

# $m$ -fold Hypergeometric Solutions of Linear Recurrence Equations Revisited

Peter Horn, Wolfram Koepf, Torsten Sprenger

*Institute of Mathematics, University of Kassel, D-34132 Kassel, Germany*

---

## Abstract

We present two algorithms to compute  $m$ -fold hypergeometric solutions of linear recurrence equations for the classical shift case and for the  $q$ -case, respectively. The first is an  $m$ -fold generalization and  $q$ -generalization of the algorithm by van Hoeij (1998a), Cluzeau and van Hoeij (2005) for recurrence equations. The second is a combination of an improved version of the algorithms by Petkovšek (1992), Abramov et al. (1998) for recurrence and  $q$ -recurrence equations and the  $m$ -fold algorithm from Petkovšek and Salvy (1993) for recurrence equations. We will refer to the classical algorithms as *van Hoeij* or *Petkovšek* respectively.

To formulate our ideas, we first need to introduce an adapted version of an  $m$ -fold Newton polygon and its characteristic polynomials for the classical case and  $q$ -case, and to prove the important properties in this case. Using the data from the Newton polygon, we are able to present efficient  $m$ -fold versions of the van Hoeij and Petkovšek algorithms for the classical shift case and for the  $q$ -case, respectively. Furthermore, we show how one can use the Newton polygon and our characteristic polynomials to conclude for which  $m \in \mathbb{N}$  there might be an  $m$ -fold hypergeometric solution at all. Again by using the information obtained from the Newton polygon, the presentation of the  $q$ -Petkovšek algorithm can be simplified and streamlined.

Finally, we give timings for the ‘classical’  $q$ -Petkovšek, our  $q$ -van Hoeij and our modified  $q$ -Petkovšek algorithm on some classes of problems and we present a *Maple* implementation of the  $m$ -fold algorithms for the  $q$ -case.

*Key words:* right factors of  $m$ -fold hypergeometric type, linear recurrence operators,  $m$ -fold hypergeometric solutions, linear recurrence equations, linear  $q$ -recurrence equations, Newton polygon

---

## 1. Introduction

For the whole paper let  $\mathbb{F}$  be a computable field of characteristic zero,  $x$  be an indeterminate over  $\mathbb{F}$  and  $\mathbb{F}(x)$  the field of rational functions in  $x$  over  $\mathbb{F}$ . We consider

---

*Email addresses:* [horn@math.uni-kassel.de](mailto:horn@math.uni-kassel.de) (Peter Horn), [koepf@math.uni-kassel.de](mailto:koepf@math.uni-kassel.de) (Wolfram Koepf), [sprenger@math.uni-kassel.de](mailto:sprenger@math.uni-kassel.de) (Torsten Sprenger).

$\sigma : \mathbb{F}(x) \rightarrow \mathbb{F}(x)$  to be an automorphism fixing  $\mathbb{F}$ . In some cases we need to specify the considered automorphism  $\sigma$ . In this paper we deal especially with two automorphisms, namely the ‘classical’ *shift*

$$\varepsilon : \mathbb{F}(x) \rightarrow \mathbb{F}(x), \quad x \mapsto x + 1$$

and the *q-shift*

$$\varepsilon_q : \mathbb{F}(x) \rightarrow \mathbb{F}(x), \quad x \mapsto qx.$$

In the *q*-case we require that  $\mathbb{F} = \mathbb{K}(q)$ , where  $\mathbb{K}$  is a computable field of characteristic zero,  $q$  an indeterminate over  $\mathbb{K}$  and  $\mathbb{K}(q)$  the field of rational functions in  $q$  over  $\mathbb{K}$ . Let furthermore  $\mathbb{F}(x)[\sigma]$  denote the non-commutative algebra of recurrence operators with rational function coefficients. Here, we notationally identify the operator  $\sigma$  and the automorphism  $\sigma$ . We always write occurring operators  $L \in \mathbb{F}(x)[\sigma]$  in expanded normal form

$$L = \sum_{i=0}^n a_i \sigma^i \quad \text{with} \quad a_i \in \mathbb{F}(x) \quad \text{and} \quad a_0 \neq 0, a_n \neq 0.$$

Then  $n$  is called the *order* of the operator  $L$ . Throughout the paper we omit the argument  $x$  in the notation of polynomials and functions. W.l.o.g. we assume that the coefficients  $a_i$  are polynomials.

In the following we are interested in determining right factors of *m-fold hypergeometric type*, i.e.  $\sigma^m - r$  with  $r \in \mathbb{F}(x)$ , of a given linear recurrence operator  $L \in \mathbb{F}[x][\sigma]$ . The computation of right factors of *m-fold hypergeometric type* of  $L$  corresponds one-to-one to the computation of *m-fold hypergeometric term* solutions of the recurrence equation  $L(u) = 0$ , being solutions  $u$  for which

$$\text{cert}_m^\sigma(u) := \frac{\sigma^m(u)}{u} \in \mathbb{F}(x)$$

holds. We call  $\text{cert}_m^\sigma(u)$  the *m-fold certificate*<sup>1</sup> of  $u$  w.r.t.  $\sigma$ . Furthermore, we assume that all *m-fold hypergeometric terms*  $u$  considered are *primitive*, i.e. no rational functions  $r_0, \dots, r_{m-2}$  in  $x$  exist with  $\sigma^{m-1}(u) + r_{m-2}\sigma^{m-2}(u) + \dots + r_1\sigma(u) + r_0u = 0$ . Non-primitive *m-fold hypergeometric solutions* are considered in Hendriks and Singer (1998).

**Example 1.** Let  $u$  be a hypergeometric term with  $r = \frac{\sigma(u)}{u} \in \mathbb{F}(x)$ . Then  $u$  is also an *m-fold hypergeometric term* with  $m \neq 1$  since

$$\frac{\sigma^m(u)}{u} = \frac{\sigma^m(u)}{\sigma^{m-1}(u)} \frac{\sigma^{m-1}(u)}{\sigma^{m-2}(u)} \dots \frac{\sigma^2(u)}{\sigma(u)} \frac{\sigma(u)}{u} = \sigma^{m-1}(r)\sigma^{m-2}(r) \dots \sigma(r)r \in \mathbb{F}(x),$$

but  $u$  is *not* primitive because of  $\sigma(u) - ru = 0$ .

**Definition 1.** Let  $\gamma \in \overline{\mathbb{F}}$  and  $k \in \mathbb{Z}$ . Then  $\gamma^{\sigma^k}$  denotes the root of  $\sigma^k(x - \gamma)$ . For  $\alpha, \beta \in \overline{\mathbb{F}}$  define the equivalence relation

$$\alpha \sim_m^\sigma \beta \quad :\iff \quad \alpha = \beta^{\sigma^{mk}} \quad \text{for some } k \in \mathbb{Z}.$$

---

<sup>1</sup> If  $m = 1$ , then we omit the term *m-fold* and call a solution just *hypergeometric term* and  $\text{cert}^\sigma(u)$  its *certificate*.

With  $[\alpha]_m^\sigma$  we denote the equivalence class of  $\alpha$  in  $\overline{\mathbb{F}}/\sim_m^\sigma$ . For an  $m$ -fold hypergeometric term  $u$  we define the *local type* at  $[\alpha]_m^\sigma \in \overline{\mathbb{F}}/\sim_m^\sigma$  to be

$$\text{ltype}_{[\alpha]_m^\sigma}(u) = \sum_{\beta \in [\alpha]_m^\sigma} (\text{root order of } \text{cert}_m^\sigma(u) \text{ in } \beta) - (\text{pole order of } \text{cert}_m^\sigma(u) \text{ in } \beta).$$

Here it is noteworthy that  $\text{ltype}_{[\alpha]_m^\sigma}(u)$  is well defined and it is a nonzero integer only for finitely many classes  $[\alpha]_m^\sigma$ . For  $\infty$  we define

$$\text{ltype}_\infty(u) = (c_\infty, v_\infty) \text{ with } \text{cert}_m^\sigma(u) = c_\infty x^{-v_\infty} (1 + \mathcal{O}(\frac{1}{x}))$$

using the series expansion of  $\text{cert}_m^\sigma(u)$  at  $\infty$ . In the  $q$ -case we have additionally

$$\text{ltype}_0(u) = (c_0, v_0) \text{ with } \text{cert}_m^{\varepsilon q}(u) = c_0 x^{v_0} (1 + \mathcal{O}(x))$$

as 0 behaves exceptionally like  $\infty$ , in the sense that  $0^{\varepsilon k} = 0$  for all  $k \in \mathbb{Z}$ . For this reason we call  $\infty$  and 0 (0 only in the  $q$ -case) *exceptional points*.

