Factorization of C-finite Sequences

Manuel Kauers and Doron Zeilberger

Dedicated to Sergei A. Abramov on the occassion of his 70th birthday

Abstract We discuss how to decide whether a given C-finite sequence can be written nontrivially as a product of two other C-finite sequences.

1 Introduction

It is well known that when $(a_n)_{n=0}^\infty$ and $(b_n)_{n=0}^\infty$ are two sequences that satisfy some linear recurrences with constant coefficients, then the product sequence $(a_n b_n)_{n=0}^\infty$ also satisfies such a recurrence. Sequences satisfying linear recurrences with constant coefficients are called C-finite [17, 8, 19], and the fact just refered to is one of several closure properties that this class of sequences enjoys. In this paper, we will consider the inverse problem: given a C-finite sequence $(c_n)_{n=0}^\infty$, can we write it in a nontrivial way as the product of two other C-finite sequences? This question is of interest in its own right, but it is also useful in some applications in combinatorics. For example, the celebrated solution by Kasteleyn, and Temperley-Fisher, of the dimer problem [3, 7] as well as the even more celebrated Onsager solution of the two-dimensional Ising model [10] can be (re)discovered using an algorithm for factorization of C-finite sequences.

A C-finite sequence is uniquely determined by a recurrence and a choice of sufficiently many initial values. The prototypical example of a C-finite sequence is the Fibonacci sequence $(F_n)_{n=0}^\infty$ defined by

Manuel Kauers

Institute for Algebra, Johannes Kepler Universty, Linz, Austria. e-mail: `manuel.kauers@jku.at`

Doron Zeilberger

Department of Mathematics, Rutgers University, New Brunswick NJ, USA. e-mail: `DoronZeil@gmail.com`

$$F_{n+2} - F_{n+1} - F_n = 0, \qquad F_0 = 0, F_1 = 1.$$

Whether a C-finite sequence $(c_n)_{n=0}^{\infty}$ admits a factorization depends in general on both the recurrence as well as the initial values. For example, the sequence $(3^n + 4^n + 6^n + 8^n)_{n=0}^{\infty}$, which satisfies the recurrence

$$c_{n+4} - 21c_{n+3} + 158c_{n+2} - 504c_{n+1} + 576c_n = 0,$$

can be factored as $3^n + 4^n + 6^n + 8^n = (1 + 2^n)(3^n + 4^n)$, while the sequence $3^n + 4^n + 6^n - 8^n$, which satisfies the same recurrence, cannot be factored.

We shall consider a variant of the factorization problem that does not depend on initial values but only on the recurrence equations. Linear recurrences may be viewed as polynomials $p = p_0 + p_1 x + \cdots + p_d x^d \in k[x]$ acting on sequences $(a_n)_{n=0}^{\infty}$ via

$$p \cdot (a_n)_{n=0}^{\infty} := (p_0 a_n + p_1 a_{n+1} + \cdots + p_d a_{n+d})_{n=0}^{\infty}.$$

For every fixed $p \in k[x]$, denote by $V(p)$ the set of all sequences $(a_n)_{n=0}^{\infty}$ with $p \cdot (a_n)_{n=0}^{\infty} = (0)_{n=0}^{\infty}$, i.e., the solution space of the recurrence equation encoded by p. This is a vector space of dimension $\deg(p)$. For any two operators $p, q \in k[x] \setminus \{0\}$ there exists a unique monic polynomial $r \in k[x]$ such that $V(r)$ is the vector space generated by all sequences $(a_n b_n)_{n=0}^{\infty}$ with $(a_n)_{n=0}^{\infty} \in V(p)$ and $(b_n)_{n=0}^{\infty} \in V(q)$, i.e., $V(r) = V(p) \otimes V(q)$. We write $r = p \otimes q$.

Our problem shall be to decide, for a given monic polynomial $r \in k[x]$, whether there exist $p, q \in k[x]$ such that $r = p \otimes q$. In principle, it is known how to do this. Singer [12] gives a general algorithm for the analogous problem for linear differential operators with rational function coefficients; the problem is further discussed in [6]. Because of their high cost, these algorithms are mainly of theoretical interest. For the special case of differential operators of order 3 or 4 (still with rational function coefficients), van Hoeij [16, 15] combines several observations to an algorithm which handle these cases efficiently. For the recurrence case, Cha [1] gives an algorithm for operators of order 3 with rational function coefficients.

Also the case of recurrence equations with constant coefficients has already been considered. Everest et al. give an algorithm [2] based on a structure theorem of Ritt [11]. This algorithm relies on Ge's algorithm [4], which is efficient in theory but according to our experience rather costly in concrete examples. An alternative algorithm for the case of constant coefficients and arbitrary order was recently sketched by the second author [19]. This description, however, only considers the "generic case". The present paper is a continuation of this work in which we give a complete algorithm which also handles "degenerate" cases. Our algorithm is efficient in the sense that it does not depend on Ge's algorithm or on Gröbner basis computations, but it is inefficient in the sense that it requires a search that may take exponential time in the worst case.

2 Preliminaries

To fix notation, let us recall the basic facts about C-finite sequences. Let k be an algebraically closed field.

Definition 1 *1. A sequence $(a_n)_{n=0}^{\infty}$ is called* C-finite, *if there exist $p_0, \ldots, p_d \in k$ with $p_0 \neq 0 \neq p_d$ such that for all $n \in \mathbb{N}$ we have $p_0 a_n + \cdots + p_d a_{n+d} = 0$.*
2. In this case, the polynomial $p = p_0 + p_1 x + \cdots + p_d x^d$ is called a characteristic polynomial *for $(a_n)_{n=0}^{\infty}$.*
3. For $p \in k[x]$, the set $V(p)$ denotes the set of all C-finite sequences whose characteristic polynomial is p. It is called the solution space *of p.*

Theorem 2. *[13, 8] Let $p = (x - \phi_1)^{e_1} \cdots (x - \phi_m)^{e_m} \in k[x]$ for pairwise distinct $\phi_1, \ldots, \phi_m \in k \setminus \{0\}$. Then $V(p)$ is the k-vector space generated by the sequences*

$$\phi_1^n, \ \ldots, \ n^{e_1 - 1} \phi_1^n,$$
$$\phi_2^n, \ \ldots, \ n^{e_2 - 1} \phi_2^n,$$
$$\ldots\ldots\ldots,$$
$$\phi_m^n, \ \ldots, \ n^{e_m - 1} \phi_m^n.$$

