Available online at www.sciencedirect.com
SCIENCE @ DIRECT®
Discrete Applied Mathematics 148 (2005) 246–255
DISCRETE APPLIED MATHEMATICS
www.elsevier.com/locate/dam

Note

Faà di Bruno's formula, lattices, and partitions

L. Hernández Encinas[a], A. Martín del Rey[b], J. Muñoz Masqué[a]

[a] *Instituto de Física Aplicada, CSIC, C/ Serrano 144, 28006-Madrid, Spain*
[b] *Departamento de Matemática Aplicada, E.P.S., Universidad de Salamanca, C/ Sto. Tomás s/n, 05003-Ávila, Spain*

Received 28 May 2002; received in revised form 18 May 2004; accepted 15 February 2005
Available online 29 April 2005

Abstract

The coefficients of $g^{(s)}$ in expanding the rth derivative of the composite function $g \circ f$ by Faà di Bruno's formula, is determined by a Diophantine linear system, which is proved to be equivalent to the problem of enumerating partitions of a finite set of integers attached to r and s canonically.

MSC: primary 11D04; secondary 05A17; 11D45; 11Y50; 15A36; 68R05

Keywords: Derivatives of composite functions; Diophantine equations with lower and upper bounds; Integer partitions; Knapsack systems; Lattices

1. Introduction

Throughout this paper, $\mathbb{Z}$, $\mathbb{Z}^+$, and $\mathbb{N}$ denotes the set of integers, positive integers, and non-negative integers, respectively. Let U, V be two open intervals in $\mathbb{R}$, and let $f: U \to V$, $g: V \to \mathbb{R}$ be two differentiable functions of class C^r, $r > 0$, such that $f(U) \subseteq V$. The derivatives of the composition is given by the Faà di Bruno formula (e.g., see [2,4,6,9–12]):

$$(g \circ f)^{(r)} = \sum_{s=1}^{r} (g^{(s)} \circ f) \sum_{\substack{\sum_{i=1}^{r} e_i = s \\ \sum_{i=1}^{r} i e_i = r}} \frac{r!}{e_1! \cdots e_r! (1!)^{e_1} \cdots (r!)^{e_r}} f^{(1)^{e_1}} \cdots f^{(r)^{e_r}}. \quad (1)$$

E-mail addresses: luis@iec.csic.es (L.H. Encinas), delrey@usal.es (A.M. del Rey), jaime@iec.csic.es (J.M. Masqué).

0166-218X/$ - see front matter
doi:10.1016/j.dam.2005.02.009

For the sake of simplicity, we denote the coefficients in formula (1) as follows:

$$C_{\mathbf{e}} = r! \prod_{i=1}^{r} \frac{1}{e_i!(i!)^{e_i}}, \quad \mathbf{e} = (e_1, \ldots, e_r) \in \mathbb{N}^r.$$

Moreover, Bell polynomials are defined as follows (e.g., see [11, 2.8]):

$$B_{r,s}(\mathbf{x}) = \sum_{\substack{\sum_{i=1}^{r} e_i = s \\ \sum_{i=1}^{r} i e_i = r}} \frac{r!}{e_1! \cdots e_r!} \left(\frac{x_1}{1!}\right)^{e_1} \cdots \left(\frac{x_r}{r!}\right)^{e_r},$$

$$B_r(\mathbf{x}, \mathbf{y}) = \sum_{s=1}^{r} y_s B_{r,s}(\mathbf{x}).$$

Faà di Bruno's formula and Bell polynomials are related as follows. Let $x_i = f^{(i)}$ and $y_i = g^{(i)}$. Then

$$(g \circ f)^{(r)} = B_r(f^{(1)}, \ldots, f^{(r)}, g^{(1)}, \ldots, g^{(r)}).$$

Accordingly, Faà di Bruno's formula and Bell polynomials determine each other tautologically.

The problem of computing the rth derivative of a composite function is thus reduced to compute the coefficients $C_{\mathbf{e}}$, where $\mathbf{e} \in \mathbb{N}^r$ should satisfy the following Diophantine linear system:

$$E_{r,s} \quad \begin{cases} E^1_{r,s} \equiv \sum_{i=1}^{r} e_i = s, \\ E^2_{r,s} \equiv \sum_{i=1}^{r} i e_i = r \end{cases} \tag{2}$$

with $1 \leqslant s \leqslant r$.

It is now clear that the coefficients $C_{\mathbf{e}}$ are known once the system $E_{r,s}$ is solved in $\mathbb{N}^r$, as the rest of operations involved are products of powers of factorials and their inverses.

The goal of this paper is to prove that solving system (2) is equivalent to computing the integer partitions of a finite set of integers $S_{r,s}$ (see the inequalities (12)–(14) below) associated with r, s. This provides an explicit parameterization of the set of indices involved in a specific coefficient, once the partition problem is solved (see Theorem 4 below). The advantage is that it allows us to implement the computation of a single term in Faà di Bruno's formula without the need for computing the rest of terms. Symbolic manipulation programs can be employed in order to obtain expansions of the derivatives of composite functions, but such programs are obliged to compute all terms necessarily. Hence the programs overflow even for rather reasonable values of the derivative order. In practice, however, one is often confined to control only a part of the coefficients.

As is remarked in [5], no specific algorithm is known to compute the coefficients in Faà di Bruno's formula, apart from the general Horner iteration (e.g., see [7, 5.2]), which allows