**Example 2.** We consider the shift case with the operator  $\varepsilon$  and the 2-fold hypergeometric term  $u$  with

$$\text{cert}_2^\varepsilon(u) = \frac{x(x-1)}{(x+1)^3(2x+1)}.$$

Then we have  $\text{ltype}_{[0]_2^\varepsilon}(u) = 1$ ,  $\text{ltype}_{[1]_2^\varepsilon}(u) = 1 - 3 = -2$  and  $\text{ltype}_{[\frac{1}{2}]_2^\varepsilon}(u) = -1$ . The local type of  $u$  in other ‘finite points’ is zero. For the local type at  $\infty$  we have  $\text{ltype}_\infty(u) = (\frac{1}{2}, 2)$  because of  $\text{cert}_2^\varepsilon(u) = \frac{1}{2} x^{-2} (1 + \mathcal{O}(\frac{1}{x}))$ . Note that if the rational function is the certificate of a hypergeometric term  $u$  then the equivalence class  $[0]_2^\varepsilon$  will coincide with  $[1]_2^\varepsilon$  and  $\text{ltype}_{[0]_1^\varepsilon}(u) = 1 + 1 - 3 = -1$ .

There is in fact a relation between the local types in ‘finite points’ and the local types in the exceptional points.

**Lemma 3** (Fuchsian relations). The following relations hold for an  $m$ -fold hypergeometric term  $u$  with  $\text{ltype}_\infty(u) = (c_\infty, v_\infty)$

$$0 = v_\infty + \sum_{[\alpha]_m^\varepsilon \in \overline{\mathbb{F}}/\sim_m^\varepsilon} \text{ltype}_{[\alpha]_m^\varepsilon}(u). \quad (1)$$

In the  $q$ -case we have

$$0 = v_0 + v_\infty + \sum_{[\alpha]_m^{\varepsilon q} \in \overline{\mathbb{F}}^\times/\sim_m^{\varepsilon q}} \text{ltype}_{[\alpha]_m^{\varepsilon q}}(u) \quad \text{and} \quad c_0 \sim_m^{\varepsilon q} c_\infty \prod_{[\alpha]_m^{\varepsilon q} \in \overline{\mathbb{F}}^\times/\sim_m^{\varepsilon q}} (-\alpha)^{\text{ltype}_{[\alpha]_m^{\varepsilon q}}(u)}, \quad (2)$$

where  $\text{ltype}_0(u) = (c_0, v_0)$ .

**Proof.** Let  $r = \frac{s}{t}$  be the  $m$ -fold certificate of  $u$  with  $s, t \in \mathbb{F}[x]$ . We consider the value  $\deg(s) - \deg(t)$ , which is obviously  $-v_\infty$ . For  $\sigma = \varepsilon$  we have

$$\deg(s) - \deg(t) = \sum_{[\alpha]_m^\varepsilon \in \overline{\mathbb{F}}/\sim_m^\varepsilon} \text{ltype}_{[\alpha]_m^\varepsilon}(u),$$

and for  $\sigma = \varepsilon_q$

$$\deg(s) - \deg(t) = v_0 + \sum_{[\alpha]_m^{\varepsilon_q} \in \overline{\mathbb{F}}^\times / \sim_m^{\varepsilon_q}} \text{ltype}_{[\alpha]_m^{\varepsilon_q}}(u),$$

hence (1) and the first equation of (2) follows. Applying Vieta's theorem to  $s$  and  $t$  the second relation of (2) follows<sup>2</sup>

$$\underbrace{\frac{\text{lcoeff}(s)}{\text{lcoeff}(t)}}_{c_\infty} \prod_{[\alpha]_m^{\varepsilon_q} \in \overline{\mathbb{F}}^\times / \sim_m^{\varepsilon_q}} (-\alpha)^{\text{ltype}_{[\alpha]_m^{\varepsilon_q}}(u)} \sim_m^{\varepsilon_q} \frac{\text{tcoeff}(s)}{\text{tcoeff}(t)} = c_0.$$

The representation depends on the choice of the representatives of  $[\alpha]_m^{\varepsilon_q}$  and therefore the terms are only identical modulo  $q$ -shifts.  $\square$

We give a definition for the *type* of an  $m$ -fold hypergeometric term similar to the classical definition (cf. Cluzeau and van Hoeij (2005)).

**Definition 2.** Two  $m$ -fold hypergeometric terms  $u, w$  are said to be *of the same type* if  $\frac{u}{w}$  is a rational function or, equivalently,  $\frac{\sigma^m(u)}{u} \frac{w}{\sigma^m(w)}$  is the  $m$ -fold certificate of a rational function.

The following simple theorem, which is crucial for the  $m$ -fold van Hoeij algorithm, connects the two type concepts.

**Theorem 3.** *Two  $m$ -fold hypergeometric terms are of the same type if and only if their local types in all exceptional points and all 'finite points' coincide.*

## 2. The $\sigma$ -Newton Polygon

In this section we consider the  $\sigma$ -Newton polygon in detail. This polygon can be easily constructed from the given recurrence operator  $L \in \mathbb{F}[x][\sigma]$  and it provides useful information about  $m$ -fold hypergeometric solutions of  $L$ .

### 2.1. Valuations

**Definition 4.** A mapping  $v : \mathbb{F}(x) \rightarrow \mathbb{R} \cup \{\infty\}$  is called a *valuation on  $\mathbb{F}(x)$* , if for all  $f, g \in \mathbb{F}(x)$  the following properties hold

- (a)  $v(f) = \infty \iff f = 0$
- (b)  $v(f \cdot g) = v(f) + v(g)$
- (c)  $v(f + g) \geq \min\{v(f), v(g)\}$
- (d)  $v(f) \neq v(g) \Rightarrow v(f + g) = \min\{v(f), v(g)\}$ .

We call a valuation *compatible* with the automorphism  $\sigma$ , if  $v(\sigma(f)) = v(f)$  is valid for all  $f \in \mathbb{F}(x)$ .

<sup>2</sup>  $\text{lcoeff}(p)$  denotes the leading coefficient and  $\text{tcoeff}(p)$  the trailing coefficient of the polynomial  $p$

In the case of  $\sigma = \varepsilon_q$ , there are essentially two compatible valuations <sup>3</sup> on  $\mathbb{F}[x]$  of our interest

$$v_{\text{deg}} : f \mapsto -\deg(f) \quad \text{and} \quad v_{\text{ldeg}} : f \mapsto \text{ldeg}(f)$$

which naturally extend to  $\mathbb{F}(x)$  by  $v(\frac{f}{g}) := v(f) - v(g)$ . For  $\sigma = \varepsilon$  we only have one important compatible valuation, namely  $v_{\text{deg}}$ . In this case the valuation  $v_{\text{ldeg}}$  is not compatible, because for  $n \in \mathbb{N}$  we have  $\text{ldeg}(x^n) = n$ , but  $\text{ldeg}(\varepsilon(x^n)) = \text{ldeg}((x+1)^n) = 0$ . With these valuations we can easily determine the local types of an  $m$ -fold hypergeometric term in the exceptional points.

**Lemma 4.** For an  $m$ -fold hypergeometric term  $u$  with  $m$ -fold certificate  $r = \frac{f}{g}$  and  $\text{ltype}_\infty(u) = (c_\infty, v_\infty)$  the following relations hold

$$v_\infty = v_{\text{deg}}(r) \quad \text{and} \quad c_\infty = \frac{\text{lcoeff}(f)}{\text{lcoeff}(g)}.$$

In the  $q$ -case we have additionally

$$v_0 = v_{\text{ldeg}}(r) \quad \text{and} \quad c_0 = \frac{\text{tcoeff}(f)}{\text{tcoeff}(g)},$$

where  $\text{ltype}_0(u) = (c_0, v_0)$ .

## 2.2. The $\sigma$ -Newton Polygon and its characteristic polynomials

In this section, we present the  $\sigma$ -Newton polygon and its characteristic polynomials. The  $\sigma$ -Newton polygon will give us some valuable a priori information about the structure of the  $m$ -fold certificates of  $m$ -fold hypergeometric solutions of linear recurrence equations.

**Definition 5.** Let  $L = \sum_{i=0}^n a_i \sigma^i \in \mathbb{F}[x][\sigma]$ . Then

$$N_v(L) := \text{convex hull of } \bigcup_{i=0}^n \{(i, y) \mid y \geq v(a_i)\} \subseteq \mathbb{R}^2$$

is the  $\sigma$ -Newton polygon of  $L$  w.r.t. the valuation  $v$ .