It is an immediate consequence of this theorem that for any two polynomials $p, q \in k[x]$ we have $V(\gcd(p,q)) = V(p) \cap V(q)$ and $V(\text{lcm}(p,q)) = V(p) + V(q)$. The latter says in particular that when $(a_n)_{n=0}^{\infty}$ and $(b_n)_{n=0}^{\infty}$ are C-finite, then so is their sum $(a_n + b_n)_{n=0}^{\infty}$. A similar result holds for the product: write $p = \prod_{i=1}^{m} (x - \phi_i)^{e_i}$ and $q = \prod_{j=1}^{\ell} (x - \psi_j)^{\varepsilon_j}$ and define

$$r := p \otimes q := \text{lcm}_{i=1}^{m} \text{lcm}_{j=1}^{\ell} (x - \phi_i \psi_j)^{e_i + \varepsilon_j - 1}. \tag{1}$$

Then r is a characteristic polynomial for the product sequence $(a_n b_n)_{n=0}^{\infty}$. Note that $\deg(p) + \deg(q) \leq \deg(r) \leq \deg(p) \deg(q)$ for every $p, q \in k[x]$ of degree at least 2. Note also that $p \otimes q = q \otimes p$ for every $p, q \in k[x]$.

Our goal is to recover p and q from a given r. The problem is thus to decide whether the roots of a given polynomial r are precisely the pairwise products of the roots of two other polynomials p and q. Besides the interpretation as a factorization of C-finite sequences, this problem can also be viewed as the factorization of algebraic numbers: given some algebraic number α, specified by its minimal polynomial r, can we write $\alpha = \beta \gamma$ where β, γ are some other algebraic numbers with respective minimal polynomials p and q.

Trivial decompositions are easy to find: For each r we obviously have $r = r \otimes (x - 1)$. Moreover, for every nonzero ϕ we have $(x - \phi) \otimes (x - \phi^{-1}) = (x - 1)$, so we can "decompose" r into $r \otimes (x - \phi)$ and $x - \phi^{-1}$. In order for a decomposition $r = p \otimes q$ to be interesting, we have to require that both p and q have at least degree 2.

Even so, a factorization is in general not unique. Obviously, if $r = p \otimes q$ is a factorization, then for any nonzero ϕ also $r = \big(p \otimes (x - \phi)\big) \otimes \big((x - \phi^{-1}) \otimes q\big)$. Translated to sequences, this ambiguity corresponds to the facts that for every $\phi \neq 0$,

both $(\phi^n)_{n=0}^{\infty}$ and $(\phi^{-n})_{n=0}^{\infty}$ are C-finite, and that a sequence $(a_n)_{n=0}^{\infty}$ is C-finite iff for all $\phi \neq 0$ the sequence $(a_n \phi^n)_{n=0}^{\infty}$ is C-finite. But there is even more non-uniqueness: the polynomial

$$r = (x-2)(x+2)(x-3)(x+3)$$

admits the two distinct factorizations

$$\begin{aligned} r &= (x-1)(x+1) \otimes (x-2)(x+3) \\ &= (x-1)(x+1) \otimes (x-2)(x-3) \end{aligned}$$

which cannot be obtained from one another by introducing factors $(x - \phi)$ and $(x - \phi^{-1})$. Our goal will be to compute a finite list of factorizations from which all others can be obtained by introducing factors $(x - \phi) \otimes (x - \phi^{-1})$.

There is a naive but very expensive algorithm which does this job when r is squarefree: For some choice n, m of degrees, make an ansatz $p = (x - \phi_1) \cdots (x - \phi_n)$ and $q = (x - \psi_1) \cdots (x - \psi_m)$ with variables $\phi_1, \ldots, \phi_n, \psi_1, \ldots, \psi_m$. Equate the coefficients of $r - \prod_{i=1}^{n} \prod_{j=1}^{m} (x - \phi_i \psi_j)$ with respect to x to zero and solve the resulting system of algebraic equations for $\phi_1, \ldots, \phi_n, \psi_1, \ldots, \psi_m$. After trying all possible degree combinations $n \geq m \geq 2$ with $n + m \leq \deg(r) \leq nm$, either a decomposition has been found, or there is none.

3 The Generic Case

Typically, when p and q are square-free polynomials and $\phi_1, \ldots, \phi_n \neq 0$ are the roots of p and $\psi_1, \ldots, \psi_m \neq 0$ are the roots of q, then the products $\phi_i \psi_j$ for $i = 1, \ldots, n$, $j = 1, \ldots, m$ will all be pairwise distinct. In this case, $r = p \otimes q$ will have exactly nm roots, and the factorization problem consists in recovering $\phi_1, \ldots, \phi_n$ and $\psi_1, \ldots, \psi_m$ from the (known) roots $\rho_1, \ldots, \rho_{nm}$ of r.

As observed in [19], a necessary condition for r to admit a factorization into two polynomials of respective degrees n and m is then that there is a bijection $\pi \colon \{1, \ldots, n\} \times \{1, \ldots, m\} \to \{1, \ldots, nm\}$ such that for all j_1, j_2 we have

$$\frac{\rho_{\pi(1,j_1)}}{\rho_{\pi(1,j_2)}} = \frac{\rho_{\pi(2,j_1)}}{\rho_{\pi(2,j_2)}} = \cdots = \frac{\rho_{\pi(n,j_1)}}{\rho_{\pi(n,j_2)}}$$

and for all i_1, i_2 we have

$$\frac{\rho_{\pi(i_1,1)}}{\rho_{\pi(i_2,1)}} = \frac{\rho_{\pi(i_1,2)}}{\rho_{\pi(i_2,2)}} = \cdots = \frac{\rho_{\pi(i_1,m)}}{\rho_{\pi(i_2,m)}}.$$

The explanation is simply that when a factorization exists, then the roots ρ_ℓ of r are precisely the products $\phi_i \psi_j$, and if we define π so that it maps each pair (i, j) to the corresponding root index ℓ, then the quotients

$$\frac{\rho_{\pi(i,j_1)}}{\rho_{\pi(i,j_2)}} = \frac{\phi_i \psi_{j_1}}{\phi_i \psi_{j_2}} = \frac{\psi_{j_1}}{\psi_{j_2}}$$

do not depend on i and the quotients

$$\frac{\rho_{\pi(i_1,j)}}{\rho_{\pi(i_2,j)}} = \frac{\phi_{i_1} \psi_j}{\phi_{i_2} \psi_j} = \frac{\phi_{i_1}}{\phi_{i_2}}$$

do not depend on j.

