \\ \frac{\sigma_1(\alpha_n)}{\sigma(\alpha_n)} & \cdots & \frac{\sigma_n(\alpha_n)}{\sigma(\alpha_n)} \end{pmatrix} \right|$$

In order to demonstrate this equality, we need to classify the failing sorted-permutations. We do this by examining the bramble. It is a direct result of the building procedure that the wreath root of each wreath is the unique vertex of that wreath's loop of lowest label. The bramble consists of wreaths ordered by wreath root. We classify a wreath by its *cycle*, a subset of the loop vertices consisting of vertices no two of which have the same label. The cycle of a wreath is found in a precise manner beginning with the wreath root and traveling around the loop until a label is repeated. All vertices from the first occurrence up to, but not including, the second occurrence are removed from the direct loop. Repeat this process until all labels are distinct. What remains is the wreath's cycle.

To any sorted-permutation, we associate a set of cycles (the empty set when the sorted-permutation builds to an arbor with empty bramble) that are disjoint – meaning no label is repeated in this set. This is obtained by starting with the rightmost wreath's cycle and putting it in the set. Then, proceeding to the left, adding a cycle to the set when it is disjoint from those already in the set. We call this set $\Phi(A)$, where $A$ is the arbor built from the given sorted-permutation.

Next, we categorize $\Phi(A)$ according to its *cycle-type*. A cycle-type is a permutation, $\tau$, of a subset of $\{1, 2, \ldots, n\}$. So, for example, if

$$\Phi(A) = \{(1_2^{200}), (2^{111}, 3^{010})\},$$

where the superscript is the child-type of the vertex and the subscript denotes which child is the next loop vertex, then it's cycle-type is (1)(2,3).

Further, given a set of disjoint cycles, we can build an arbor, $A$, such that $\Phi(A)$ is that set. It is worth noting that this procedure is more complex and involves some subtlety, so we won't go into it here. More importantly to us here, this building procedure is the inverse procedure to $\Phi$ (see Lemmas 6.1 and 6.3 in [3]).

Given a cycle-type $\tau$, then the number of sets of cycles in $L_\alpha$ of type $\tau$ is $\prod_{i\in\tau} \sigma_{\tau(i)}(\alpha_i)$, where we say $i \in \tau$ if the label $i$ appears in $\tau$.

Now, letting $C$ denote a set of cycles of type $\tau$ and $\alpha'$ be the inventory of $S' = S - C$, then

$$M(\alpha') = \prod_{i\notin\tau}(\sigma(\alpha_i)!) \prod_{i\in\tau}((\sigma(\alpha_i) - 1)!).$$

Thus, the number of arbors, $A$, such that $\Phi(A)$ contains a set $C$ of cycle-type $\tau$ is

$$\prod_{i\in\tau}(\sigma_{\tau(i)}(\alpha_i))\prod_{i\notin\tau}(\sigma(\alpha_i)!)\prod_{i\in\tau}((\sigma(\alpha_i)-1)!).$$

Recall, we are regarding each element of $L_\alpha$ as distinct (by shading), so $M(\alpha)=\prod_{i=1}^{n}(\sigma(\alpha_i)!)$. Now, we turn to $R(\alpha)$. Let $e(\tau)$ be the number of cycles in $\tau$ (cycles used in the normal sense here). Then,

$$\begin{aligned} R(\alpha) &= M(\alpha)\left| I_n - \begin{pmatrix} \frac{\sigma_1(\alpha_1)}{\sigma(\alpha_1)} & \cdots & \frac{\sigma_n(\alpha_1)}{\sigma(\alpha_1)} \\ \vdots & \vdots & \vdots \\ \frac{\sigma_1(\alpha_n)}{\sigma(\alpha_n)} & \cdots & \frac{\sigma_n(\alpha_n)}{\sigma(\alpha_n)} \end{pmatrix}\right| \\ &= M(\alpha) \\ &- \left[\sum_{\tau}(-1)^{e(\tau)+1}\prod_{i\in\tau}(\sigma_{\tau(i)}(\alpha_i))\prod_{i\notin\tau}(\sigma(\alpha_i)!)\prod_{i\in\tau}((\sigma(\alpha_i)-1)!)\right] \\ &= M(\alpha) \\ &- \left[\sum_{\tau}(-1)^{e(\tau)+1}\Big|\{A\in A_\alpha|\Phi(A) \text{ contains a set} \right. \\ &\qquad \left. \text{of cycle} - \text{type } \tau\}\Big|\right] \end{aligned}$$

The first term, $M(\alpha)$, counts the total number of shaded arbors. The second term in the equation counts the number of failures as follows. If an arbor $A$ has $\Phi(A)$ containing a subset $C$ of cycle-type $\tau$ with $e(\tau)=1$, then that arbor contributes to the portion of the summation over all $\tau$ with $e(\tau)=1$. However, this is too many – those arbors whose $\Phi(A)$ has two or more cycles have been counted more than once. The remaining terms compensate for this overcount in a manner that coincides with the cardinality of the union of sets:

$$E_1\cup\cdots\cup E_N=\sum_{j=1}^{N}(-1)^{j+1}\left(\sum_{1\le i_1<\cdots<i_j\le N}|E_{i_1}\cap\cdots\cap E_{i_i}|\right)$$

Removing the shading from the elements of $L_\alpha$ then gives

$$M(\alpha)=\prod_{i=1}^{n}\left[\frac{\sigma(\alpha_i)!}{\prod(\alpha_i^k!)}\right]$$

and

$$R(\alpha)=M(\alpha)-\frac{1}{\prod(\alpha_i^k!)}\left[\sum_{\tau}(-1)^{e(\tau)+1}\Big|\{A\in A_\alpha|\Phi(A) \text{ contains a}\right.$$

$$\text{set of cycle} - \text{type } \tau\}\Big|\Big]$$

Thus, we are able to show directly that the number of forests $S$ with inventory $\alpha$ is given by the determinantal formula for $R(\alpha)$ .