In the following we consider the edges with finite slope of the  $\sigma$ -Newton polygon in detail. We denote the sections of the  $\sigma$ -Newton polygon with slope  $w \in \mathbb{Q}$  as the *edge*  $w$ . The *length* of edge  $w$  is the length of the projection of that edge onto the  $x$ -axis.

**Definition 6.** Let  $L = \sum_{i=0}^n \sum_{k=0}^{m_i} \alpha_{i,k} x^k \sigma^i \in \mathbb{F}[x][\sigma]$  and  $v$  a valuation. In the shift case the *characteristic polynomial(s)* of  $N_v(L)$  w.r.t. the edge  $w \in \mathbb{Q}$  with length  $l \in \mathbb{N}$  is (are) given by

$$P_{L,v,w}^m(T) := \sum_{\substack{i,k \text{ where} \\ (im+j, v(\alpha_{im+j,k} x^k)) \\ \text{lies on edge } w}} \alpha_{im+j,k} T^{i-i_0} \in \mathbb{F}[T]$$

and in the  $q$ -case by

$$P_{L,v,w}^m(T) := \sum_{\substack{i,k \text{ where} \\ (im+j, v(\alpha_{im+j,k} x^k)) \\ \text{lies on edge } w}} q^{-v(x) \left( \frac{i(i-1)}{2} m^2 + i j m \right) w} \alpha_{im+j,k} T^{i-i_0} \in \mathbb{F}[T]$$

<sup>3</sup>  $\deg(p)$  denotes the degree and  $\text{ldeg}(p)$  the low degree of the polynomial  $p$

for all  $m \in \mathbb{N}$  such that  $mw \in \mathbb{Z}$  and  $m \mid l$ , where  $i_0 \in \mathbb{N}_0$  is such that  $T \nmid P_{L,v,w}^m(T)$ . In the special case  $w \in \mathbb{Z}$ , we have<sup>4</sup>

$$P_{L,v,w}(T) = \sum_{\substack{i,k \text{ where} \\ (i,v(\alpha_{i,k}x^k)) \\ \text{lies on edge } w}} \alpha_{i,k} T^{i-i_0} \in \mathbb{F}[T] \quad \text{and} \quad P_{L,v,w}(T) = \sum_{\substack{i,k \text{ where} \\ (i,v(\alpha_{i,k}x^k)) \\ \text{lies on edge } w}} q^{-v(x)\frac{i(i-1)}{2}w} \alpha_{i,k} T^{i-i_0} \in \mathbb{F}[T]$$

respectively.

In general, the number  $m$  in Definition 6 is not uniquely determined. For the edge  $w = 1$  of length  $l \in \mathbb{N}$ , we have for example  $d(l)$  characteristic polynomials, where  $d$  is the divisor function.

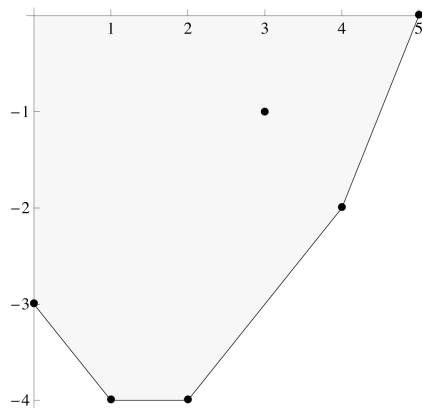


Figure 1. The  $\sigma$ -Newton polygon  $N_{v_{\text{deg}}}(L)$  of Example 5

To illustrate the  $\sigma$ -Newton polygon, we consider the following example.

**Example 5.** Let  $\sigma = \varepsilon_q$ ,  $v = v_{\text{deg}}$  and

$$L = \varepsilon_q^5 + (q^2 x^2 - 1)\varepsilon_q^4 + x\varepsilon_q^3 - (x^4 + x + 1)\varepsilon_q^2 + q^2 x^4 \varepsilon_q - (x^3 + q).$$

Then the vertices of the  $\varepsilon_q$ -Newton polygon  $N_{v_{\text{deg}}}(L)$  are given by

$$\{(0, -3), (1, -4), (2, -4), (4, -2), (5, 0)\}$$

(see Figure 1). The point  $(3, -1)$ , which corresponds to the term  $x\varepsilon_q^3$ , lies in the interior of the  $\varepsilon_q$ -Newton polygon and therefore it is missing in the above list. The characteristic polynomials of  $N_{v_{\text{deg}}}(L)$  are

$$\begin{aligned} P_{L,v_{\text{deg}},-1}(T) &= q^2 T - 1, \\ P_{L,v_{\text{deg}},0}(T) &= -T + q^2, \\ P_{L,v_{\text{deg}},1}(T) &= q^8 T^2 - q, \\ P_{L,v_{\text{deg}},2}(T) &= q^{20} T + q^{14}, \\ P_{L,v_{\text{deg}},1}^2(T) &= q^6 T - 1. \end{aligned}$$

<sup>4</sup> If  $m = 1$ , then we omit the number  $m$  in the notation of the characteristic polynomial.

With the following algorithm we can determine the  $\sigma$ -Newton polygon and its characteristic polynomials w.r.t.  $m = 1$ .

---

**Algorithm 1 ( $\sigma$ -Newton polygon)** Determination of the  $\sigma$ -Newton polygon of a linear recurrence operator with associated characteristic polynomials w.r.t.  $m = 1$

---

**Input** :  $L = \sum_{i=0}^n a_i \sigma^i$  of order  $n$  and a valuation  $v$

**Output** :  $\sigma$ -Newton polygon as list of vertices and characteristic polynomials w.r.t. the edges as list of triples  $(w, l, P_{L,v,w}(T))$  where  $w$  denotes the edge and  $l$  its length

```

1 begin
2   NP ← ∅, charpols ← ∅, i ← 0
3   while i ≠ n do
4     if ai = 0 then next i
5     NP = NP ∪ {(i, v(ai))}
6     slope ← ∞
7     for j ← i + 1 to n do
8       if aj = 0 then next j
9       s ←  $\frac{v(a_j) - v(a_i)}{j - i}$ 
10      if s < slope then
11        slope ← s
12        charpol ← ai,|v(ai)|
13      end
14      if s = slope then
15        charpol ← charpol + aj,|v(aj)| · Tj-i
16        k ← j
17      end
18    end
19    charpols = charpols ∪ {(slope, deg(charpol), charpol)}
20    i ← k
21  end
22  NP = NP ∪ {(n, v(an))}
23  return NP, charpols
24 end

```

---

For the  $q$ -case lines 12 and 15 of Algorithm 1 have to be adapted by multiplying  $q^{-v(x) \frac{k(k-1)}{2}} s$  to the coefficient  $a_{k,|v(a_k)|}$ .

In order to determine all characteristic polynomials we consider one edge  $w$  of length  $l$  out of the computed list  $charpols$  in detail. Then, for all divisors  $m \neq 1$  of  $l$  with  $mw \in \mathbb{Z}$ , we determine  $P_{L,v,w}^m(T)$  as in Definition 6. This will be done for every edge  $w$  in the list to compute all characteristic polynomials.

The main theorem of this paper connects the  $\sigma$ -Newton polygon of a recurrence operator with its  $m$ -fold hypergeometric solutions.

**Theorem 7.** *Let  $u$  be an  $m$ -fold hypergeometric solution of  $L(u) = 0$  with  $m$ -fold certificate  $r = c \frac{s}{t}$ , where  $c \in \mathbb{F}^\times$ ,  $s, t \in \mathbb{F}[x]$  monic<sup>5</sup> w.r.t. the given valuation  $v \in \{v_{\text{deg}}, v_{1\text{deg}}\}$  and  $\gcd(s, t) = 1$ . Then the  $\sigma$ -Newton polygon  $N_v(L)$  has an edge with slope  $w = -\frac{v(r)}{m}$  and length  $km$  with  $k \in \mathbb{N}$  and  $P_{L,v,w}^m(c) = 0$ .*

---

<sup>5</sup> For  $v = v_{\text{deg}}$  we consider  $\text{lcoeff}(s) = \text{lcoeff}(t) = 1$  and for  $v = v_{1\text{deg}}$  we consider  $\text{tcoeff}(s) = \text{tcoeff}(t) = 1$ .

**Proof.** The existence of the edge with the corresponding slope can easily (but lengthly) be adopted from the ‘classical’ difference or differential case (e.g. Robba (1980), Duval (1983)).