In fact, the existence of such a bijection π is also sufficient for the existence of a factorization: choose $\phi_1 \neq 0$ arbitrarily and set $\psi_1 := \rho_{\pi(1,1)}/\phi_1$ and

$$\phi_i := \phi_1 \frac{\rho_{\pi(i,1)}}{\rho_{\pi(1,1)}} \qquad (i = 2,\dots,n)$$

and

$$\psi_j := \psi_1 \frac{\rho_{\pi(1,j)}}{\rho_{\pi(1,1)}} \qquad (j = 2,\dots,m).$$

Then we have $\rho_{\pi(i,j)} = \phi_i \psi_j$ for all i, j, and therefore for $p = (x - \phi_1) \cdots (x - \phi_n)$ and $q = (x - \psi_1) \cdots (x - \psi_m)$ we have $r = p \otimes q$. Note that p and q are squarefree, because if we have, say, $\phi_{i_1} = \phi_{i_2}$ for some i_1, i_2, then $\rho_{\pi(i_1,1)} = \rho_{\pi(i_2,1)}$, and then $\pi(i_1, 1) = \pi(i_2, 1)$, and then $i_1 = i_2$.

Example 1. 1. Consider $r = (x-4)(x-6)(x+6)(x+9)$, i.e., $\rho_1 = 4$, $\rho_2 = 6$, $\rho_3 = -6$, $\rho_4 = -9$. A possible choice for $\pi \colon \{1,2\} \times \{1,2\} \to \{1,2,3,4\}$ is given by the table

π	1	2
1	1	2
2	3	4

(to be read like, e.g., $\pi(2,1) = 3$), because

$$\frac{\rho_{\pi(1,2)}}{\rho_{\pi(1,1)}} = \frac{\rho_2}{\rho_1} = \frac{6}{4} = \frac{-9}{-6} = \frac{\rho_4}{\rho_3} = \frac{\rho_{\pi(2,2)}}{\rho_{\pi(2,1)}}$$

and

$$\frac{\rho_{\pi(2,1)}}{\rho_{\pi(1,1)}} = \frac{\rho_3}{\rho_1} = \frac{-6}{4} = \frac{-9}{6} = \frac{\rho_4}{\rho_2} = \frac{\rho_{\pi(2,2)}}{\rho_{\pi(1,2)}}.$$

Take $\phi_1 = 15$ (for no particular reason), $\psi_1 = \frac{4}{15}$, $\phi_2 = 15\,\frac{6}{4} = \frac{45}{2}$, $\psi_2 = \frac{4}{15}\frac{(-6)}{4} = -\frac{2}{5}$. Then

$$(x - 15)(x - \tfrac{45}{2}) \otimes (x - \tfrac{4}{15})(x + \tfrac{2}{5})$$
$$= (x - 15\tfrac{4}{15})(x + 15\tfrac{2}{5})(x - \tfrac{45}{2}\tfrac{4}{15})(x + \tfrac{45}{2}\tfrac{2}{5})$$
$$= (x - 4)(x + 6)(x - 6)(x + 9),$$

as required.

In this example, no other factorizations exist except for those that are obtained by replacing p and q by $p \otimes (x - \xi)$ and $(x - \xi^{-1}) \otimes q$ for some $\xi \neq 0$. This degree of freedom is reflected by the arbitrary choice of ϕ_1.

2. The polynomial $(x-1)(x-2)(x-3)(x-4)$ cannot be written as $p \otimes q$ for two quadratic polynomials p and q, because $\frac{1}{2} \neq \frac{3}{4}$, $\frac{1}{2} \neq \frac{4}{3}$, $\frac{1}{3} \neq \frac{2}{4}$, $\frac{1}{3} \neq \frac{4}{2}$, $\frac{1}{4} \neq \frac{2}{3}$, $\frac{1}{4} \neq \frac{3}{2}$.

3. Consider $r = (x-2)(x+2)(x-3)(x+3)$, i.e., $\rho_1 = 2$, $\rho_2 = -2$, $\rho_3 = 3$, $\rho_4 = -3$. We have seen that in this case there are two distinct factorizations. They correspond to the two bijections $\pi, \pi': \{1,2\} \times \{1,2\} \to \{1,2,3,4\}$ defined via

	$(1,1)$	$(1,2)$	$(2,1)$	$(2,2)$
π	1	2	3	4
π'	1	2	4	3

4 Product Clashes

Again let $p, q \in k[x]$ be two square-free polynomials, and write $\phi_1, \ldots, \phi_n$ for the roots of p and $\psi_1, \ldots, \psi_m$ for the roots of q. Generically, the degree of $p \otimes q$ is equal to $\deg(p)\deg(q)$. It cannot be larger than this, and it is smaller if and only if there are two index pairs $(i,j) \neq (i',j')$ with $\phi_i \psi_j = \phi_{i'} \psi_{j'}$. In this case, we say that p and q have a product clash. Recall from equation (1) that $p \otimes q$ is formed as the least common multiple of the factors $x - \phi_i \psi_j$, not as their product.

Product clashes appear naturally in the computation of $p \otimes p$. For example, for $p = (x - \phi_1)(x - \phi_2)$ we have

$$p \otimes p = \mathrm{lcm}(x - \phi_1\phi_1, x - \phi_1\phi_2, x - \phi_2\phi_1, x - \phi_2\phi_2)$$
$$= (x - \phi_1\phi_1)(x - \phi_1\phi_2)(x - \phi_2\phi_2),$$

because $\phi_1\phi_2 = \phi_2\phi_1$ is a clash. More generally, if p is a square-free polynomial of degree $d \geq 2$, then $\deg(p \otimes p) \leq \frac{1}{2}d(d+1) < d^2$.

As an example that does not come from a product of the form $p \otimes p$, consider $p = (x-1)(x-2)(x-4)$ and $q = (x - \frac{1}{2})(x - \frac{1}{4})$. Here we have the clashes $1 \cdot \frac{1}{2} = 2 \cdot \frac{1}{4}$ and $2 \cdot \frac{1}{2} = 4 \cdot \frac{1}{4}$, so that $p \otimes q = (x - \frac{1}{2})(x - \frac{1}{4})(x-1)(x-2)$ only has degree 4.

In order to include product clashes into the framework of the previous section, we need to relax the requirement that π be injective. We still want it to be surjective, because every root of r must be produced by the product $\phi\psi$ of some root ϕ of p and some root ψ of q. If the ϕ_i and the ψ_j are defined according to the formulas above, it can now happen that $\phi_{i_1} = \phi_{i_2}$ for some $i_1 \neq i_2$. We therefore adjust the definition of p and q to $p = \mathrm{lcm}(x - \phi_1, \ldots, x - \phi_n)$, $q = \mathrm{lcm}(x - \psi_1, \ldots, x - \psi_m)$. Then p and q are squarefree and for the set of roots of $p \otimes q$ we obtain

$$\{\phi_i\psi_j : i = 1, \ldots, n; j = 1, \ldots, m\} = \{\rho_1, \ldots, \rho_\ell\},$$

as desired.