## 3. Developing $P_r(J(H))$

Here, we turn our attention to the power series $P_r(J(H))$. We will express their coefficients in terms of the coefficients of $H$. Then, in Section 4, we relate this new expression for $P_r(J(H))$ to the Raney coefficients using the counting formula summarized in section 2.3. This leads us to another way of viewing the Jacobian hypothesis.

### 3.1. Graphs and Their Polynomials

We will work over fixed dimension $n$. We will define a special class of graphs $\mathcal{G}$ and express $P_r(J(H))$ in terms of these graphs. Recall that a *graph*, $G$, is a set $V(G)$ of vertices and a set $E(G) \subseteq V(G) \times V(G)$ of edges. A *directed graph* is a graph where $(v_1, v_2) \in E(G)$ indicates a directed edge from $v_1$ to $v_2$. With directed graphs, we can discuss a vertex's *indegree* and *outdegree*, the number of edges coming into and going out from a vertex, respectively. All graphs henceforth are directed. We now focus on a restricted class of graphs, $\mathcal{G}$. For $G \in \mathcal{G}$ we require that $|V(G)| \le n$ and, for every $v \in V(G)$, indeg$(v)$ = outdeg$(v) = 1$. Notice that this forces $|V(G)| = |E(G)|$. Thus defined, each $G \in \mathcal{G}$ is the union of directed loops. A loop has *length* $i$ when there are $i$ vertices in the loop.

For $G \in \mathcal{G}$, we say the *loop-class* of $G$ is the $n$-tuple $(b_1, \ldots, b_n)$, where $b_i$ is the number of loops of length $i$. Thus, $\sum_{i=1}^{n} ib_i = |V(G)| \le n$. Note, there is a one-to-one correspondence between loop-classes, $b$, and isomorphism classes of graphs $G \in \mathcal{G}$.

Next, we want to label these graphs. We define the set

**Notation 1.**

$$\overline{\mathcal{S}_r} = \{\tau | \tau \text{ chooses } r \text{ distinct elements from } \{1, 2, \ldots, n\}$$

$$\text{and forms a permutation with them}\}$$

$$\ell(\tau) = \text{ the set of numbers chosen from } \{1, 2, \ldots, n\}.$$

For example,with $n = 4$, $\tau = (2)(3,4) \in \overline{\mathcal{S}_3}$ and $\ell(\tau) = \{2,3,4\}$. Notice that $\tau$ has a loop-class. As in Section 2.3 we will also refer to $\tau$'s cycle-type. Next, we label these graphs.

For $G \in \mathcal{G}$ having $|V(G)| = r \leq n$ and loop-class $(b_1, \ldots, b_n)$, a *label* on $G$ is an *injective* function $\ell : V(G) \longrightarrow \{1, 2, \ldots, n\}$. Labeling endows each vertex of $G$ with a number unique to that vertex. Since $G$ consists entirely of loops, this labeling determines some $\tau \in \overline{\mathcal{S}_r}$ such that $\ell(\tau) = \ell(V(G))$ and $\tau$ has loop-class $(b_1, \ldots, b_n)$. Namely, $\tau(i)$ is the label on the node that is the $i$-node's child.

We have isomorphism classes of labeled graphs, where two labeled graphs are isomorphic if they have the same $\tau$ and, hence, the same loop-class. Note that permutations do not distinguish between the order of sub-cycles, i.e. (1)(2, 3,4) is the same permutation as (3,4,2)(1). Thus, we may choose the unique class representative as that $\tau$ that lists each sub-cycle with the lowest number first and lists the sub-cycles in increasing order of leading numbers.

Let $\mathcal{L}$ be the set isomorphism classes of labeled graphs. Let $L \in \mathcal{L}$. So, $L = [(G, \ell)]$, the class of $(G, \ell)$ where $G \in \mathcal{G}$ and $\ell$ is the label. To each $L$, we associate a polynomial. Define

**Notation 2.**

$$v^+ = \text{the child of } v$$

$$\ell(v^+) \text{ the label on the cycle node}, v^+.$$

Let

$$P_L = \operatorname{sgn}(\tau) \prod_{v \in V(L)} D_{\ell(v^+)} H_{\ell(v)}$$

where $\tau$ is that element of $\overline{\mathcal{S}_r}$ determined by $\ell$ and $G$. Note that the elements of $\mathcal{L}$ are in one-to-one correspondence with the elements of $\overline{\mathcal{S}_r}$ as $r$ varies.