Based on the fact that  $\sigma^m(u) = ur$  holds, we obtain  $\sigma^{im+j}(u) = \sigma^j(u) \prod_{k=0}^{i-1} \sigma^{km+j}(r)$  for  $i \in \mathbb{N}$  and  $j = 0, \dots, m-1$  by induction. It follows that

$$L(u) = \sum_{j=0}^{m-1} \left( \sum_{i=0}^{d_j} a_{im+j} \prod_{k=0}^{i-1} \sigma^{km+j}(r) \right) \sigma^j(u) = \sum_{j=0}^{m-1} \left( \sum_{i=0}^{d_j} a_{im+j} \prod_{k=0}^{i-1} \sigma^{km+j} \left( \frac{s}{t} \right) \right) \sigma^j(u) = 0$$

where  $d_j$  is such that  $a_{d_j m+j}$  is the ‘highest’ nonvanishing polynomial of all  $a_{im+j}$ ’s with  $i \in \mathbb{N}_{\geq 0}$ , leading to

$$\sum_{i=0}^{d_j} a_{im+j} c^i \prod_{k=0}^{i-1} \sigma^{km+j}(s) \prod_{k=i}^{d_j-1} \sigma^{km+j}(t) = 0$$

for  $j = 0, \dots, m-1$ . Note that this conclusion is correct, since we are dealing with primitive  $m$ -fold hypergeometric solutions. From the definition

$$P_j(T) := \sum_{i=0}^{d_j} \underbrace{\left( a_{im+j} \prod_{k=0}^{i-1} \sigma^{km+j}(s) \prod_{k=i}^{d_j-1} \sigma^{km+j}(t) \right)}_{b_{j,i}} T^i$$

it follows that  $P_j(c) = 0$  for  $j = 0, \dots, m-1$ . We consider  $P_j(T)$  as polynomial in  $x$  and select those polynomials  $b_{j,i}$ , whose valuations are minimal, because their leading (or trailing resp.) coefficient contributes to the highest (or lowest resp.) coefficient of  $P_j(T)$  w.r.t.  $x$ . Every coefficient is a polynomial in  $T$  vanishing at  $c$ . The valuation of  $b_{j,i}$  is given by

$$\begin{aligned} v(b_{j,i}) &= v \left( a_{im+j} \prod_{k=0}^{i-1} \sigma^{km+j}(s) \prod_{k=i}^{d_j-1} \sigma^{km+j}(t) \right) \\ &= v(a_{im+j}) + \sum_{k=0}^{i-1} v(s) + \sum_{k=i}^{d_j-1} v(t) \\ &= v(a_{im+j}) + i v(s) + (d_j - i) v(t) \\ &= v(a_{im+j}) - i m w + d_j v(t), \end{aligned}$$

where the second equation follows from the fact that we only consider valuations which are compatible with  $\sigma$ . The value  $v(b_{j,i})$  becomes minimal, if the corresponding point  $(im+j, v(a_{im+j}))$  lies on the edge  $w$  of the  $\sigma$ -Newton polygon  $N_v(L)$ , because for all  $k < \underline{i}$

$$\frac{v(a_{im+j}) - v(a_{km+j})}{(\underline{i} - k)m} < w \implies v(a_{\underline{i}}) - \underline{i} m w < v(a_k) - k m w$$

and for all  $k > \bar{i}$

$$\frac{v(a_{\bar{i}m+j}) - v(a_{km+j})}{(\bar{i} - k)m} > w \implies v(a_{\bar{i}}) - \bar{i} m w < v(a_k) - k m w,$$



where  $\underline{i}$  and  $\bar{i}$  denote the smallest and biggest index for which  $(im + j, v(a_{im+j}))$  lies on the edge  $w$ . Furthermore, for  $k$  with  $\underline{i} < k < \bar{i}$ , where  $(km + j, v(a_{km+j}))$  does not lie on the edge  $w$ , the above estimates are also valid. Hence we obtain for the shift case

$$\sum_{\substack{i,k \text{ where} \\ (im+j, v(\alpha_{im+j,k} x^k)) \\ \text{lies on edge } w}} \alpha_{im+j,k} c^i = P_{L,v,w}^m(c) c^{i_0} = 0$$

with  $w = -\frac{v(r)}{m}$ ,  $\alpha_{im+j,k}$  and  $i_0$  from Definition 6. For the  $q$ -case the characteristic polynomials are more complicated since a monic polynomial does not remain monic under the  $q$ -shift. We consider the highest coefficient of  $b_{j,i}$  in detail (case  $v = v_{\text{deg}}$ ) and obtain

$$\begin{aligned} \text{lcoeff}(b_{j,i}) &= \text{lcoeff}(a_{im+j}) \prod_{k=0}^{i-1} q^{(km+j) \deg(s)} \prod_{k=i}^{d_j-1} q^{(km+j) \deg(t)} \\ &= \text{lcoeff}(a_{im+j}) q^{ji \deg(s)} q^{\frac{i(i-1)}{2} m \deg(s)} q^{j(d_j-i) \deg(t)} q^{\left(\frac{d_j(d_j-1)}{2} - \frac{i(i-1)}{2}\right) m \deg(t)} \\ &= \text{lcoeff}(a_{im+j}) q^{ji(\deg(s)-\deg(t))} q^{\frac{i(i-1)}{2} m(\deg(s)-\deg(t))} q^{jd_j \deg(t)} q^{\frac{d_j(d_j-1)}{2} m \deg(t)} \\ &= \text{lcoeff}(a_{im+j}) q^{jimw} q^{\frac{i(i-1)}{2} m^2 w} q^{jd_j \deg(t)} q^{\frac{d_j(d_j-1)}{2} m \deg(t)} \end{aligned}$$

with  $w = \frac{\deg(s)-\deg(t)}{m} = -\frac{v_{\text{deg}}(r)}{m}$ . The factor  $q^{jd_j \deg(t)} q^{\frac{d_j(d_j-1)}{2} m \deg(t)}$ , which occurs in every  $b_{j,i}$ , is independent of  $i$  and therefore can be cancelled. In the case  $v = v_{\text{ldeg}}$  we analogously consider the trailing coefficient of  $b_{j,i}$  and obtain

$$\text{tcoeff}(b_{j,i}) = \text{tcoeff}(a_{im+j}) q^{-jimw} q^{-\frac{i(i-1)}{2} m^2 w} q^{jd_j \text{ldeg}(t)} q^{\frac{d_j(d_j-1)}{2} m \text{ldeg}(t)}$$

with  $w = -\frac{(\text{ldeg}(s)-\text{ldeg}(t))}{m} = -\frac{v_{\text{ldeg}}(r)}{m}$ . Again, we can neglect the occurring constant factor w.r.t.  $i$ . Summarizing, we obtain

$$\sum_{\substack{i,k \text{ where} \\ (im+j, v(\alpha_{im+j,k} x^k)) \\ \text{lies on edge } w}} q^{-v(x) \left(\frac{i(i-1)}{2} m^2 + i j m\right) w} \alpha_{im+j,k} c^i = P_{L,v,w}^m(c) c^{i_0} = 0$$

with  $w = -\frac{v(r)}{m}$ ,  $\alpha_{im+j,k}$  and  $i_0$  from Definition 6.  $\square$

If we look for all  $m$ -fold hypergeometric solutions of a recurrence operator  $L \in \mathbb{F}[x][\sigma]$  with  $m \neq 1$ , then we determine all characteristic polynomials and read off all candidates for the number  $m$ . For instance, if there is no characteristic polynomial  $P_{L,v,w}^m(T)$ , then  $L$  has no  $m$ -fold hypergeometric solutions by Theorem 7. Hence, the operator considered in Example 5 could only have  $m$ -fold  $q$ -hypergeometric solutions for  $m \in \{1, 2\}$ . In fact, in this example, there are no  $m$ -fold  $q$ -hypergeometric solutions at all. In the  $q$ -case we can actually determine two Newton polygons and their characteristic polynomials and for each of them we get a set of candidates for  $m$ . By intersecting both sets we get a finite set of possible  $m$ 's which could correspond to an  $m$ -fold  $q$ -hypergeometric solution. Once again, in Example 5, we have  $\{1, 2\}$  for  $v = v_{\text{deg}}$  and  $\{1, 5\}$  for  $v = v_{\text{ldeg}}$ , hence we can even state (in almost no time), that there are no  $m$ -fold  $q$ -hypergeometric solutions for  $m \neq 1$ . Thus, the candidate set for  $m$  should be ‘computed’ before one tries to compute  $m$ -fold hypergeometric solutions for a specific  $m$ .