Example 2. 1. To find the factorization $(x - \phi_1^2)(x - \phi_1\phi_2)(x - \phi_2^2) = (x - \phi_1)(x - \phi_2) \otimes (x - \phi_1)(x - \phi_2)$, set $\rho_1 = \phi_1^2$, $\rho_2 = \phi_1\phi_2$, $\rho_3 = \phi_2^2$. Then a suitable choice for $\pi \colon \{1,2\} \times \{1,2\} \to \{1,2,3\}$ is given by

$$
\begin{array}{c|cc}
\pi & 1 & 2 \\
\hline
1 & 1 & 2 \\
2 & 2 & 3
\end{array}
$$

because

$$\frac{\rho_{\pi(1,1)}}{\rho_{\pi(1,2)}} = \frac{\rho_1}{\rho_2} = \frac{\phi_1}{\phi_2} = \frac{\rho_2}{\rho_3} = \frac{\rho_{\pi(2,1)}}{\rho_{\pi(2,2)}}$$

and

$$\frac{\rho_{\pi(1,1)}}{\rho_{\pi(2,1)}} = \frac{\rho_1}{\rho_2} = \frac{\phi_1}{\phi_2} = \frac{\rho_2}{\rho_3} = \frac{\rho_{\pi(1,2)}}{\rho_{\pi(2,2)}}.$$

2. Consider $r = (x - \frac{1}{2})(x - \frac{1}{4})(x - 1)(x - 2)$, i.e., $\rho_1 = \frac{1}{2}$, $\rho_2 = \frac{1}{4}$, $\rho_3 = 1$, $\rho_4 = 2$. A possible choice for $\pi \colon \{1,2\} \times \{1,2,3\} \to \{1,2,3,4\}$ is

$$
\begin{array}{c|ccc}
\pi & 1 & 2 & 3 \\
\hline
1 & 1 & 3 & 4 \\
2 & 2 & 1 & 3
\end{array}
$$

because

$$\left\{ \frac{\rho_{\pi(1,1)}}{\rho_{\pi(1,2)}}, \frac{\rho_{\pi(2,1)}}{\rho_{\pi(2,2)}} \right\} = \left\{ \frac{\rho_1}{\rho_3}, \frac{\rho_2}{\rho_1} \right\} = \left\{ \frac{1}{2} \right\}$$

$$\left\{ \frac{\rho_{\pi(1,1)}}{\rho_{\pi(1,3)}}, \frac{\rho_{\pi(2,1)}}{\rho_{\pi(2,3)}} \right\} = \left\{ \frac{\rho_1}{\rho_4}, \frac{\rho_2}{\rho_3} \right\} = \left\{ \frac{1}{4} \right\}$$

$$\left\{ \frac{\rho_{\pi(1,2)}}{\rho_{\pi(1,3)}}, \frac{\rho_{\pi(2,2)}}{\rho_{\pi(2,3)}} \right\} = \left\{ \frac{\rho_3}{\rho_4}, \frac{\rho_1}{\rho_3} \right\} = \left\{ \frac{1}{2} \right\}$$

and

$$\left\{ \frac{\rho_{\pi(1,1)}}{\rho_{\pi(2,1)}}, \frac{\rho_{\pi(1,2)}}{\rho_{\pi(2,2)}}, \frac{\rho_{\pi(1,3)}}{\rho_{\pi(2,3)}} \right\} = \left\{ \frac{\rho_1}{\rho_2}, \frac{\rho_3}{\rho_1}, \frac{\rho_4}{\rho_3} \right\} = \left\{ 2 \right\}$$

5 Searching for Assignments

We now turn to the question how for a given $r = (x - \rho_1) \cdots (x - \rho_\ell) \in k[x]$ we can find a map π as required. Of course, since ℓ is finite, there are only finitely many possible choices for n and m such that $n + m \le \ell \le nm$, and for each choice n, m there are only finitely many functions $\pi \colon \{1, \ldots, n\} \times \{1, \ldots, m\} \to \{1, \ldots, \ell\}$. We can simply try them all. But going through all these $(nm)^\ell$ many functions one by one would take very long.

In order to improve the efficiency of the search, we can exploit the fact that for most partial functions π it is easy to see that they cannot be extended to a total function with the required properties. We can further reduce the search space by

taking into account that the order of the roots of the factors is irrelevant, i.e., we can restrict the search to functions π with $\pi(1,1) \leq \pi(2,1) \leq \cdots \leq \pi(n,1)$ and $\pi(1,1) \leq \pi(1,2) \leq \cdots \leq \pi(1,m)$. Furthermore, because of surjectivity, the root ρ_1 must be reached, and we can choose to set $\pi(1,1) = 1$ without loss of generality. Next, discard all functions with $\pi(i,j_1) = \pi(i,j_2)$ for some i, j_1, j_2 with $j_1 \neq j_2$ or with $\pi(i_1,j) = \pi(i_2,j)$ for some i_1, i_2, j with $i_1 \neq i_2$, because these just signal some roots of a factor of r several times without providing any additional information. So we can in fact enforce $1 = \pi(1,1) < \pi(2,1) < \cdots < \pi(n,1)$ and $\pi(1,1) < \pi(1,2) < \cdots < \pi(1,m)$. Next, π is a solution iff $\pi^{\top} \colon \{1,\ldots,m\} \times \{1,\ldots,n\} \to \{1,\ldots,\ell\}$ with $\pi^{\top}(i,j) = \pi(j,i)$ is a solution. We can therefore restrict the search to functions where $n \leq m$.

The following algorithm takes these observations into account. It maintains an assignment table M which encodes a function $\pi \colon \{1,\ldots,n\} \times \{1,\ldots,m\} \to \{1,\ldots,\ell\}$ with

$$\frac{\rho_{\pi(1,j_1)}}{\rho_{\pi(1,j_2)}} = \frac{\rho_{\pi(2,j_1)}}{\rho_{\pi(2,j_2)}} = \cdots = \frac{\rho_{\pi(n,j_1)}}{\rho_{\pi(n,j_2)}}$$

for all i, j_1, j_2 and

$$\frac{\rho_{\pi(i_1,1)}}{\rho_{\pi(i_2,1)}} = \frac{\rho_{\pi(i_1,2)}}{\rho_{\pi(i_2,2)}} = \cdots = \frac{\rho_{\pi(i_1,m)}}{\rho_{\pi(i_2,m)}}.$$

for all i_1, i_2, j. At every recursion level, the candidate under consideration is extended to a function π with $\pi(n+1,1) = p$ for some p. As soon as p is chosen, there is for each $j = 2,\ldots,m$ at most one choice $q \in \{1,\ldots,\ell\}$ for the value of $\pi(n+1,j)$. The matrix M stores these values q and marks the indices j for which no q exists with $q = 0$. The result is a function $\{1,\ldots,n+1\} \times \{1,\ldots,\tilde{m}\} \to \{1,\ldots,\ell\}$ for some $\tilde{m} \leq m$. If this function is surjective, we have found a solution. Otherwise, we proceed recursively unless we already have $n+1 = \tilde{m}$, because in this case any further extension could only produce transposes of solutions that will be found at some other stage of the search.