**Claim 3.**

$$P_r(J(H)) = (-1)^r \sum_{\substack{L \in \mathcal{L} \\ |V(L)|=r}} P_L$$

*Proof.* From Section 1, we have

$$P_r(J(H)) = (-1)^r \sum_{1 \leq j_1 < \cdots < j_r \leq n} \det \begin{pmatrix} D_{j_1} H_{j_1} & \cdots & D_{j_1} H_{j_r} \\ \vdots & & \vdots \\ D_{j_r} H_{j_1} & \cdots & D_{j_r} H_{j_r} \end{pmatrix}.$$

Then,

$$\sum_{1\le j_1<\cdots<j_r\le n} \det\begin{pmatrix} D_{j_1}H_{j_1} & \cdots & D_{j_1}H_{j_r} \\ \vdots & & \vdots \\ D_{j_r}H_{j_1} & \cdots & D_{j_r}H_{j_r} \end{pmatrix}$$

$$= \sum_{1\le j_1<\cdots<j_r\le n} \sum_{\gamma\in\mathcal{S}_r} \operatorname{sgn}(\gamma) D_{j_1}H_{\gamma(j_1)} D_{j_2}H_{\gamma(j_2)} \ldots D_{j_r}H_{\gamma(j_r)}$$

$$= \sum_{\substack{1\le j_1<\cdots< \\ j_r\le n}} \sum_{\substack{b=(b_1,\ldots,b_r) \\ \sum ib_i=r}} \sum_{\substack{\gamma\in\mathcal{S}_r \text{ of} \\ \text{loop-class } b}} \operatorname{sgn}(\gamma) D_{j_1}H_{\gamma(j_1)} D_{j_2}H_{\gamma(j_2)} \ldots D_{j_r}H_{\gamma(j_r)}$$

$$= \sum_{\substack{b=(b_1,\ldots,b_r) \\ \sum ib_i=r}} \sum_{\substack{1\le j_1<\cdots< \\ j_r\le n}} \sum_{\substack{\gamma\in\mathcal{S}_r \text{ of} \\ \text{loop-class } b}} \operatorname{sgn}(\gamma) D_{j_1}H_{\gamma(j_1)} D_{j_2}H_{\gamma(j_2)} \ldots D_{j_r}H_{\gamma(j_r)}$$

Recall, there is a one-to-one correspondence between loop-classes, $b = (b_1, \ldots b_r)$, and isomorphism classes of graphs with $|V(G)| = r$. Further, a label, $\ell$, for $G$, unique up to isomorphism class of $(G, \ell)$ in $\mathcal{L}$, is defined by the inner two sums. Finally, for fixed $L$, the summand $\operatorname{sgn}(\gamma) D_{j_1}H_{\gamma(j_1)} D_{j_2}H_{\gamma(j_2)} \ldots D_{j_r}H_{\gamma(j_r)}$ is the same as $P_L$. Thus, we have

$$P_r(J(H)) = (-1)^r \sum_{\substack{L\in\mathcal{L} \\ |V(L)|=r}} P_L$$

as desired. $\square$

### 3.2. The Coefficients of $P_r(J(H))$

Next, we want to describe the coefficients of $P_r(J(H))$. As in Section 2.2 , let

$$H_i = \sum_{|k|\ge 2} a_i^k X^k. \tag{2}$$

For $l$ an $n$-tuple, we write $P_L$ as $\operatorname{sgn}(\tau)\sum_l c_{l,L}X^l$. Our goal is to express $c_{l,L}$ in terms of the coefficients,$a_i^k$, of $H$. Notice that, in $P_L$, $c_{l,L} = 0$ if $|l| < |V(L)|$, because for each vertex in $L$, we take the derivative once of an $H_i$. This gives either a zero term or a new non-zero term of degree at least 1. Taking the product of $|V(L)|$ of these non-zero terms gives a degree of at least $|V(L)|$. Thus, we have

$$P_L = \operatorname{sgn}(\tau) \sum_{|l|\ge|V(L)|} c_{l,L}X^l.$$

The development of these ideas closely parallels a similar concept using rooted trees developed by Wright in "The Tree Formulas for Reversion of Power Series" [5].

Notice that $X^l$ arises in $P_L$ as the summation over all functions $t : V(L) \longrightarrow \mathbf{N}^n$ such that $\sum_{v \in V(L)} t(v) = l$. Letting $d_{t(v)}$ denote the coefficient of $X^{t(v)}$ in $D_{\ell(v^+)} H_{\ell(v)}$, we have

$$c_{l,L} = \sum_{\substack{t:V(L)\to\mathbf{N}^n \\ \sum t(v)=l}} \prod_{v\in V(L)} d_{t(v)}.$$

Note that, since $l$ is fixed, this is a finite sum.

As before, let $k(v)$ denote the *child-type* of $v$. That is, $k(v)$ is an $n$-tuple and $k_i(v)$ denotes the number of children of $v$ labeled $i$. If $v^+$ is labeled $i$, then $k(v) = e_i$, where $e_i$ is the standard unit vector.