### 2.3. Computing Candidates for Local Types

By the previous theorem, the  $\sigma$ -Newton polygon gives us relevant information about the  $m$ -fold certificates of possible  $m$ -fold hypergeometric solutions, namely the local types in the exceptional points. Therefore we define

$$T_0^m(L) := \{(c_0, v_0) \mid N_{v_{\text{ldeg}}}(L) \text{ has edge } -\frac{v_0}{m} \text{ of length } km \text{ and} \\ P_{L, v_{\text{ldeg}}, -\frac{v_0}{m}}^m(c_0) = 0, \text{ where } k \in \mathbb{N} \text{ and } v_0 \in \mathbb{Z}\}$$

and additionally for the  $q$ -case

$$T_\infty^m(L) := \{(c_\infty, v_\infty) \mid N_{v_{\text{deg}}}(L) \text{ has edge } -\frac{v_\infty}{m} \text{ of length } km \text{ and} \\ P_{L, v_{\text{deg}}, -\frac{v_\infty}{m}}^m(c_\infty) = 0, \text{ where } k \in \mathbb{N} \text{ and } v_\infty \in \mathbb{Z}\}.$$

**Corollary 6.** If  $u$  is an  $m$ -fold hypergeometric solution of  $L(u) = 0$  then  $\text{ltype}_\infty(u) \in T_\infty^m(L)$  and in the  $q$ -case  $\text{ltype}_0(u) \in T_0^m(L)$ .

**Proof.** This follows directly from Theorem 7 and the fact that the valuation of a nonzero rational function is always an integer.  $\square$

**Example 7.** We continue Example 5 and get the following candidates for the local type of a potential  $q$ -hypergeometric solution of  $L$  at  $\infty$

$$T_\infty(L) = \left\{ (q^{-2}, 1), (q^2, 0), (q^{-\frac{7}{2}}, -1), (-q^{-\frac{7}{2}}, -1), (-q^{-6}, -2) \right\}.$$

If we consider 2-fold  $q$ -hypergeometric solutions of  $L$ , we obtain

$$T_\infty^2(L) = \{(q^{-6}, -2)\}.$$

In order to determine bounds for the local type of an  $m$ -hypergeometric solution in a ‘finite point’, we consider Petkovšek’s normal form.

**Theorem 8.** Let  $L = \sum_{i=0}^n a_i \sigma^i \in \mathbb{F}[x][\sigma]$  and

$$\text{cert}_m^\sigma(u) = z \frac{\sigma^m(f) g}{f h} \tag{3}$$

be the  $m$ -fold certificate of an  $m$ -fold hypergeometric solution  $u$  in Petkovšek normal form, with  $z \in \mathbb{F}^\times$  and  $f, g, h \in \mathbb{F}[x]$  monic with

$$\begin{aligned} \gcd(g, (\sigma^m)^k(h)) &= 1 \text{ for all } k \in \mathbb{N}, \\ \gcd(g, f) &= 1, \\ \gcd(h, \sigma^m(f)) &= 1, \\ f(0) &\neq 0 \text{ (only in the } q\text{-case),} \end{aligned}$$

similar to Lemma 1 of Petkovšek and Salvy (1993) for the shift case. Then for the shift case we have

$$(a) \text{ ltype}_\infty(u) = (c_\infty, v_\infty) = (z, \deg(h) - \deg(g))$$

and for the  $q$ -case we have

$$(b) \text{ltype}_\infty(u) = (c_\infty, v_\infty) = (zq^{\deg(f)}, \deg(h) - \deg(g)),$$

$$(c) \text{ltype}_0(u) = (c_0, v_0) = \left( z \frac{\text{tcoeff}(g)}{\text{tcoeff}(h)}, \text{ldeg}(g) - \text{ldeg}(h) \right),$$

$$(d) \frac{\text{tcoeff}(g)}{\text{tcoeff}(h)} c_\infty \sim_m^{\varepsilon_q} c_0.$$

**Proof.** For (a), (b) and (c) one uses the formulas from Lemma 4 taking into account that all occurring polynomials are monic. Relation (d) follows from (b) and (c).  $\square$

The core idea of the classical  $m$ -Petkovšek algorithm by Petkovšek and Salvy (1993), which we will modify in the next section, is, that if we use the normal form (3), then one can show that the relations

$$g \mid a_0 \quad \text{and} \quad h \mid \sigma^{-(n-m)}(a_n) \tag{4}$$

hold (see Petkovšek and Salvy (1993)). The local type of the polynomial part of the  $m$ -fold hypergeometric solution is in every ‘finite point’ 0, because if we consider the corresponding factor  $\frac{\sigma^m f}{f}$  of the  $m$ -fold certificate, then every root of the denominator occurs shifted in the numerator, too. Because of (4), we define for  $\alpha \in \overline{\mathbb{F}}$

$$T_{[\alpha]_m^\sigma}(L) = \left\{ -\sum_{\beta \in [\alpha]_m^\sigma} \text{mult}(\sigma^{-(n-m)}(a_n), \beta), \dots, \sum_{\beta \in [\alpha]_m^\sigma} \text{mult}(a_0, \beta) \right\}$$

where  $\text{mult}(a, \beta)$  is the multiplicity of the root  $\beta$  of the polynomial  $a$ .

**Corollary 8.** If  $u$  is an  $m$ -fold hypergeometric solution of  $L(u) = 0$  then for every  $\alpha \in \overline{\mathbb{F}}$  we have  $\text{ltype}_{[\alpha]_m^\sigma}(u) \in T_{[\alpha]_m^\sigma}(L)$ .

Hence to compute these bounds, one first factors the leading and trailing coefficients and collects the factors up to  $m$ -shift equivalence. Then one adds up the multiplicities in each class.

### 3. Right Factors of $m$ -fold Hypergeometric Type

We assume that the operators for which right factors of  $m$ -fold hypergeometric type are sought do not have polynomial or rational solutions. Otherwise, these should be computed beforehand with known efficient algorithms by Abramov et al. (1995), Abramov (1995), van Hoeij (1998b), Böing and Koepf (1999).

#### 3.1. $m$ -fold Van Hoeij Approach

From the above sections we get a finite set of possible types of  $m$ -fold hypergeometric solutions of a recurrence equation. Each of these candidates amounts to the  $m$ -fold certificate of such a solution determined up to  $m$ -shifts for each irreducible factor of numerator and denominator. So we can easily compute a set of possible correct-up-to- $m$ -shift  $m$ -fold certificates. In order to reconstruct the ‘real’  $m$ -fold certificate of an  $m$ -fold hypergeometric solution, we need the following recurrence operator.

**Definition 9.** Let  $L_1$  and  $L_2$  be recurrence operators. The *symmetric product*  $L_1 \mathbb{S} L_2$  is defined as the unique monic recurrence operator of minimal order such that for all  $m$ -fold hypergeometric terms  $u, w$  holds:

$$L_1(u) = 0 \wedge L_2(w) = 0 \implies (L_1 \mathbb{S} L_2)(u \cdot w) = 0.$$

Note that  $L_1 \mathbb{S} L_2$  can easily be computed from  $L_1$  and  $L_2$  by linear algebra (see e.g. Cluzeau and van Hoeij (2005)). From Theorem 3 and Definition 2 we know that once we have a ‘candidate’  $m$ -fold certificate  $\tilde{r}$  corresponding to the  $m$ -fold hypergeometric term  $\tilde{u}$ , and compute  $\tilde{L} := L \mathbb{S} (\sigma^m - \frac{1}{\tilde{r}})$ , then  $\tilde{L}$  must have a rational solution  $s = \frac{u}{\tilde{u}}$  if  $\tilde{u}$  was of the same type as an actual solution  $u$  with  $m$ -fold certificate  $r$ . Then, this  $r$  can be reconstructed from  $s$  and  $\tilde{r}$  by

$$\frac{\sigma^m(s)}{s} \cdot \tilde{r} = \frac{\sigma^m(u) \tilde{u}}{u \sigma^m(\tilde{u})} \cdot \frac{\sigma^m(\tilde{u})}{\tilde{u}} = \frac{\sigma^m(u)}{u} = r.$$

We obtain the following  $m$ -version of the van Hoeij-type algorithm.