INPUT: The roots $\rho_1,\ldots,\rho_\ell$ of some square-free polynomial $r \in k[x]$.
OUTPUT: A list of functions π as required for solving the factorization problem.

1   let $M = ((M[i,j]))_{i,j=1}^{\ell}$ be a matrix with $M[1,j] = j$ for $j = 1,\ldots,\ell$.
2   call the procedure addRow$(M,2)$ as defined below.
3   stop.

4   procedure addRow(M,n)
5   for $p = M[n-1,1]+1,\ldots,\ell$ do:
6   set the nth row of M to $(p,0,\ldots,0)$ and let J be the empty list
7   for $j = 2,\ldots,\ell$ do:
8   if $M[n-1,j] \neq 0$ and there exists $q \in \{1,\ldots,\ell\}$ such that $\rho_1/\rho_p = \rho_j/\rho_q$ and $\rho_1/\rho_j = \rho_p/\rho_q$
9   set $M[n,j] = q$ and append j to J
10   if $\{M[i,j] : i = 1,\ldots,n; j \in J\} = \{1,\ldots,\ell\}$ then:
11   report the assignment $\pi \colon \{1,\ldots,n\} \times \{1,\ldots,|J|\} \to \{1,\ldots,\ell\}$ with $\pi(i,j) = M[i,J[j]]$ for all i,j.

12   else if $|\{ j : M[n, j] \neq 0 \}| < n$ then

13   recursively call the procedure $\mathrm{addRow}(M, n+1)$

In the interest of readability, we have refrained from some obvious optimizations. For example, an actual implementation might perform some precomputation in order to improve the search for q in Step 8.

It is not hard to implement the algorithm. A Mathematica implementation by the authors is available on the website of this paper,

```
http://www.math.rutgers.edu/~zeilberg/mamarim/
                mamarimhtml/Cfac.html.
```

The relevant function is `CFiniteFactor`.

Example 3. Let $r = (x - \rho_1) \cdots (x - \rho_6)$ where $\rho_1 = -8$, $\rho_2 = -6$, $\rho_3 = -4$, $\rho_4 = -3$, $\rho_5 = -2$, $\rho_6 = -1$.

After initialisation, at the first level of the recursion, there are five choices for the first entry in the second row of M. Each of them uniquely determines the rest of the row, as follows (writing $\cdot$ for 0):

$$\begin{pmatrix} 1\ 2\ 3\ 4\ 5\ 6 \\ 2\ \cdot\ 4\ \cdot\ \cdot\ \cdot \end{pmatrix},$$

$$\begin{pmatrix} 1\ 2\ 3\ 4\ 5\ 6 \\ 3\ 4\ 5\ \cdot\ 6\ \cdot \end{pmatrix},$$

$$\begin{pmatrix} 1\ 2\ 3\ 4\ 5\ 6 \\ 4\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot \end{pmatrix},$$

$$\begin{pmatrix} 1\ 2\ 3\ 4\ 5\ 6 \\ 5\ \cdot\ 6\ \cdot\ \cdot\ \cdot \end{pmatrix},$$

$$\begin{pmatrix} 1\ 2\ 3\ 4\ 5\ 6 \\ 6\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot \end{pmatrix}.$$

The second of these matrices corresponds to a solution

$$\pi \colon \{1, 2\} \times \{1, 2, 3, 4\} \to \{1, 2, 3, 4, 5, 6\},$$

which gives rise to the factorization

$$r = (x - 1)(x - \tfrac{1}{2}) \otimes (x + 8)(x + 6)(x + 4)(x + 2),$$

while the other partial solutions cannot be continued to further solutions.

6 Multiple Roots

Let us now drop the condition that $r \in k[x]$ is square free. Write r^* for the square free part of r. It is clear from equation (1) that when $p, q \in k[x]$ are such that $r = p \otimes q$, then $r^* = p^* \otimes q^*$, where p^*, q^* denote the square free parts of p and q, respectively. It is therefore natural to first determine factorizations of the square free part r^* of r and in a second step obtain p and q from p^* and q^* (if possible) by assigning appropriate multiplicities to their roots. As the multiplicities in p or q cannot exceed those in r, there are again just finitely many candidates and we could simply try them all. And again, the search can be improved because many possibilities can be ruled out easily. In fact, the freedom for the multiplicities is so limited that we can compute them rather than search for them.

First consider the case when p^* and q^* were obtained from an injective map π, i.e., the case when there are no product clashes. In this case, each root ρ_ℓ of r^* corresponds to exactly one product $\phi_i \psi_j$ of a root ϕ_i of p^* and a root ψ_j of q^*. The multiplicities e_i of ϕ_i in p and ε_j of ψ_j in q, respectively, must be such that $e_i + \varepsilon_j - 1$ equals the multiplicity of ρ_ℓ in r. This gives a linear system of equations. Every solution of this system in the positive integers gives rise to a factorization for r, and if there is no solution for the linear system of any of the factorizations of the square-free part r^*, then r admits no factorization.

When there are product clashes, there are roots ρ of r which are obtained in several distinct ways as products of roots of p and q, for instance $\rho = \phi_{i_1} \psi_{j_1} = \phi_{i_2} \psi_{j_2}$ for some $(i_1, j_1) \neq (i_2, j_2)$. If m is the multiplicity of ρ in r, then the requirement for the multiplicities $e_{i_1}, e_{i_2}, \varepsilon_{j_1}, \varepsilon_{j_2}$ of $\phi_{i_1}, \phi_{i_2}, \psi_{j_1}, \psi_{j_2}$ in p and q, respectively, is that

$$\max(e_{i_1} + \varepsilon_{j_1} - 1, e_{i_2} + \varepsilon_{j_2} - 1) = m.$$

We obtain a system of such equations, one equation for reach root of r, and they can be solved by exhaustive search.