Now, $d_{t(v)} X^{t(v)}$ arises in $D_{\ell(v^+)} H_{\ell(v)}$ from applying $D_{\ell(v^+)}$ to $a_{\ell(v)}^{t(v)+e_i} X^{t(v)+e_i}$. So,

$$d_{t(v)} = (t_i(v) + 1)\, a_{\ell(v)}^{t(v)+e_i}$$

where $\ell(v^+) = i$ and $e_i$ is the standard unit vector. This gives us

$$c_{l,L} = \sum_{\substack{t:V(L)\to\mathbf{N}^n \\ \sum t(v)=l}} \prod_{v\in V(L)} (t_{\ell(v^+)}(v) + 1)\, a_{\ell(v)}^{t(v)+e_{\ell(v^+)}}$$

Starting with the expression from Claim 3, we get

$$P_r(J(H)) = (-1)^r \sum_{\substack{L\in\mathcal{L} \\ |V(L)|=r}} P_L = (-1)^r \sum_{\substack{L\in\mathcal{L} \\ |V(L)|=r}} \operatorname{sgn}(\tau) \sum_{|l|\geq r} c_{l,L} X^l$$

$$= (-1)^r \sum_{\substack{L\in\mathcal{L} \\ |V(L)|=r}} \operatorname{sgn}(\tau) \sum_{|l|\geq r} \sum_{\substack{t:V(L)\to\mathbf{N}^n \\ \sum t(v)=l}} \prod_{v\in V(L)} (t_{\ell(v^+)}(v) + 1)\, a_{\ell(v)}^{t(v)+e_{\ell(v^+)}} X^l$$

where, again, $\tau \in \overline{\mathcal{S}_r}$ arises from $L = [(G, \ell)]$. That is,

$$P_r(J(H)) = (-1)^r \sum_{|l|\geq r} h_{r,l} X^l \tag{3}$$

where

$$h_{r,l} = \sum_{\substack{L\in\mathcal{L} \\ |V(L)|=r}} \sum_{\substack{t:V(L)\to\mathbf{N}^n \\ \sum t(v)=l}} \operatorname{sgn}(\tau) \prod_{v\in V(L)} (t_{\ell(v^+)}(v) + 1)\, a_{\ell(v)}^{t(v)+e_{\ell(v^+)}} \tag{4}$$

Note, this expresses $h_{r,l}$ as a polynomial in $\{a_i^k\}$.

We want to simplify this expression, so we introduce the concept of extending $L$ by $t$. This is, again, parallel to the exposition in Wright's paper, [5]. Given $L$ and $t : V(L) \longrightarrow \mathbf{N}^n$ with $\sum t(v) = l$, we extend $L$ by $t$ as follows. For each $v \in V(L)$, add $|t(v)|$ children to $v$ with $t_j(v)$ of them labeled $j$. At this point, it does not matter in which order we place the children. Call this extended graph $L^t$. Notice that, whereas $L$ consists of cycle nodes only, $L^t$ is a graph of cycles with added leaves (nodes with no children). Thus $L^t \notin \mathcal{L}$ for non-trivial $t$.

We want to rewrite equation (4) summing over pairs $(L, t)$, and we want none of the factors $a_{\ell(v)}^{t(v)+e_{\ell(v^+)}}$ to be zero. Thus, we specify that $|t(v)+e_{\ell(v^+)}| \geq 2$ for all $v \in V(L)$, since deg$H \geq 2$ (see equation (2)). So, in $L^t$, we know that each non-leaf node has child-type $t(v) + e_{\ell(v^+)}$ and $|t(v) + e_{\ell(v^+)}| \geq 2$, hence $|t(v)| \geq 1$. Note, here, that the definition of $\ell(v^+)$ specifies it to be the label on the child of $v$ that is the cycle node (see Notation 2), so the definition remains unambiguous even when we add children to $v$. We can use this to re-write equation (4) as

$$h_{r,l} = \sum_{L,t} \mathrm{sgn}(\tau) \prod_{v \in V(L)} (t_{\ell(v^+)}(v) + 1)\, a_{\ell(v)}^{t(v)+e_{\ell(v^+)}}. \tag{5}$$

where $|V(L)| = r$, $t : V(L) \longrightarrow \mathbf{N}^n$, $|t(v)| \geq 1$ for all $v$, and $\sum t(v) = l$. Note again, the product on the inside does not depend on the order in which we placed the new children.

In $L^t$, it makes sense to talk about *leaf-type*. A graph has leaf-type $l = (l_1, \ldots, l_n)$ if there are $l_i$ leaves labeled $i$. We also note that two graphs of this type are isomorphic, namely $L_1^{t_1} \cong L_2^{t_2}$, precisely when $L_1$ and $L_2$ are isomorphic graphs in $\mathcal{L}$ and when $t_1 = t_2$. Then, we let $\mathbb{L}_l$ be the set of isomorphism classes of graphs where each vertex has indegree 1 and each non-leaf vertex has precisely one non-leaf child and at least one leaf child. Further, the labels on all non-leaf vertices are unique among the non-leaf nodes, and the graph has leaf-type $l$. Thus, each $L^t \in \mathbb{L}_l$ for some $l$, and each $R \in \mathbb{L}_l$ has the form $L^t$ for some $L$ and some $t$ with $\sum t(v) = l$. Denote by $R^*$ the graph $R$ with the leaves removed. For $R = L^t$, then, $R^* = L$.