---

**Algorithm 2 ( $m$ -fold van Hoeij)** Determination of all right factors of  $m$ -fold hypergeometric type of a linear recurrence operator

---

**Input** :  $m \in \mathbb{N}$  and a linear recurrence operator  $L = \sum_{i=0}^n a_i \varepsilon^i$

**Output** : all right factors of  $m$ -fold hypergeometric type of  $L$

```

1 begin
2    $C \leftarrow \emptyset$ 
3   Compute  $T_\infty^m(L)$  and  $T_{[\alpha]_m^\varepsilon}(L)$  for all  $\alpha \in \overline{\mathbb{F}}$ 
4    $\tilde{C} \leftarrow \left\{ \tilde{c}_\infty \prod_{[\alpha]_m^\varepsilon \in \overline{\mathbb{F}}/\sim_m^\varepsilon} (x - \alpha)^{\tilde{v}_\alpha} \mid (\tilde{c}_\infty, *) \in T_\infty^m(L), \tilde{v}_\alpha \in T_{[\alpha]_m^\varepsilon}(L) \right\}$ 
5   for all  $\tilde{r}(x) \in \tilde{C}$  do
6     if the Fuchsian relation (1) is not satisfied for  $\tilde{r}(x)$  then
7       next
8     end
9      $\tilde{L} \leftarrow L \mathbb{S} (\varepsilon^m - \frac{1}{\tilde{r}})$ 
10     $S \leftarrow$  rational solutions of  $\tilde{L}$ 
11     $C \leftarrow C \cup \left\{ \varepsilon^m - \frac{\varepsilon^m(s)}{s} \cdot \tilde{r} \mid s \in S \right\}$ 
12  end
13  return  $C$ 
14 end
```

---

Note that in line 3 of Algorithms 2 and 3 the sets  $T_{[\alpha]_m^\varepsilon}(L)$  are computed only for those  $\alpha \in \overline{\mathbb{F}}$  that are roots of  $a_0 \cdot \sigma^{-(n-m)}(a_n)$ . In line 6 we use the Fuchsian relations to get rid of  $m$ -fold certificates that certainly do not correspond to proper factors, which reduces the number of possible  $m$ -fold certificates of solutions significantly.

---

**Algorithm 3** (*m-fold q-van Hoesij*) Determination of all right factors of  $m$ -fold  $q$ -hypergeometric type of a linear  $q$ -recurrence operator

---

**Input** :  $m \in \mathbb{N}$  and a linear  $q$ -recurrence operator  $L = \sum_{i=0}^n a_i \varepsilon_q^i$

**Output** : all right factors of  $m$ -fold  $q$ -hypergeometric type of  $L$

---

```

1 begin
2    $C \leftarrow \emptyset$ 
3   Compute  $T_\infty^m(L), T_0^m(L)$  and  $T_{[\alpha]_m^{\varepsilon_q}}(L)$  for all  $\alpha \in \overline{\mathbb{F}}^\times$ 
4    $\tilde{C} \leftarrow \left\{ \tilde{c}_\infty x^{\tilde{v}_0} \prod_{[\alpha]_m^{\varepsilon_q} \in \overline{\mathbb{F}}^\times / \sim_m^{\varepsilon_q}} (x - \alpha)^{\tilde{v}_\alpha} \mid (\tilde{c}_\infty, *) \in T_\infty^m(L), (*, \tilde{v}_0) \in T_0^m(L), \tilde{v}_\alpha \in T_{[\alpha]_m^{\varepsilon_q}}(L) \right\}$ 
5   for all  $\tilde{r}(x) \in \tilde{C}$  do
6     if one of the  $q$ -Fuchsian relations (2) is not satisfied for  $\tilde{r}(x)$  then
7       next
8     end
9      $\tilde{L} \leftarrow L \otimes (\varepsilon_q^m - \frac{1}{\tilde{r}})$ 
10     $S \leftarrow$  rational solutions of  $\tilde{L}$ 
11     $C \leftarrow C \cup \left\{ \varepsilon_q^m - \frac{\varepsilon_q^m(s)}{s} \cdot \tilde{r} \mid s \in S \right\}$ 
12  end
13  return  $C$ 
14 end
```

---

We point out that our version of van Hoesij's algorithm does not compute solutions over algebraic extensions as described in Cluzeau and van Hoesij (2005). Therefore, our van Hoesij-type algorithm consists of only one part of the algorithm proposed in Cluzeau and van Hoesij (2005) which for most purposes is sufficient. We consider another method for computing right factors of  $m$ -fold hypergeometric type in the next section.

### 3.2. $m$ -fold Petkovšek Revisited

In this section, we give a version of the  $m$ -fold Petkovšek algorithm (Petkovšek and Salvy (1993)) that is modified in two ways using the information from the  $\sigma$ -Newton polygon. First, we are able to simplify the computation of the 'leading coefficient' of the certificate. Second, we can significantly reduce the number of candidates that have to be taken into consideration, which leads to an efficient algorithm to compute  $m$ -fold hypergeometric solutions. Once again, we assume that the  $m$ -fold certificate of an  $m$ -fold hypergeometric solution is in Petkovšek normal form. From (4) we get a finite number of possible choices for  $g$  and  $h$ . To compute candidates for  $z$ , we use the candidates for the local types in the exceptional points<sup>6</sup>. If  $z$ ,  $g$ , and  $h$  contribute to a proper factor of an  $m$ -fold certificate of an  $m$ -fold hypergeometric solution,  $f$  is a polynomial solution of

$$\sum_{i=0}^{d_j} \left( z^i a_{im+j} \prod_{k=0}^{i-1} \sigma^{km+j}(g) \prod_{k=i}^{d_j-1} \sigma^{km+j}(h) \right) \sigma^i(f) = 0. \quad (5)$$

for all  $j = 0, \dots, m-1$ . This can be deduced as in the proof of Theorem 7 using the normal form (3) for  $r$  (instead of  $c \frac{\varepsilon}{t}$ ).

The modified versions of the  $m$ -fold Petkovšek algorithm which are described in Algorithms 4 and 5 filter out all polynomials  $g$  and  $h$  according to Corollary 6 and Theorem

---

<sup>6</sup> Note that in the classical  $m$ -Petkovšek algorithm  $z$  is computed in a different way.

---

**Algorithm 4 (modified  $m$ -fold Petkovšek)** Determination of all right factors of  $m$ -fold hypergeometric type of a linear recurrence operator

---

**Input** :  $m \in \mathbb{N}$  and a linear recurrence operator  $L = \sum_{i=0}^n a_i \varepsilon^i$

**Output** : all right factors of  $m$ -fold hypergeometric type of  $L$

```

1 begin
2    $C \leftarrow \emptyset$ 
3   for every monic factor  $g$  of  $a_0$  and  $h$  of  $\varepsilon^{-(n-m)}a_n$  do
4      $v_\infty \leftarrow v_{\deg}(g) - v_{\deg}(h)$ 
5     for all  $(c_\infty, v_\infty) \in T_\infty^m(L)$  do
6        $z \leftarrow c_\infty$ 
7       for  $j = 0, \dots, m-1$  do
8          $L_j \leftarrow \sum_{i=0}^{d_j} \left( z^i a_{im+j} \prod_{k=0}^{i-1} \varepsilon^{km+j}(g) \prod_{k=i}^{d_j-1} \varepsilon^{km+j}(h) \right) \varepsilon^i$ 
9       end
10       $S \leftarrow$  polynomial solutions of  $L_j(f) = 0$  for one specific  $j$ 
11       $C \leftarrow C \cup \left\{ z \frac{\varepsilon^m(f)}{f} \frac{g}{h} \mid f \in S \text{ and } L_j(f) = 0 \text{ for all } j = 0, \dots, m-1 \right\}$ 
12    end
13  end
14  return  $C$ 
15 end
```

---

8, which do not contribute to a part of an  $m$ -fold certificate of an  $m$ -fold hypergeometric solution. Thus, the algorithm is substantially more efficient than the classical  $m$ -fold Petkovšek algorithm (especially also for  $m = 1$ ), because in many cases the considered recurrence operator (5) does not need to be constructed and eventually no polynomial solver needs to be used. If we consider linear recurrence operators which have leading and trailing coefficients with many factors, this improvement is enormous.

Notes on Algorithms 4 and 5:

- (1) Line 3 of Algorithm 4 and lines 3 and 6 of Algorithm 5 are iterations over the cartesian products.
- (2) In line 6 of Algorithm 5 the values  $v_0$  and  $v_\infty$  are already computed and we iterate only over those pairs of  $T_0^m(L)$  and  $T_\infty^m(L)$  with second component  $v_0$  and  $v_\infty$  respectively. In the shift case the same is true w.r.t.  $v_\infty$  in line 5 of Algorithm 4.
- (3) From line 8 of Algorithm 5 we get the possible degrees of the polynomial solutions in line 12 by  $q^{\deg(f)} = \frac{c_\infty}{z}$ .
- (4) In line 10 of Algorithm 4 and line 12 of Algorithm 5 the number  $j \in \{0, \dots, m-1\}$  can be chosen such that  $L_j$  is the first nontrivial operator.
- (5) In line 11 of Algorithm 4 and line 13 of Algorithm 5 we must check if  $f$  is annihilated by all  $L_j$ 's ( $j = 0, \dots, m-1$ ). Alternatively, we can perform a division with remainder of our given operator  $L$  and the operator which corresponds to our candidate  $m$ -fold certificate. If the remainder is zero, the candidate is an  $m$ -fold certificate of an  $m$ -fold hypergeometric solution.