Example 4. 1. Let $r = (x-2)(x+2)^2(x-3)^2(x+3)^3$. We have seen earlier that the square free part r^* of r admits two distinct factorizations

$$\begin{aligned} r^* &= (x-1)(x+1) \otimes (x-2)(x+3) \\ &= (x-1)(x+1) \otimes (x-2)(x-3). \end{aligned}$$

Assigning multiplicities to the first, we get

$$\begin{aligned} &(x-1)^{e_1}(x+1)^{e_2} \otimes (x-2)^{\varepsilon_1}(x+3)^{\varepsilon_2} \\ &= (x+2)^{e_1+\varepsilon_1-1}(x-3)^{e_1+\varepsilon_2-1}(x-2)^{e_2+\varepsilon_1-1}(x+3)^{e_2+\varepsilon_2-1}. \end{aligned}$$

Comparing the exponents to those of r gives the linear system

$$\begin{aligned} e_1 + \varepsilon_1 - 1 &= 2, & e_1 + \varepsilon_2 - 1 &= 2, \\ e_2 + \varepsilon_1 - 1 &= 1, & e_2 + \varepsilon_2 - 1 &= 3, \end{aligned}$$

which has no solution. For the second factorization, we get

$$(x-1)^{e_1}(x+1)^{e_2} \otimes (x-2)^{\varepsilon_1}(x-3)^{\varepsilon_2}$$
$$= (x+2)^{e_1+\varepsilon_1-1}(x+3)^{e_1+\varepsilon_2-1}(x-2)^{e_2+\varepsilon_1-1}(x-3)^{e_2+\varepsilon_2-1}.$$

Comparing the exponents to those of r gives the linear system

$$e_1 + \varepsilon_1 - 1 = 2, \qquad e_1 + \varepsilon_2 - 1 = 3,$$
$$e_2 + \varepsilon_1 - 1 = 1, \qquad e_2 + \varepsilon_2 - 1 = 2,$$

whose unique solution in the positive integers is $e_1 = 2$, $e_2 = 1$, $\varepsilon_1 = 1$, $\varepsilon_2 = 2$, thus

$$r = (x-1)^2(x+1) \otimes (x-2)(x-3)^2.$$

2. Let $r = (x-\frac{1}{2})^2(x-\frac{1}{4})(x-1)^2(x-2)^3$. We have seen earlier that the square free part r^* of r admits the factorization

$$r^* = (x-\tfrac{1}{2})(x-\tfrac{1}{4}) \otimes (x-1)(x-2)(x-4).$$

Assigning multiplicities to the factors, we get

$$(x-\tfrac{1}{2})^{e_1}(x-\tfrac{1}{4})^{e_2} \otimes (x-1)^{\varepsilon_1}(x-2)^{\varepsilon_2}(x-4)^{\varepsilon_3}$$
$$= (x-\tfrac{1}{2})^{\max(e_1+\varepsilon_1-1,e_2+\varepsilon_2-1)}$$
$$(x-1)^{\max(e_1+\varepsilon_2-1,e_2+\varepsilon_3-1)}$$
$$(x-2)^{e_1+\varepsilon_3-1}(x-\tfrac{1}{4})^{e_2+\varepsilon_1-1}.$$

Comparing the exponents to the exponents of the factors of r gives a tropical linear system in the unknowns $e_1, e_2, \varepsilon_1, \varepsilon_2, \varepsilon_3$, which turns out to have two solutions. They correspond to the two factorizations

$$r = (x-\tfrac{1}{2})^2(x-\tfrac{1}{4}) \otimes (x-1)(x-2)(x-4)^2$$
$$= (x-\tfrac{1}{2})^2(x-\tfrac{1}{4}) \otimes (x-1)(x-2)^2(x-4)^2$$

7 When we don't want to find the roots

Sometimes our polynomials are with integer coefficients, and we prefer not to factorize them over the complex numbers. Of course, all the roots are algebraic numbers, by definition, and computer-algebra systems know how to compute with them (without "cheating" and using floating-point approximations), but it may be more convenient to find the tensor product (in the generic case: no product clashes and no repeated roots) of $p = p_0 + \cdots + p_m x^m$ and $q = q_0 + \cdots + q_n x^n$, a certain polynomial r of degree mn, as follows. If the roots of p are $\phi_1, \ldots, \phi_n$ and the roots of q are $\psi_1, \ldots, \psi_m$, then the roots of $p \otimes q$ are, of course

$$\{\, \phi_i \psi_j \mid 1 \le i \le m, 1 \le j \le n \,\}.$$

Let $P_k(p) := \sum_{i=1}^{m} \phi_i^k$ be the *power-sum symmetric functions* [9], then of course

$$P_k(p \otimes q) = P_k(p)P_k(q), \quad 1 \le k \le nm.$$

Now using *Newton's relations* (e.g. [9], Eq. I.(2.11') p. 23), one can go back and forth from the elementary symmetric functions (essentially the coefficients of the polynomial up to sign) to the power-functions, and *back,* enabling us easily to compute the tensor product without factorizing.

If you define the reverse of a polynomial p, to be $\bar{p}(x) := x^d p(1/x)$, where d is the degree of p, then $p \otimes \bar{p}$ has, of course, the factor $(x-1)^d$ but otherwise (generically) all distinct roots, unless it has good reasons not to. On the other hand, if $r = p \otimes q$ for some non-trivial polynomials p and q then $r \otimes \bar{r}$ has repeated roots other than $1-x$, and the *repetition profile* can be easily predicted as above, or "experimentally". So using this approach it is easy to test quickly whether r "factorizes" (with high probability), in the tensor-product sense. However, to actually find the factors would take more effort.

This is implemented in the Maple package accompanying this article, linked to from

```
http://www.math.rutgers.edu/~zeilberg/mamarim/
               mamarimhtml/Cfac.html.
```

The tensor product operation is called `Mul` and the testing procedure is `TestFact`.

8 Linear Combinations of Factorizations

For almost all polynomials $r \in k[x]$ there does not exist a factorization. When no factorization exists, we may wonder whether r admits a decomposition of a more general type. For example, we can ask whether there exist polynomials p_1, p_2, q_1, q_2 of degree at least two such that

$$r = \mathrm{lcm}(p_1 \otimes q_1, \ p_2 \otimes q_2).$$

Translated to the language of C-finite sequences, this means that we seek to write a given C-finite sequence $(a_n)_{n=0}^{\infty}$ as

$$a_n = b_n c_n + u_n v_n$$

for C-finite sequences $(b_n)_{n=0}^{\infty}$, $(c_n)_{n=0}^{\infty}$, $(u_n)_{n=0}^{\infty}$, $(v_n)_{n=0}^{\infty}$, none of which should satisfy a first-order recurrence in order to make the problem nontrivial.