We want to rewrite equation (5), summing over $\mathbb{L}_l$ instead of $(L, t)$. The question is, how many pairs $(L, t)$, with $L \in \mathcal{L}$ and $t : V(L) \longrightarrow \mathbf{N}^n$, give rise to the same $R \in \mathbb{L}_l$? The answer is 1, since in order for $L_1^{t_1} = L_2^{t_2} = R$, we must have $L_1 = L_2$ in $\mathcal{L}$ and $t_1 = t_2$. Since $t(v) + e_{\ell(v^+)}$ is the child-type of $v$

in $R$, relabel $t(v) + e_{\ell(v^+)}$ as $k(v)$. We write equation (5) as

$$h_{r,l} = \sum_{\substack{R \in \mathbb{L}_l \\ |V(R^*)|=r}} \operatorname{sgn}(\tau) \prod_{v \in V(R^*)} k_{\ell(v^+)}(v) a_{\ell(v)}^{k(v)} \tag{6}$$

where $k_{\ell(v^+)}(v)$ in $R$ is $t_{\ell(v^+)}(v) + 1$ in $(L, t)$.

Now that our graphs have leaves, it makes sense to define *planar graphs*, $Q$. A planar graph is simply a graph with a total ordering placed on the children of each vertex, such that $u \leq u'$ whenever $\ell(u) \leq \ell(u')$. We denote by $|Q|$ the underlying element $R \in \mathbb{L}_l$. Also, we let $\mathbb{Q}_l$ be the set of isomorphism classes of planar graphs, $Q$, having leaf-type $l$, such that $|Q| \in \mathbb{L}_l$. Notice that in $\mathbb{Q}_l$, the placement of the leaves now matters – we get different planar graphs if the leaves are inside or outside the loop of the graph, as this indicates a different ordering on the children. Two planar graphs, $Q_1, Q_2$, are isomorphic as planar graphs if there is an isomorphism of graphs $Q_1 \longrightarrow Q_2$ that preserves labels and orderings of children (and hence preserves child-types). Note, also, that each connected component of $Q$ is a special kind of wreath. Namely, the loop of the component has no repeated labels, and the branches of the component consist solely of leaves.

To write equation (6) as a sum over $\mathbb{Q}_l$, we must ask, for a fixed $R \in \mathbb{L}_l$, how many isomorphism classes of planar graphs $Q \in \mathbb{Q}_l$ have $|Q| = R$? Only those planar graphs having different orderings on the cycle nodes will be pairwise non-isomorphic. The number of isomorphism classes, then, is $\prod_{v \in V(|Q|^*)} k_{\ell(v^+)}(v)$. This allows us to write equation (6) as

$$h_{r,l} = \sum_{\substack{Q \in \mathbb{Q}_l \\ |V(|Q|^*)|=r}} \operatorname{sgn}(\tau) \frac{1}{\prod_{v \in V(|Q|^*)} k_{\ell(v^+)}(v)} \prod_{v \in V(|Q|^*)} k_{\ell(v^+)}(v) a_{\ell(v)}^{k(v)},$$

that is,

$$h_{r,l} = \sum_{\substack{Q \in \mathbb{Q}_l \\ |V(|Q|^*)|=r}} \operatorname{sgn}(\tau) \prod_{v \in V(|Q|^*)} a_{\ell(v)}^{k(v)} \tag{7}$$

Equation 7 together with equation 3 gives us a nice expression for $P_r(J(H))$ in terms of the coefficients of $H$.

## 4. Relating Raney Coefficients and $P_r(J(H))$

The main goal of this section is to show that the polynomials $e_p^q$ and $d_p^q$, defined below, are congruent modulo the ideal generated by the polynomials $h_l$

in $\mathbb{Z}[\{a_i^k\}]$, $|k| \geq 2$. More specifically, we first fix the dimension to be $n$. For a forest, we denote the root-type of the forest as an $n$-tuple, $p$, and the leaf-type as an $n$-tuple, $q$. Let $I_p^q$ denote the set of inventories, $\alpha$, with root-type $p$ and leaf-type $q$, such that all non-leaves have at least 2 children. That is, if $|k| = 1$, then $\alpha_i^k = 0$ (see Section 2.1 for notation). This forces $I_p^q$ to be finite. Note that, if $|p| > |q|$, then $I_p^q = \emptyset$, because every tree must have at least one leaf.

For $\alpha \in I_p^q$, let

$$a^{\alpha^*} = \prod_{i=1}^{n} \prod_{|k|\geq 2} (a_i^k)^{\alpha_i^k}$$

and note that this product has a finite number of non-trivial factors. Recall, from Section 2.2, we defined

$$e_p^q = \sum_{\alpha \in I_p^q} R(\alpha) a^{\alpha^*}.$$

Let

$$d_p^q = \sum_{\alpha \in I_p^q} M(\alpha) a^{\alpha^*}.$$

Each of $e_p^q$ and $d_p^q$ is a polynomial since $I_p^q$ is finite.