Let  $m = 1$ , then for  $\sigma = \varepsilon$  the van Hoeij algorithm (van Hoeij (1998a), Cluzeau and van Hoeij (2005)) is the most efficient algorithm. In the  $q$ -case the modified  $q$ -Petkovšek algo-

---

**Algorithm 5 (modified  $m$ -fold  $q$ -Petkovšek)** Determination of all right factors of  $m$ -fold  $q$ -hypergeometric type of a linear  $q$ -recurrence operator

---

**Input** :  $m \in \mathbb{N}$  and a linear  $q$ -recurrence operator  $L = \sum_{i=0}^n a_i \varepsilon_q^i$

**Output** : all right factors of  $m$ -fold  $q$ -hypergeometric type of  $L$

```

1 begin
2    $C \leftarrow \emptyset$ 
3   for every monic factor  $g$  of  $a_0$  and  $h$  of  $\varepsilon_q^{-(n-m)} a_n$  do
4      $v_0 \leftarrow v_{\text{ldeg}}(g) - v_{\text{ldeg}}(h)$ ;  $v_\infty \leftarrow v_{\text{deg}}(g) - v_{\text{deg}}(h)$ 
5      $c \leftarrow \frac{\text{tcoeff}(g)}{\text{tcoeff}(h)}$ 
6     for all  $(c_0, v_0) \in T_0^m(L)$  and  $(c_\infty, v_\infty) \in T_\infty^m(L)$  do
7       if  $c \cdot c_\infty \sim_m^{\varepsilon_q} c_0$  then
8          $z \leftarrow \frac{c_0}{c}$ 
9         for  $j = 0, \dots, m-1$  do
10           $L_j \leftarrow \sum_{i=0}^{d_j} \left( z^i a_{im+j} \prod_{k=0}^{i-1} \varepsilon_q^{km+j}(g) \prod_{k=i}^{d_j-1} \varepsilon_q^{km+j}(h) \right) \varepsilon_q^i$ 
11          end
12           $S \leftarrow$  polynomial solutions of  $L_j(f) = 0$  for one specific  $j = 0, \dots, m-1$ 
13           $C \leftarrow C \cup \left\{ z \frac{\varepsilon_q^m(f)}{f} \frac{g}{h} \mid f \in S \text{ and } L_j(f) = 0 \text{ for all } j = 0, \dots, m-1 \right\}$ 
14        end
15      end
16    end
17  return  $C$ 
18 end
```

---

rithm (Horn (2008)) is in many cases better than a  $q$ -version of the van Hoeij algorithm (see next section). For  $m > 1$  the modified Petkovšek algorithm should be preferred.

### 3.3. The Special Case $m = 1$ and $\sigma = \varepsilon_q$ ( $q$ -Hypergeometric Solutions)

Now, let  $m = 1$  and  $\sigma = \varepsilon_q$ . We consider operators of order three parametrized by  $j \in \mathbb{N}$ , where the leading coefficient has degree  $\lfloor \frac{j+1}{2} \rfloor$  and the trailing coefficient has degree  $\lfloor \frac{j}{2} \rfloor + 1$ , and both factor into linear factors. To compare the algorithms, we construct two different series of operators, first

$$\left( \left( \prod_{i=1}^{\lfloor \frac{j+1}{2} \rfloor} (x + iq^i) \right) \varepsilon^2 + \prod_{i=1}^{\lfloor \frac{j}{2} \rfloor} (x - iq^i) \right) \cdot (\varepsilon - x), \quad (6)$$

where none of the occurring linear factors is  $q$ -shift equivalent to another. Second

$$\left( \left( \prod_{i=1}^{\lfloor \frac{j+1}{2} \rfloor} (x - q^i) \right) \varepsilon^2 + \prod_{i=1}^{\lfloor \frac{j}{2} \rfloor} (x - iq^i) \right) \cdot (\varepsilon - x), \quad (7)$$

where all linear factors of the leading coefficient are  $q$ -shift equivalent whereas the classes of the trailing coefficient are distinct. Obviously, both operators have  $q$ -hypergeometric solutions with  $q$ -certificate  $x$ .

The timings are in seconds and were recorded on a 3 GHz Intel Xeon with 16GB of RAM using Maple 12. A dash denotes a case in which no result was computed after three hours.

$j$	Operator (6)						Operator (7)		
	Time			Candidates			Time		
	Pet	Hoeij	modPet	Pet	Hoeij	modPet	Pet	Hoeij	modPet
1	0.05	0.24	0.05	2	3	2	0.06	0.23	0.06
2	0.08	0.07	0.04	4	1	1	0.07	0.07	0.05
3	0.11	0.19	0.07	8	2	2	0.06	0.25	0.05
4	0.20	0.10	0.06	16	1	1	0.07	0.07	0.04
5	0.32	3.84	0.10	32	10	6	0.10	0.09	0.07
6	0.70	0.40	0.09	64	2	2	0.21	0.10	0.06
7	1.11	2.04	0.18	128	8	8	0.31	0.12	0.08
8	2.83	1.33	0.15	256	4	4	0.65	0.14	0.07
9	6.70	62.31	0.52	512	40	24	1.31	0.18	0.16
10	14.72	5.04	0.52	1024	8	8	2.84	0.25	0.15
11	21.05	35.60	1.45	2048	32	32	6.24	0.32	0.44
12	104.45	39.00	1.15	4096	20	20	14.72	0.47	0.40
13	309.95	2315.12	5.35	8192	200	120	34.37	0.53	1.61
14	775.04	335.54	4.19	16384	40	40	93.92	0.73	1.17
15	535.45	3092.98	19.48	32768	168	168	306.08	1.14	5.16
16	5102.05	3930.01	14.54	65536	96	96	936.54	1.64	3.80
17	9401.35	-	70.41	131072	960	576	2425.72	2.33	22.22
18	-	-	53.61	262144	216	216	5348.26	4.37	15.27
19	-	-	278.14	524288	944	944	-	5.90	74.40
20	-	-	218.66	1048576	604	604	-	11.26	52.79
21	-	-	1106.30	2097152	6040	3624	-	13.79	296.69
22	-	-	903.62	4194304	1208	1208	-	28.25	224.87
23	-	-	4534.31	8388608	5380	5380	-	35.00	1172.01
24	-	-	3446.78	16777216	3936	3936	-	58.57	808.60
25	-	-	-	-	-	-	-	75.04	4756.82
26	-	-	-	-	-	-	-	117.49	3546.98
27	-	-	-	-	-	-	-	153.91	-
28	-	-	-	-	-	-	-	237.06	-
30	-	-	-	-	-	-	-	482.38	-
32	-	-	-	-	-	-	-	979.83	-
34	-	-	-	-	-	-	-	2174.75	-
36	-	-	-	-	-	-	-	4018.23	-
38	-	-	-	-	-	-	-	7538.78	-
40	-	-	-	-	-	-	-	9750.52	-

**Table 1.** Timings of different *Maple* procedures for the determination of first-order right factors of operators (6) and (7) and the number of candidates w.r.t. operator (6).



The classical  $q$ -Petkovšek algorithm is exponential in  $j$ , in fact the number of candidates is  $2^j$  for (6) and (7). The modified  $q$ -Petkovšek algorithm is still exponential in  $j$ , but has to perform only ‘tests’ in most cases, hence it investigates only few cases in detail. For the operator (6) the number of candidates is also shown in Table 1. The  $q$ -van Hoeff algorithm is exponential in the number of occurring shift equivalence classes but uses similar improvements as the modified  $q$ -Petkovšek algorithm. For (7) both the  $q$ -van Hoeff and the modified Petkovšek algorithm reject all but one candidate for all  $j$ .

With the above knowledge, a hybrid implementation would be feasible and reasonable. The decision for one of the algorithms can be made after the computation of the  $q$ -Newton polygon and the factorization of the leading and trailing coefficients.

### 3.4. Maple Implementation

A *Maple* implementation of the  $m$ -fold  $q$ -van Hoeff and  $q$ -Petkovšek algorithm can be found at <http://www.mathematik.uni-kassel.de/~sprenger/mfoldhypergeom.php>.

In the following, we give an example for the use of the algorithms, where we construct a linear  $q$ -recurrence equation of order 5, which has one 2-fold  $q$ -hypergeometric solution and another 3-fold  $q$ -hypergeometric solution. The output of the following function calls of `qHypergeomSolveRE` are the  $m$ -fold  $q$ -certificates of the  $m$ -fold  $q$ -hypergeometric solutions.