It is not difficult to adapt the algorithm in Section 5 so that it can also discover such factorizations. Suppose that r is squarefree. Then, instead of searching for a single surjective map

$$\pi \colon \{1,\dots,n\} \times \{1,\dots,m\} \to \{1,\dots,\ell\},$$

it suffices to find two functions

$$\pi_1 \colon \{1,\dots,n_1\} \times \{1,\dots,m_1\} \to \{1,\dots,\ell\}$$
$$\pi_2 \colon \{1,\dots,n_2\} \times \{1,\dots,m_2\} \to \{1,\dots,\ell\}$$

satisfying the same conditions previously requested for π but with surjectivity replaced by $\operatorname{im}\pi_1 \cup \operatorname{im}\pi_2 = \{1,\dots,\ell\}$. Once two such maps π_1,π_2 have been found, we can construct p_1,p_2,q_1,q_2 by choosing ϕ_1^1 and ϕ_1^2 arbitrarily, setting $\psi_1^1 = \rho_{\pi_1(1,1)}/\phi_1^1$, $\psi_1^2 = \rho_{\pi_2(1,1)}/\phi_1^2$ and

$$\phi_i^1 = \phi_1^1 \frac{\rho_{\pi_1(i,1)}}{\rho_{\pi_1(1,1)}}, \qquad \psi_j^1 = \psi_1^1 \frac{\rho_{\pi_1(1,j)}}{\rho_{\pi_1(1,1)}},$$
$$\phi_i^2 = \phi_1^2 \frac{\rho_{\pi_2(i,1)}}{\rho_{\pi_2(1,1)}}, \qquad \psi_j^2 = \psi_1^2 \frac{\rho_{\pi_2(1,j)}}{\rho_{\pi_2(1,1)}}$$

for all i,j in question. Then $p_1 := \prod_{i=1}^{n_1}(x-\phi_i^1)$, $q_1 := \prod_{i=1}^{m_1}(x-\psi_j^1)$, $p_2 := \prod_{i=1}^{n_2}(x-\phi_i^2)$, $q_2 := \prod_{i=1}^{m_2}(x-\psi_j^2)$, are such that $r = \operatorname{lcm}(p_1 \otimes q_1, p_2 \otimes q_2)$.

In order to search for a pair π_1,π_2, we can search for π_1 very much like we searched for π before, and for each partial solution encountered during the recursion, initiate a search for another function π_2 which is required to hit all the indices $1,\dots,\ell$ not hit by the partial solution π_1. Note that it is fine if some indices are hit by both π_1 and π_2. The suggested modification amounts to replacing lines 12 and 13 of the algorithm from Section 5 by the following:

12   else
13   let $Q = \{M[i,j] : i = 1,\dots,n; j \in J\}$.
14   let M_2 be an $\ell \times \ell$-matrix with $(1,\dots,\ell)$ as first row.
15   call the procedure $\mathrm{addRow}_2(M_2,2,Q)$ defined below.
16   for each function π_2 it reports, report (π,π_2).
17   if no π_2 is found and $|\{j : M[n,j] \neq 0\}| < n$ then
18   recursively call $\mathrm{addRow}(M,n+1)$

19 procedure $\mathrm{addRow}_2(M,n,Q)$
20   [lines 5–9 literally as in the definition of addRow]
21   if $\{1,\dots,\ell\} \setminus Q \subseteq \{M[i,j] : i = 1,\dots,n; j \in J\}$ then:
22   [literally as line 11 in the definition of addRow]
23   else if $|\{j : M[n,j] \neq 0\}| < n$ then
24   recursively call $\mathrm{addRow}_2(M,n+1,Q)$.

This settles the case of square free input. The extension to arbitrary polynomials is like in the previous section. For every factorization of the square free part we can assign variables for the multiplicities of all the roots and compare the resulting multiplicities for $\operatorname{lcm}(p_1 \otimes q_1, p_2 \otimes q_2)$ to those of r. This gives again a tropical linear system of equations which can be solved with Grigoriev's algorithm [5].

Example 5. The polynomial $r = (x-1)(x-2)(x-3)(x-4)(x-6)(x-12)$ cannot be written as $r = p \otimes q$ for some $p, q \in k[x]$. However, we have the representation

$$r = \mathrm{lcm}(p_1 \otimes q_1, p_2 \otimes q_2)$$

for

$$p_1 = (x-1)(x-2), \qquad p_2 = (x-1)(x-3),$$
$$q_1 = (x-2)(x-3), \qquad q_2 = (x-1)(x-4).$$

Note that the roots 3 and 4 of r are produced by both $p_1 \otimes q_1$ and $p_2 \otimes q_2$.

9 Examples

Our main motivation for studying the factorization problem for C-finite sequences are two interesting identities that can be interpreted as such factorizations. They both originate from the transfer matrix method.

The first is a tiling problem studied in [7, 3], and more recently in [18]. Given a rectangle of size $m \times n$, the question is in how many different ways we can fill it using tiles of size 2×1 or 1×2. If n and m are even, it turns out that

$$T_{n,m} = 2^{nm/2} \prod_{i=1}^{m/2} \prod_{j=1}^{n/2} \left(z_\square^2 \cos^2\left(\frac{i\pi}{m+1}\right) + z_\boxplus^2 \cos^2\left(\frac{j\pi}{n+1}\right) \right)$$

is a bivariate polynomial in the variables $z_\square, z_\boxplus$ where the coefficient of a monomial $z_\square^u z_\boxplus^v$ is exactly the number of tilings of the $m \times n$ rectangle that uses exactly u tiles of size 2×1 and v tiles of size 1×2. The transfer matrix method can be used to prove this result automatically for every fixed m and arbitrary n (or vice versa). For every fixed choice of m (say), it delivers a polynomial r which encodes a recurrence for $(T_{n,m})_{n=0}^{\infty}$. For every fixed $i \in \{1, \ldots, m\}$, the sequence

$$2^{n/2} \prod_{j=1}^{n} \left(z_\square^2 \cos^2\left(\frac{i\pi}{m+1}\right) + z_\boxplus^2 \cos^2\left(\frac{j\pi}{n+1}\right) \right)$$
$$= \frac{1}{w} z_\boxplus^n T_n(\sqrt{w}) + \left(1 - \frac{1}{w}\right) z_\boxplus^n U_n(\sqrt{w})$$

with $w = 1 + \left(\frac{z_\square}{z_\boxplus} \cos\left(\frac{i\pi}{m+1}\right)\right)^2$ and T_n and U_n the Chebyshev polynomials of the first and second kind, is C-finite with respect to n. An annihilating polynomial is

$$p_i = x^2 - 2\left(z_\boxplus^2 + 2z_\square^2 \cos^2\left(\frac{i\pi}{2m+1}\right)\right)x + z_\boxplus^4.$$

The formula for $T_{n,m}$ can be proven for each particular choice of m and arbitrary n by checking $r = p_1 \otimes \cdots \otimes p_m$ and comparing the first 2^m initial terms. While the

standard algorithms can confirm the correctness of some conjectured factorization, the algorithm described in the present paper can help discover the factorization in the first place, taking only r as input. Fisher, Temperly [3] or Kasteleyn [7] would probably have found it useful back in the 1960s to apply the algorithm to $m = 2, 4, 6, 8, 10$ and to detect the general pattern from the outputs.