Again, as in Section 2.2, we have

$$M(\alpha_i) = \frac{\sigma(\alpha_i)!}{\prod(\alpha_i^k!)}$$

and

$$M(\alpha) = \prod_{i=1}^{n} M(\alpha_i).$$

For $h_{r,l}$ as defined in equation 7, let

$$h_l = \sum_{r=1}^{n} (-1)^{r+1} h_{r,l}.$$

Finally, we define the ideal in $\mathbb{Z}[\{a_i^k\}]$ generated by the coefficients of $P_r(J(H))$,

$$\mathcal{I} = < h_l >_{|l|\geq 1} .$$

Then,

**Theorem 4.** *For a given $p$, $q$,*

$$d_p^q - e_p^q = \sum_l d_{p+l}^q \; h_l$$

*and this is a finite sum.*

First, we will express the left-hand side of this equation using our work in Section 2.3. Then, we will find an expression for the $h_l$ in similar terms. This will allow us to prove the equality. Once the equality is shown, finitude follows because $d_{p+l}^q = 0$ when $p + l > q$. The goal of this section is now an easy corollary.

**Corollary 5.** $d_p^q - e_p^q \in \mathcal{I}$

*Proof.* This is immediate from the finitude of the sum. □

*Proof.* (of Theorem 4) By definition,

$$d_p^q - e_p^q = \sum_{\alpha \in I_p^q} (M(\alpha) - R(\alpha)) \; a^{\alpha^*} \tag{8}$$

For a fixed $\alpha \in I_p^q$, let $L_\alpha$ be the shaded set consisting of $\alpha_i^k$ nodes of color $i$ having child-type $k$, with duplicate elements shaded as before. We know that $(M(\alpha) - R(\alpha))$ counts the number of distinct unshaded failing sorted permutations. Thus, by equation (1) and Section 2.3, equation (8) is equal to

$$\sum_{\alpha \in I_p^q} \frac{1}{\prod(\alpha_i^k!)} \left[ \sum_\tau (-1)^{e(\tau)+1} \Big| \{A \in A_\alpha | \Phi(A) \text{ contains a set of} \right.$$

$$\left. \text{cycle} - \text{type } \tau\} \Big| \right] a^{\alpha^*} \tag{9}$$

Next, we need to re-organize this slightly. Instead of grouping by $e(\tau)$, the number of disjoint cycle-types, we want to group according to $r$, the number of nodes in these cycle-types. Notice that equation (9) is equivalent to

$$\sum_{\alpha \in I_p^q} \frac{1}{\prod(\alpha_i^k!)} \sum_{r=1}^{n} \sum_{\tau \in \overline{\mathcal{S}_r}} (-1)^{r+1} \mathrm{sgn}(\tau) \sum_{*} a^{\alpha^*} \tag{10}$$

where $*$ is the set of all shaded arbors $A \in A_\alpha$ such that $\Phi(A)$ contains a set of cycle-type $\tau$.

Next, we ask, for a fixed $\alpha$ and $\tau$, how many shaded $L_\alpha$-sorted permutations build to arbors, $A$, such that $\Phi(A)$ contains a set of cycles of type $\tau$? The answer amounts to the product of $(i)$ the number of ways to choose elements of $L_\alpha$ to associate to $\tau$'s labels, $(ii)$ the number of ways to choose the edges, and $(iii)$ the number of arrangements of the remaining elements. Given $i \in \ell(\tau)$, let $s_i$ be an $n$-tuple. Let $\overline{s}$ be a set containing, for each $i \in \ell(\tau)$, one choice of $n$-tuple for $s_i$. Let $T_\tau$ be the set of all such $\overline{s}$. Note that $T_\tau$ is independent of $\alpha$. Recalling that we denote a letter, $i^k$, by its label with its child-type as a superscript (see Section 2.3), given $\overline{s}$, there are $\prod_{i\in\ell(\tau)} \alpha_i^{s_i}$ ways to choose elements $i^{s_i}$ of $L_\alpha$. This product will be zero if any $\alpha_i^{s_i} = 0$. Denote by

$$\delta = \prod_{i\in\ell(\tau)} \alpha_i^{s_i},$$

the number of ways to make this choice. So, $\sum_{\overline{s}\in T_\tau} \delta$ is part $(i)$ of our product.

Once these elements have been chosen from $L_\alpha$, there are $\prod_{i\in\ell(\tau)}(s_i)_{\tau(i)}$ ways to form the cycles of $\tau$, by designating which edge of each cycle node belongs to the cycle. This is part $(ii)$ of our product. Finally, let $\overline{L_{\overline{s}}}$ be the shaded set consisting of elements of $L_\alpha$ omitting those elements chosen from $L_\alpha$ once $\overline{s}$ is specified. Let $e_i$ be the standard unit vector and $l$ be the $n$-tuple $l = \sum_{i\in\ell(\tau)}(s_i - e_{\tau(i)})$. Note, since we are summing over the letters of $\tau$, that $l = \sum_{i\in\ell(\tau)}(s_i - e_i)$.