#### Example 9.

```

> m1:=2;
                                     m1 := 2
> cert1:=q*(q-1)*x/(x-1);
                                     cert1 :=  $\frac{q(q-1)x}{x-1}$ 
> RE1:=denom(cert1)*qshift(f(x),[x$m1],q)-numer(cert1)*f(x)=0;
                                     RE1 :=  $(x-1)(Sq_{x,x})(f(x)) - q(q-1)xf(x) = 0$ 
> m2:=3;
                                     m2 := 3
> cert2:=(q-1)*(x-3)^2/(x-q);
                                     cert2 :=  $\frac{(x-3)^2(q-1)}{x-q}$ 
> RE2:=denom(cert2)*qshift(f(x),[x$m2],q)-numer(cert2)*f(x)=0;
                                     RE2 :=  $(-x+q)(Sq_{x,x,x})(f(x)) + (q-1)(x-3)^2f(x) = 0$ 
> RE:=qNormal(qLCM(RE2,RE1),f(x)):
> qOrder(RE,f(x));
                                     5

```

The coefficients of our  $q$ -recurrence equation RE of order 5 are quite huge, hence we suppressed the output. Now, we determine all  $m$ -fold hypergeometric of solutions of RE.

```

> st:=time();
> qHypergeomSolveRE(RE,f(x),method=qVanHoeff,mhypersol=2);
> time()-st;

```

$$\left[ \frac{q(q-1)x}{x-1} \right]$$

16.377

```
> st:=time():
> qHypergeomSolveRE(RE,f(x),method=qPetkovsek,mhypersol=2);
> time()-st;
```

$$\left[ \frac{q(q-1)x}{x-1} \right]$$

1.997

```
> st:=time():
> qHypergeomSolveRE(RE,f(x),method=modqPetkovsek,mhypersol=2);
> time()-st;
```

$$\left[ \frac{q(q-1)x}{x-1} \right]$$

0.281

```
> st:=time():
> qHypergeomSolveRE(RE,f(x),method=qVanHoeij,mhypersol=3);
> time()-st;
```

$$\left[ -\frac{(x-3)^2(q-1)}{-x+q} \right]$$

20.836

```
> st:=time():
> qHypergeomSolveRE(RE,f(x),method=qPetkovsek,mhypersol=3);
> time()-st;
```

$$\left[ -\frac{(x-3)^2(q-1)}{-x+q} \right]$$

2.317

```
> st:=time():
> qHypergeomSolveRE(RE,f(x),method=modqPetkovsek,mhypersol=3);
> time()-st;
```

$$\left[ -\frac{(x-3)^2(q-1)}{-x+q} \right]$$

1.076

```
> st:=time():
> qHypergeomSolveRE(RE,f(x),mhypersol=0);
> time()-st;
```

$$\left\{ \left[ 2, \left[ \frac{q(q-1)x}{x-1} \right] \right], \left[ 3, \left[ -\frac{(x-3)^2(q-1)}{-x+q} \right] \right] \right\}$$

2.174

In the last line we determine all  $m$ -fold hypergeometric solutions in one function call by setting `mhypersol=0`. In that call, the most efficient algorithm, the modified  $m$ -fold  $q$ -Petkovšek algorithm, is used by default. The  $m$ -fold  $q$ -van Hoeij algorithm is rather

slow in this example, because the symmetric product has complex leading and trailing coefficients.

We consider another example presenting the algorithm in practice.

**Example 10.** We determine an  $m$ -fold  $q$ -hypergeometric series representation with expansion point  $a$  for a function  $F(y)$  with algorithms described in Sprenger and Koepf (2011), i.e.

$$F(y) = \sum_{j=0}^{\infty} c_{mj}(y \ominus a)_q^{mj} + \sum_{j=0}^{\infty} c_{mj+1}(y \ominus a)_q^{mj+1} + \dots + \sum_{j=0}^{\infty} c_{m,j+(m-1)}(y \ominus a)_q^{mj+(m-1)},$$

where  $(y \ominus a)_q^j = (y - a) \cdot (y - aq) \cdots (y - aq^{j-1})$ . In this example we consider the small  $q$ -sine function  $F(y) = \sin_q(y)$ . First, we develop a  $q$ -differential equation for  $\sin_q(y)$ .

```
> qDE:=qHolonomicDE(qsin(y,q),F(y));
```

$$qDE := F(y) + (q - 1)^2 (D_{q,y,y})(F(y)) = 0$$

Then we convert the  $q$ -differential equation into a  $q$ -recurrence equation for the  $q$ -series coefficients  $c_j$  with a pattern-matching algorithm.

```
> RE:=qDEtoRE(qDE,F(y),c(j),base=qpower,expansionpt=a);
```

$$RE := c(j) + qa q^j (1 + q) c(j + 1) + \left( a^2 (q^j)^2 q^4 + (q^j)^2 q^3 - q^j q^2 - q^j q + 1 \right) c(j + 2) \\ + qa q^j (1 + q) (q^j q^3 - 1) (q^j q^2 - 1) c(j + 3) + a^2 (q^j)^2 q^4 (q^j q^4 - 1) (q^j q^3 - 1) c(j + 4) = 0$$

Note that this  $q$ -recurrence equation of  $c_j$  can be transformed via  $q^j = x$  and  $c_j = f(x)$  into a  $q$ -recurrence equation of  $f(x)$  as in the previous example ( $c_{j+n} = f(q^n q^j) = f(q^n x) = \varepsilon_q^n f(x)$ ). Finally, we solve this equation by the  $m$ -fold  $q$ -Petkovšek algorithm. For  $m = 2$  we get

```
> qHypergeomSolveRE(RE,c(j),mhypersol=2);
```

$$\left[ \frac{A_1 (-1)^j}{qpochhammer(q, q, 2j)}, \frac{A_2 (q - 1) (-1)^j}{((q^j)^2 q - 1) qpochhammer(q, q, 2j)} \right]$$

From the  $q$ -Taylor theorem we obtain the two initial values  $c_0 = \sin_q(a)$  and  $c_1 = -\frac{\cos_q(a)}{q-1}$ , leading to

$$\sin_q(y) = \sum_{j=0}^{\infty} (-1)^j \frac{\sin_q(a)}{(q; q)_{2j}} (y \ominus a)_q^{2j} + \sum_{j=0}^{\infty} (-1)^j \frac{\cos_q(a)}{(q; q)_{2j+1}} (y \ominus a)_q^{2j+1}.$$

## Acknowledgment

We would like to thank the anonymous referees for their important remarks which helped to improve our paper substantially.

## References

- Abramov, S., 1995. Rational solutions to linear difference and  $q$ -difference equations with polynomial coefficients. *Programmirovanie* (6), 3–11.
- Abramov, S., Bronstein, M., Petkovšek, M., 1995. On polynomial solutions of linear operator equations. In: *ISSAC 1995 Proceedings*. pp. 290–296.
- Abramov, S., Paule, P., Petkovšek, M., 1998.  $q$ -Hypergeometric solutions of  $q$ -difference equations. *Discrete Math.* 180, 3–22.
- Böing, H., Koepf, W., 1999. Algorithms for  $q$ -hypergeometric summation in computer algebra. *J. Symbolic Comput.* 28, 777–799.
- Cluzeau, T., van Hoeij, M., 2005. Computing hypergeometric solutions of linear recurrence equations. *Appl. Algebra Engrg. Comm. Comput.* 17, 83–115.
- Duval, A., 1983. Lemmes de Hensel et Factorisation Formelle pour les Opérateurs aux Différences. *Funkcialaj Ekvacioj* 26, 349–368.
- Hendriks, P. A., Singer, M. F., 1998. Solving difference equations in finite terms. *J. Symbolic Comput.* 27, 239–259.
- Horn, P., 2008. Faktorisierung in Schief-Polynomringen. Ph.D. thesis, Universität Kassel.
- Petkovšek, M., 1992. Hypergeometric solutions of linear recurrences with polynomial coefficients. *J. Symbolic Comput.* 14 (2-3), 243–264.
- Petkovšek, M., Salvy, B., 1993. Finding all hypergeometric solutions of linear differential equations. In: *ISSAC 1993 Proceedings*. pp. 27–33.
- Robba, P., 1980. Lemmes de Hensel pour les opérateurs différentiels. Application à la réduction formelle des équations différentielles. *L'Enseignement Mathématique* 26 (3-4), 279–311.
- Sprenger, T., Koepf, W., 2011. Algorithmic determination of  $q$ -power series for  $q$ -holonomic functions. *J. Symbolic Comput.* doi:10.1016/j.jsc.2011.12.004.
- van Hoeij, M., 1998a. Finite singularities and hypergeometric solutions of linear recurrence equations. *J. Pure Appl. Algebra* 139, 109–131.
- van Hoeij, M., 1998b. Rational solutions of linear difference equations. In: *ISSAC 1998 Proceedings*. pp. 120–123.