The second identity which motivated our study has a similar nature. It describes the Ising model on an $n \times m$ grid wrapped around a torus [10, 14]. Starting from a certain model in statistical physics that we do not want to explain here, the transfer matrix method produces for every fixed $m \in \mathbb{N}$ an annihilating polynomial r of degree 2^m for a certain C-finite sequence in n. The asymptotic behaviour of this sequence for $n \to \infty$ is of interest. In view of Theorem 2, it is goverend by the root of r with the largest absolute value. Onsager discovered that this largest root of r is equal to

$$(2\sinh(2v))^{m/2} \exp\left(\tfrac{1}{2}(\gamma_1 + \gamma_3 + \cdots + \gamma_{2m-1})\right)$$

where v is some physical constant and γ_k is defined as

$$\gamma_k = \operatorname{arccosh}\left(\cosh(2v)\coth(2v) - \cos(\tfrac{\pi k}{m})\right)$$

for $k = 1, 3, \ldots, 2m - 1$ (compare eq. (V.5.1) (p. 131) in [14]).

Let us translate these formulas to a more familiar form. First note that because of periodicity and symmetry of the cosine, we have $\gamma_k = \gamma_{2m-k}$ for $k = 1, 3, \ldots$. Hence each of the γ_k in the argument of exp appears twice, except the middle term γ_m, which only appears for odd m. Set $z = \exp(v)$ and $x_k = \exp(\gamma_k)$ for $k = 1, 3, \ldots, 2m - 1$. Then $2\sinh(2v) = z^2 - z^{-2}$, and Onsager's expression for the largest root of r simplifies to

$$\begin{cases} (z^2 + z^{-2})^{m/2} x_1 x_3 \cdots x_{m-1} & \text{if } m \text{ is even} \\ (z^2 + z^{-2})^{(m-1)/2}(1 + z^2) x_1 x_3 \cdots x_{m-1} & \text{if } m \text{ is odd.} \end{cases}$$

For the second case we have used $\sqrt{(z^2 + z^{-2})x_m} = 1 + z^2$. The equation for γ_k says that x_k is a root of

$$p_k := x^2 + \left(2\cos(\tfrac{\pi k}{m}) - \frac{(z^4 + 1)^2}{(z^4 - 1)z^2}\right)x + 1.$$

Set $q = x - (z^2 - z^{-2})^{m/2}$ when m is even and set $q = x - (z^2 - z^{-2})^{(m-1)/2}(1 + z^2)$ when m is odd. Then Onsager's formula says that the largest root of r is equal to the largest root of $q \otimes p_1 \otimes p_3 \otimes \cdots \otimes p_{m-1}$.

In fact, the polynomial $q \otimes p_1 \otimes p_3 \otimes \cdots \otimes p_{m-1} \in \mathbb{Q}(z)[x]$ happens to be exactly the irreducible factor of $r \in \mathbb{Q}(z)[x]$ corresponding to the largest root of r. Therefore, our algorithm applied to this irreducible factor of r could have helped Onsager discover his formula.

References

1. Yongjae Cha. Closed form solutions of linear difference equations in terms of symmetric products. *Journal of Symbolic Computation*, 60:62–77, 2014.
2. Graham R. Everest and Alf J. van der Poorten. Factorization in the ring of exponential polynomials. *Proceedings of the AMS*, 125(5):1293–1298, 1997.
3. M. Fisher and H. Temperley. Dimer problems in statistical mechanics–an exact result. *Philos. Mag.*, 6:1061–1063, 1961.
4. Guoqiang Ge. *Algorithms related to multiplicative representations of algebraic numbers*. PhD thesis, U.C. Berkeley, 1993.
5. Dima Grigoriev. Complexity of solving tropical linear systems. *Computational Complexity*, 22:71–88, 2013.
6. Sabrina Hessinger. *Computing Galois Groups of Linear Differential Equations of Order Four*. PhD thesis, North Carolina State University, 1997.
7. P. W. Kasteleyn. The statistics of dimers on a lattice: I. the number of dimer arrangements in a quadratic lattice. *Physica*, 27:1209–1225, 1961.
8. Manuel Kauers and Peter Paule. *The Concrete Tetrahedron*. Springer, 2011.
9. Ian Macdonald. *Symmetric Functions and Hall Polynomials*. Clarendon Press, Oxford, 2nd edition, 1995.
10. Lars Onsager. Crystal statistics, I. a two-dimensional model with an order-disorder transition. *Physical Review*, 65:117–149, 1944.
11. Joseph F. Ritt. A factorization theory for functions $\sum_{i=1}^{n} a_i e^{\alpha_i x}$. *Transactions of the AMS*, 29(3):584–596, 1927.
12. Michael F. Singer. Solving homogeneous linear differential equations in terms of second order linear differential equations. *American Journal of Mathematics*, 107(3):663–696, 1985.
13. Richard P. Stanley. *Enumerative Combinatorics, Volume 2*. Cambridge Studies in Advanced Mathematics 62. Cambridge University Press, 1999.
14. Colin J. Thompson. *Mathematical Statistical Mechanics*. Princeton University Press, 1972.
15. Mark van Hoeij. Decomposing a 4th order linear differential equation as a symmetric product. *Banach Center Publications*, 58:89–96, 2002.
16. Mark van Hoeij. Solving third order lienar differential equations in terms of second order equations. In *Proceedings of ISSAC'07*, pages 355–360, 2007.
17. Doron Zeilberger. A holonomic systems approach to special function identities. *Journal of Computational and Applied Mathematics*, 32:321–368, 1990.
18. Doron Zeilberger. CounTilings. The Personal Journal of Shalosh B. Ekhad and Doron Zeilberger, 2006.
19. Doron Zeilberger. The C-finite ansatz. *The Ramanujan Journal*, 31(1):23–32, 2013.