**Note 6.** $\overline{L_{\overline{s}}}$ has inventory $\overline{\alpha} \in I^q_{p+l}$.

There are $\prod_{i=1}^n(\sigma(\overline{\alpha}_i)!)$ distinct shaded sorted permutations formed using $\overline{L_{\overline{s}}}$. So, we have part $(iii)$ of our product. Hence, the number of $L_\alpha$-sorted-permutations that build to an arbors, $A$, having as a subset of $\Phi(A)$ cycles of type $\tau$ is

$$\sum_{\overline{s}\in T_\tau} \prod_{i=1}^{n}(\sigma(\overline{\alpha}_i)!)\delta \prod_{i\in\ell(\tau)} (s_i)_{\tau(i)}.$$

Now, we can re-write equation (10) using the following notation. Let

$$m_{(\tau,\overline{s})} = \prod_{i\in\ell(\tau)} a_i^{s_i}$$

be the monomial formed from the set of elements of $\tau$ using $\overline{s}$ as their child-types. Let

$$m_{\overline{L_{\overline{s}}}} = \prod_{i\notin\ell(\tau)} (a_i^k)^{\alpha_i^k} \prod_{\substack{i\in L(\tau)\\ s_i\notin\overline{s}}} (a_i^{s_i})^{\alpha_i^{s_i}} \prod_{\substack{i\in\ell(\tau)\\ s_i\in\overline{s}}} (a_i^{s_i})^{\alpha_i^{s_i}-1}.$$

Letting

$$\overline{X} = \delta \prod_{i=1}^{n} (\sigma(\overline{\alpha}_i)!) m_{\overline{L_{\overline{s}}}} \prod_{i \in \ell(\tau)} (s_i)_{\tau(i)} m_{(\tau,\overline{s})},$$

we can rewrite equation (10) as

$$\sum_{\alpha \in I_p^q} \frac{1}{\prod(\alpha_i^k!)} \sum_{r=1}^{n} \sum_{\tau \in \overline{\mathcal{S}_r}} (-1)^{r+1} \text{sgn}(\tau) \sum_{\overline{s} \in T_\tau} \overline{X} \tag{11}$$

Next, we group by $l = \sum_{i \in \ell(\tau)} (s_i - e_i)$. Recall, each node must have zero or at least two children. If $|l| < r$, then some $s_i$ has $|s_i| = 1$ and, therefore, $\alpha_i^{s_i} = 0$, making $\overline{X} = 0$. So, we may sum over all $l$, as the inappropriate $l$ lend only zeroes to the sum. equation (11) then becomes

$$\sum_{\alpha \in I_p^q} \frac{1}{\prod(\alpha_i^k!)} \sum_{r=1}^{n} \sum_{\tau \in \overline{\mathcal{S}_r}} \sum_{l} (-1)^{r+1} \text{sgn}(\tau) \sum_{**} \overline{X}$$

where $**$ sums over the set of $\overline{s} \in T_\tau$ such that $l = \sum_{i \in \ell(\tau)} (s_i - e_i)$.

Since $r$, $\tau$, and $\overline{s}$ do not depend on $\alpha$, we may switch the order of summation to get

$$\sum_{r=1}^{n} \sum_{\tau \in \overline{\mathcal{S}_r}} \sum_{l} (-1)^{r+1} \text{sgn}(\tau) \sum_{**} \sum_{\alpha \in I_p^q} \frac{1}{\prod(\alpha_i^k!)} \overline{X} \tag{12}$$

Notice that

$$\begin{aligned}
\sum_{\alpha \in I_p^q} \frac{1}{\prod(\alpha_i^k!)} \overline{X} &= \sum_{\alpha \in I_p^q} \frac{1}{\prod(\alpha_i^k!)} \delta \prod_{i=1}^{n} (\sigma(\overline{\alpha}_i)!) m_{\overline{L_{\overline{s}}}} \prod_{i \in \ell(\tau)} (s_i)_{\tau(i)} m_{(\tau,\overline{s})} \\
&= \sum_{\overline{\alpha} \in I_{p+l}^q} M(\overline{\alpha}) m_{\overline{L_{\overline{s}}}} \prod_{i \in \ell(\tau)} (s_i)_{\tau(i)} m_{(\tau,\overline{s})} \\
&= d_{p+l}^q \prod_{i \in \ell(\tau)} (s_i)_{\tau(i)} m_{(\tau,\overline{s})}
\end{aligned}$$

Indeed, the last equality is the definition of $d_{p+l}^q$. The middle equality stems from the fact that, for each $\alpha \in I_p^q$ such that $\delta \neq 0$, we have

$$\frac{1}{\prod(\alpha_i^k!)} \delta \prod_{i=1}^{n} (\sigma(\overline{\alpha}_i)!) = \frac{1}{\prod(\overline{\alpha}_i^k!)} \prod_{i=1}^{n} (\sigma(\overline{\alpha}_i)!) = M(\overline{\alpha}),$$

and by Note 6, $\overline{\alpha} \in I^q_{p+l}$.

So, equation (12) is

$$\sum_{r=1}^{n}(-1)^{r+1}\sum_{\tau\in\overline{\mathcal{S}_r}}\operatorname{sgn}(\tau)\sum_{l} d^q_{p+l}\sum_{**}\prod_{i\in\ell(\tau)}(s_i)_{\tau(i)}m_{(\tau,\overline{s})} \tag{13}$$

where $**$ sums over all $\overline{s} \in T_\tau$ with $l = \sum_{i\in\ell(\tau)}(s_i - e_i)$.

Next, we turn to the expression $h_l$. Recall that, in equation (7), we found

$$h_{r,l} = \sum_{\substack{Q\in\mathbb{Q}_l \text{ with}\\ |V(|Q|^*)|=r}} \operatorname{sgn}(\tau)\prod_{v\in V(|Q|^*)} a^{k(v)}_{\ell(v)}$$

Now, if there are $r$ cycle nodes, a labeling on $Q^*$ amounts to a choice $\tau \in \overline{\mathcal{S}_r}$ such that $\tau$ and $Q$ have the same cycle-type. Placing the labeled leaves on $Q \in \mathbb{Q}_l$ amounts to choosing $n$-tuples, $s_i$ for each $i \in \ell(\tau)$ such that $l = \sum_{i\in\ell(\tau)}(s_i - e_i)$.

For fixed $\tau$, we denote the set of such $\overline{s} = \{$one choice for each $s_i | i \in \ell(\tau)\}$ as $\overline{T}_\tau$. Using these, we get that

$$h_{r,l} = \sum_{\tau\in\overline{\mathcal{S}_r}}\operatorname{sgn}(\tau)\sum_{\overline{s}\in\overline{T}_\tau}\left(\prod_{i\in\ell(\tau)}(s_i)_{\tau(i)}\right)m_{(\tau,\overline{s})} \tag{14}$$

Notice that, for a fixed $l$, $\overline{T}_\tau$ becomes the same set as those $\overline{s}$ in $**$ from equation (13). Thus, putting equations (13) and (14) together gives us

$$d^q_p - e^q_p = \sum_{r=1}^{n}(-1)^{r+1}\sum_{l} d^q_{p+l}h_{r,l}$$

By definition of $h_l$, this equals

$$\sum_{l} d^q_{p+l}h_l$$

as stated. □

## 5. Conclusion

We have now shown that the inverse formula for $F$ found in [2] simplifies under the Jacobian Hypothesis. Specifically, let $F = X - H$, such that $H_i = \sum_{|k|\geq 2} \overline{a}_i^k X^k$ for $\overline{a}_i^k$ in any commutative ring, $R$. Then, with the addition of the Jacobian hypothesis, $j(F) = 1$, we have

$$X_1^{p_1} \cdots X_n^{p_n} = \sum_{q=(q_1,\ldots,q_n)} d_p^q F_1^{q_1} \cdots F_n^{q_n}$$

where $d_p^q(\overline{a})$ results from specializing $a_i^k$ to $\overline{a}_i^k$ in $d_p^q$. So, under the Jacobian hypothesis, we can replace the polynomials $e_p^q$ by $d_p^q$, which are given by a simpler formula.

Thus, to prove the Jacobian hypothesis, it suffices to work with this simplified expression for the inverse of $F$.

## References

[1] H. Bass, E. Connell, D. Wright, The Jacobian conjecture: Reduction of degree and formal expansion of the inverse, *Bulletin of the American Mathematical Society*, **7**, No. 2 (1982), 287-330.

[2] C. Ching-An Cheng, J. McKay, J. Towber, S. Sui-Sheng Wang, D. Wright, Reversion of power series and the extended raney coefficients, *Transactions of the American Mathematical Society*, **349** (1997), 1769-1782.

[3] K. Lampe, A counting formula for labeled, rooted forests, *Journal of Algebraic Combinatorics*, **15**, No. 1 (2002), 71-97.

[4] G. Raney, Functional composition patterns and power series reversion, *Transactions of the American Mathematical Society*, **94** (1960), 441-451.

[5] D. Wright, The tree formulas for reversion of power series, *Journal of Pure and Applied Algebra*, **57** (1989), 191-211.

[6] D. Wright, Formal inverse expansion and the Jacobian conjecture, *Journal of Pure and Applied Algebra*, **48** (1987), 199-219.

## 6. Appendix

We provide here some examples illustrating use of the algorithms referenced in Section 2.3. Given a shaded set, $S$, we first build an arbor, given a sorted perumutation from $S$. Then, given a cycle and a complementary sorted permutation from $S$ to demonstrate the inverse procedure.

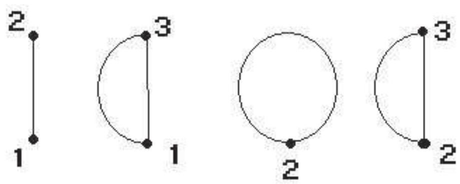
We begin with a shaded set

$$S = \{1_{001}, 1_{010}, 2_{010}, 2_{001}, 2_{000}, 3_{010}, 3_{100}\}$$

One sorted permutation from $S$ is

$$W = (1_{010}, 1_{001})(2_{000}, 2_{010}, 2_{001})(3_{100}, 3_{010}).$$

This corresponds to Arbor 1 in Figure 1 and has $\Phi(W) = \{(2_{001}, 3_{010})\}$. Here, the cycle associated to the rightmost wreath is put in the image set by Step 2 of $\Phi$ in [3]. The other cycles are not disjoint from this rightmost wreath's direct loop, so they are not part of $\Phi(W)$.




Next, suppose we are given the cycle $c = (2_{001}, 3_{010})$ and complementary sorted permutation $P = (1_{010}, 1_{001})(2_{010}, 2_{000})(3_{100})$. Following the inverse procedure described in [3] again yields the arbor shown in Figure **??**.